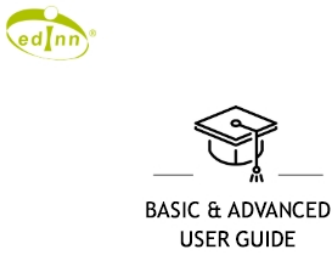
edinn Platform: User Guide

Table of contents

Introduction



This document is the complete guide of the **edinn platform** version **2021-01**. We recommend accessing also to the edinn Academy to find additional documents and files which are cited in this guide.

The edinn platform is a set of technological tools which objective is to be the **easiest**, **most powerful** and **affordable** manner to reach the maximum **Productivity** and **Total Efficiency**, through helping its users to **control, manage and improve their operations**. For this, it provides:

- A **Production Control (OEE)** and **Manufacturing Execution System (MES)** for the **4th Industrial Revolution**.
- A **Work Management** to **control**, **manage** and **improve** any type of **work** and **telework**.
- **Operational Intelligence** supported by **Big Data** and **Artificial Intelligence**.

It includes a complete set of applications or modules and is open as it is expandable to any need thanks to the **Add-ons** developed with the edinn API & UDL.

Click here to download this guide in PDF format (only available from the on-line version).

Click here to request for more information, something to improve or any doubt.

WARNING: You could be reading an obsolete version of this document because of your browser's cache. Please delete your browser's cache or type CTRL+F5 in **every document** of this guide.

Discover new version

Discover what is new in this 2024-07 version. Please read the Revision notes to know all the

changes in detail and to check if important tasks should be performed to upgrade.

See what was new in previous versions here.

Gain benefits today with Artificial Intelligence and your edinn

In 2021, we began a significant investment in AI, which has made it possible for your Edinn, powered by AI projects, to provide you with benefits in:

- **Data analysis and custom AI projects with Python and Jupyter notebooks**.
- **Intelligent assistants** to:

        - Help you improve your production.
        - Assist your users.
        - Automate manual processes.
    - **Artificial vision** to automate:
        - Classifications, such as sizes, qualities, or products.
        - Visual inspections.
    - **Predictions** of:
        - Failures, to prevent them.
        - Time series to predict, for example, ratios (OEE and others), stocks, and consumption.

Thousands of Hours of Work in Improvements and Corrections

Since the Edinn platform is a standard product—one of the few in the categories of Industry 4.0 (Smart Factories) and Work Management—every correction or improvement made in any of its installations is distributed in a short time to the rest of the installations.

This version also includes these improvements:

- **Archiving Processes and Areas**: As clients use the Edinn system for longer periods, they require features that allow them to mark processes as no longer productive and archive areas they no longer use. This will enable them to save time in their day-to-day activities and

improve the overall system performance while maintaining the ability to access their valuable information.

- **More Options in the Production Planner**: It is now possible to decide if the estimated completion time of orders is calculated based on OEE or nominal speed and whether to update this information continuously or only at shift changes. For more info, see Calculations.
- **More Elegant and Powerful Path**: The production path chart is now configurable and can be visually more elegant, with more options like displaying the forecast and even the activity rate target. More information can be found in the server help, section 8, Web Server, under "Reports Customization."
- **Improvements and More Options in Reports**:
  - New option to display quantiles in the Quality\Histogram chart.
  - Improvement in the "Top" filter of the States\Summary report, which now shows exactly the number of records indicated because it is applied after the other filters.

        - New "Only Required" option in the Analysis\Evolution report allows for clearer and faster analysis by showing only required periods, i.e., excluding holidays, weekends, etc.

    - **Improvements in Remote Monitoring via Edinn OPC Bridge**: It has become more robust for networks with issues, and signal management has been improved, including writing them to the PLCs.

    - **Improved Overall Performance**: Internal deadlocks in Big Data that sometimes caused the entire system to slow down have been reduced.

    - **More Robust and Reliable**: Hundreds of small corrections are important for being one of the best MES / MOM / Industry 4.0 systems on the market.

Please see the complete list of improvements and corrections.

Please, remember that by using this platform you are fully accepting its Terms and Conditions.

Previous versions

Discover what was new in version:

- 2021-01
- 2020-07
- 2019-11
- 2019-10
- 2019-01

Version 2021-01

Discover what is new in this 2021-01 version. Please read the Revision notes to know all the changes in detail and to check if important tasks should be performed to upgrade.

See what was new in previous versions here.

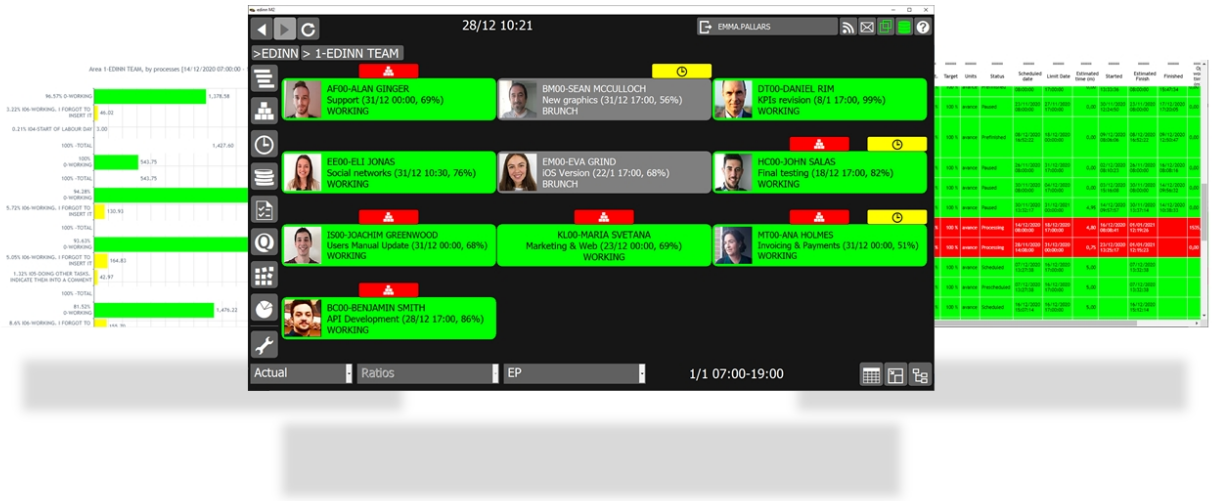
New Night Mode, better Visuals and Performance

We keep improving the **reports** for them to be the best tool:

- **New Night Mode:** switches to dark mode which is more suitable for low light environments.
- **Better Visuals**: increase in fonts sizes in widgets and the path. Better reescaling, buttons and filtering fields.
- **Better Performance** *: now dashboards and favorites are refreshed faster and consuming less resources from the server.

** Requires a manual intervention in the update*

process of the server.



New edinn Work Management module: the future of (tele) work

The edinn platform has been enhanced to also manage office work, include the whole team and extend the performance and efficiency gains to the whole organisation.

A revolutionary *edinnovation* that will provide you:

- **Increase performance and motivation** *: we humans, without a concrete measurable goal, without a "score" to beat, tend to lose performance. With edinn, we measure from the time worked, through the results obtained, to the fulfilment of commitment dates.

In an easy and useful way, almost without noticing it.

- **Higher performance in telework, in real time** *: thanks to greater visibility among the members of the organisation... Edinn brings the team together! Because we see colleagues almost as if we were in the office: having lunch? working? Everything in real time.
- **Results-oriented management**: say goodbye to annoying report sheets or timetables, which are often filled in by memory at the last minute.
- **Greater confidence for supervisors***: the task and its results, the progress, the commitment date, the status and the workload are all seen in real time and in aggregate reports.
- **Manage all types of office tasks**: both recurring tasks and projects.
- **The information you need**: because it incorporates, for example, reports of hours dedicated to each project and task by each person, of open projects or workload of each person with their commitment dates and progress percentages.

** These statements are based on estimates because they are in the process of being evaluated in order*

to confirm them with indicators.

Thousands of hours of improvement and corrections

As the edinn platform is a standard product, one of the only ones in the categories of Industry 4.0 or Smart Factories and Work Management, each correction or improvement made at any of its facilities is distributed, in a short time, to the rest of the facilities.

- **Work and telework management**: hundreds of changes, such as Operation Time, applied strategically so that you can manage and improve any work with edinn.
- **Avoid a high % of failures**: edinn Envision (failures prediction with Artificial Intelligence) has been improved dramatically: multithreading, more internal topologies, self-

training until objectives are achieved and many more.

- **Locate tasks and orders more easily**: thanks to the fact that orders can now be associated with areas. It also allows you to have orders not assigned to any process, avoiding the need to use mailbox processes.
- **Faster and easier management of processes, resources, areas, devices and states**: with the right mouse button, from the main window, only for administrators. Now you can also relate states to processes in a massive way in PSR Relations.
- **New themes and processes images**: for a more pleasant user experience in the terminals and in the reports. In addition, processes can now have images both in the tiles and in the process card.
- **Import of tasks and orders with copy and paste**: copy any task list, from excel, a notepad or from anywhere, and paste it in edinn to create in a few seconds tasks or orders in planning.
- **New "Remember me" option in the office login**: for a direct and faster entry, with total security.
- **Higher performance and multi-platform ready**: we have

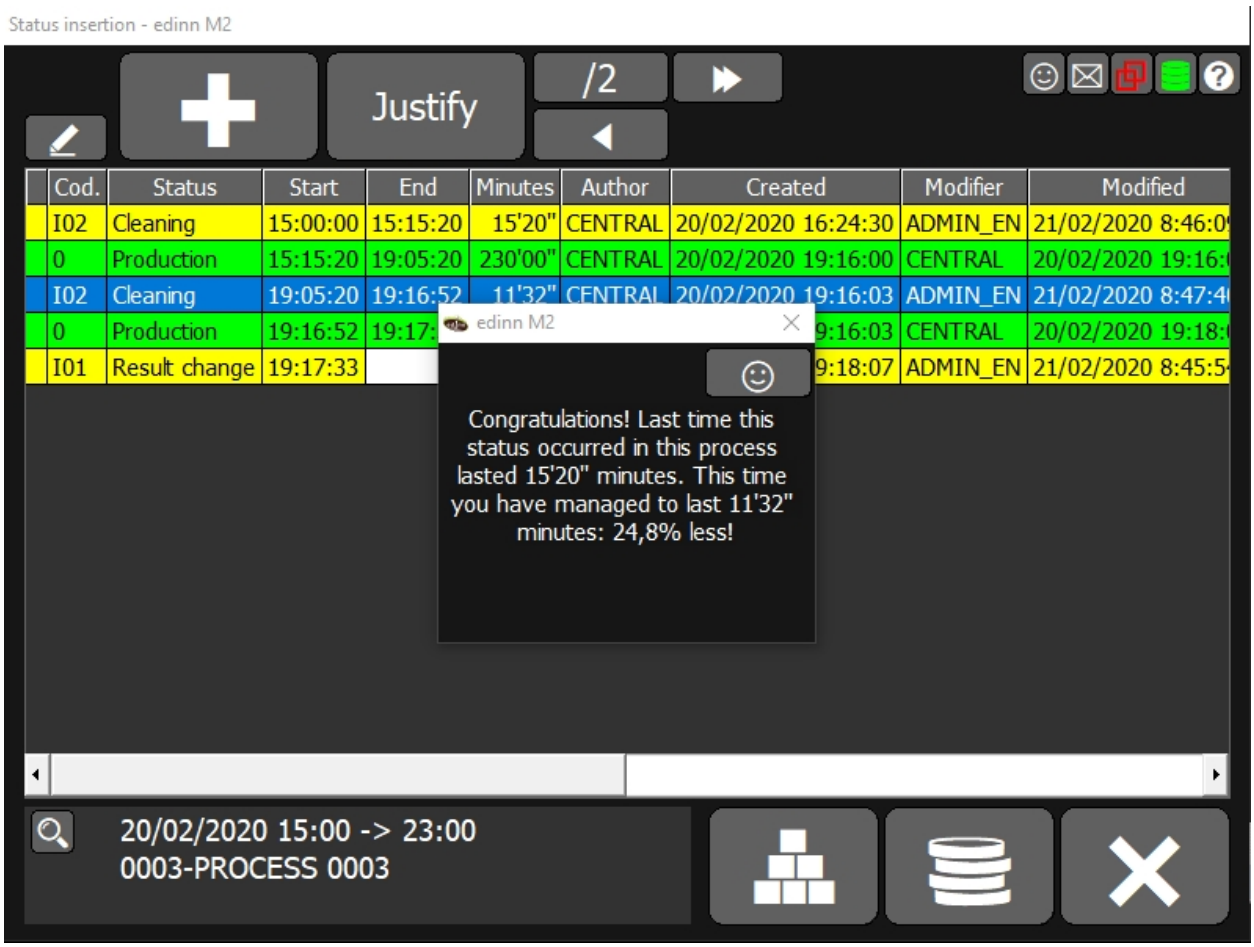
worked thousands of hours internally to improve performance, especially in the cloud, and for future multi-platform versions.

Please see the complete list of improvements and corrections.

Please, remember that by using this platform you are fully accepting its Terms and Conditions.

Version 2020-07

in this 2020-07 version. Please read the Revision notes to know all the changes in detail and to check if important tasks should be performed for the upgrade.



Recommendations: We help you stay motivated and to perform better

Useful recommendations, simple and just when you need them. To provide this information, the edinn platform makes use of all its capabilities: calculation of KPIs and trends, Big Data and Artificial Intelligence.

Introducing the new edinn Community version!

Every day more clients, partners, educational institutions, start-ups and professionals, ask edinn for environments to demonstrate, learn, develop and test. In response to their requests, edinn publishes the edinn Community version that provides a permanent free license for 1 process and the Production Control, Big Data and Artificial Intelligence and Development Platform (API & UDL) modules. This, together with the integration with FlexSim, the leading process simulator, makes of edinn possibly the best choice for the community of professionals of Industry 4.0.

Thousands of hours of improvement and corrections

As the edinn M2 platform is a **standard product**, one of the very few in the category of Industry 4.0 and Smart Factories, each correction or improvement done in any of its installations is distributed, in a very short time, to the rest of installations. This version includes, additionally, these improvements:

- Improved Artificial Intelligence (Envision) for a failure prediction which can result in incredible savings.
- API messages and integration can be now tested directly from the reports.
- Scheduling routes can now indicate areas instead of processes.

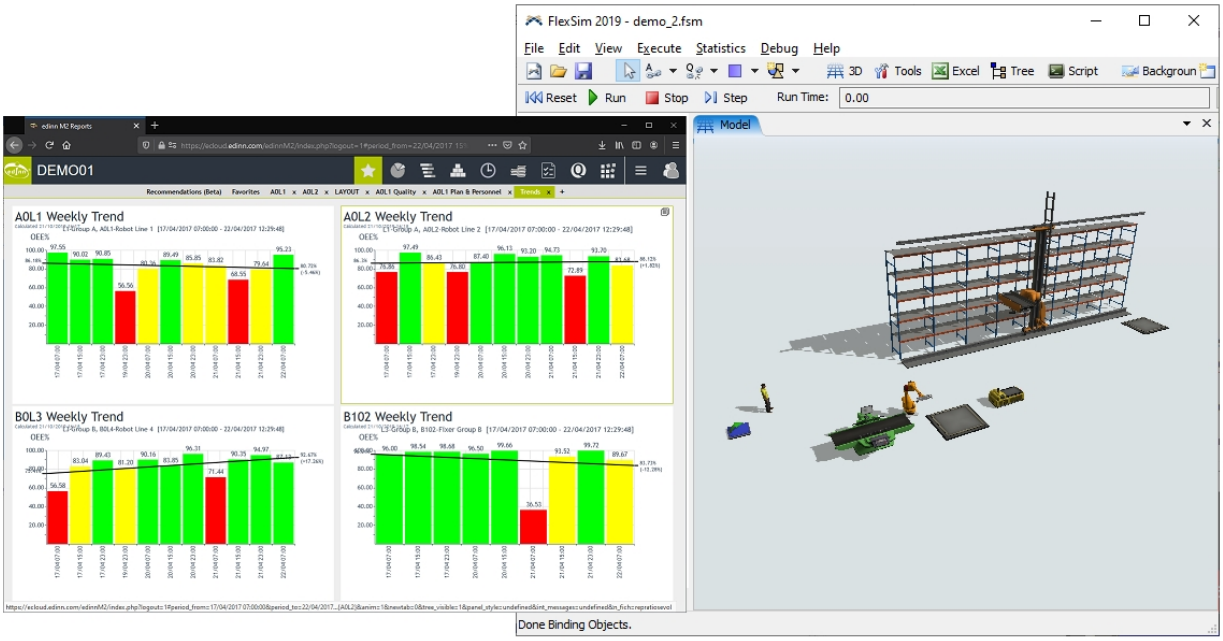
- There is a visual calendar, allowing to see how statuses are scheduled.
- Autocontrol tasks can now be disactivated, at the level of the task, or at the level of a process.
- The report of consumptions summary can now be grouped by working order, and then clic on a record to see the detail.
- Now the system manages the database and other automatic and periodical backups, reducing their space in disk. It also copies the backups to folders which can be external to the server, reducing the possibility of data loss.
- Now you can use the terminal in more devices (tablets, Apple Mac, etc.) and with slower networks; and access the Console and manage your platform even in cloud servers, as now it supports and integrates remote applications.

Please see the complete list of corrections and improvements.

Please, remember that by using this platform you are fully accepting its Terms and Conditions.

Version 2019-11

Discover what is new in this 2019-11 version. Please read the Revision notes to know all the changes in detail and to check if important tasks should be performed for the upgrade.



Discover edinn by simulating any process

Now you can simulate any type of process, connect it to edinn, in the cloud or locally and see all* the edinn benefits. You do not need any more to have a factory, to acquire any type of hardware nor to have any type of process: now you can virtually have them all!

We have thought in our community and because of that we have improved edinn to integrate also with the free version of FlexSim, leader software in process simulation. This is useful for testing, training, webinars, presentations and even to analyze productivity and efficiency of production models.

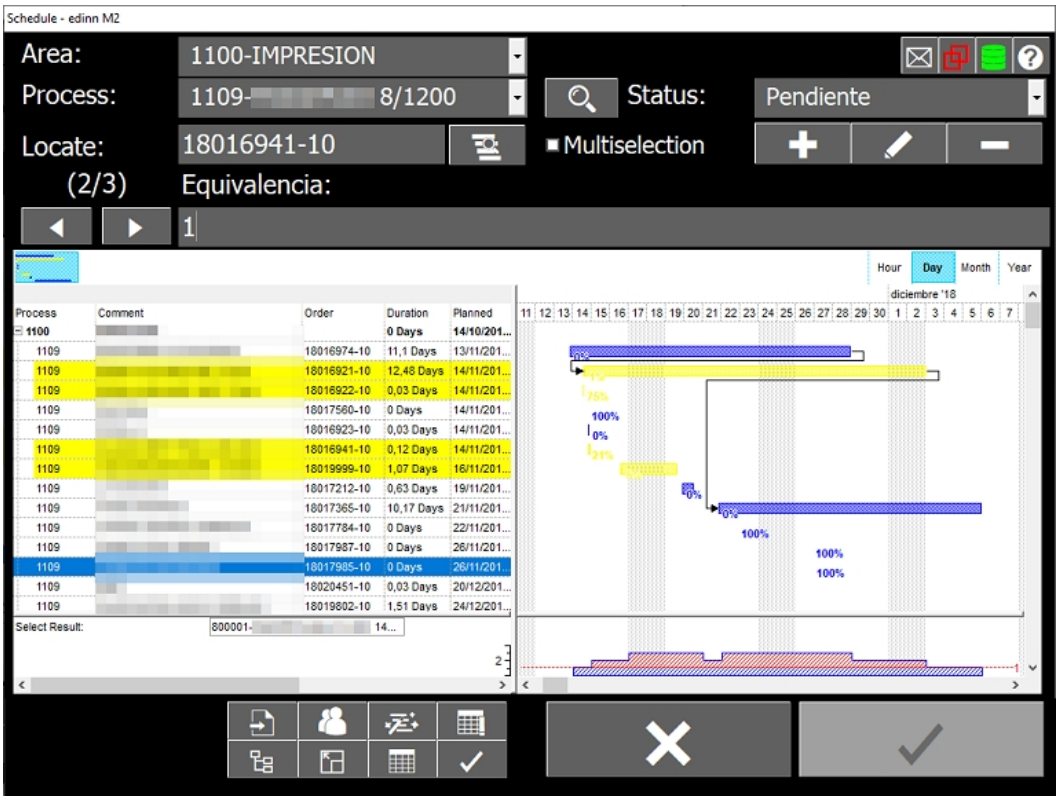
* Trying all the functionalities could require additional software licenses.

Visually better

In previous versions we dramatically increased the speed when showing windows in the operation terminal, we improved visuals and added the dark mode. We have continued with that trend and, in this version, we have improved visuals even more. Do not forget to try it in dark mode!

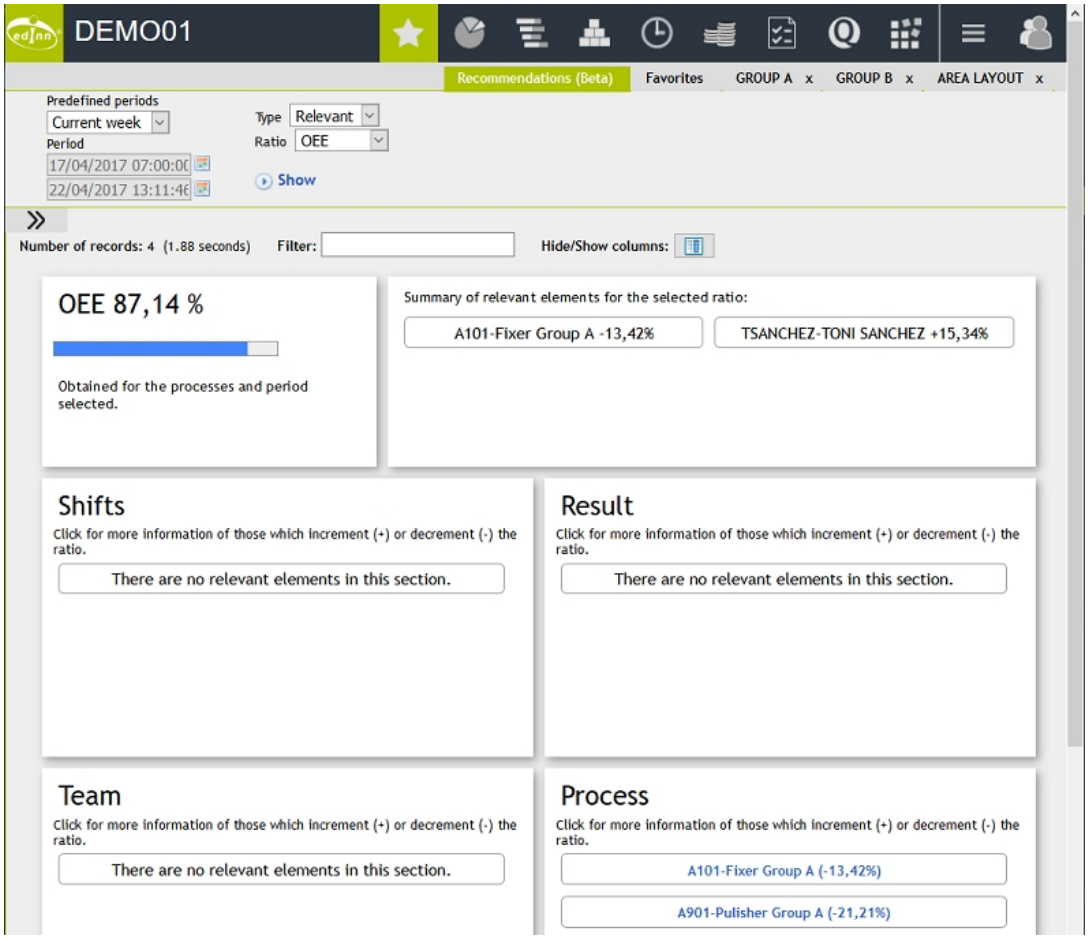
Version 2019-10

Discover what is new in this 2019-10 version. Please read the Revision notes to know all the changes in detail and to check if important tasks should be performed for the upgrade.



Schedule your orders or tasks

The Production Planning Module will propose you when and in which processes should working orders be performed to obtain the best KPIs. We have developed, together with partners and clients, a complete set of functionalities which turn this module into an Advanced Planner and Scheduler (APS).



Recommendations and Diagnosis reports

Do you know which shift, result, team, process or resources (persons) are helping to increase OEE, which are not, and how much? Once again, edinn is the first to incorporate, in its standard functionality, the Recommendations Relevant report, based on Big Data and statistics, to answer all these questions from a single point, helping you to detect best practices and areas of improvement.

In addition, the Diagnosis report will allow you to know how the edinn platform is being used.

* Only available with the Big Data Module.

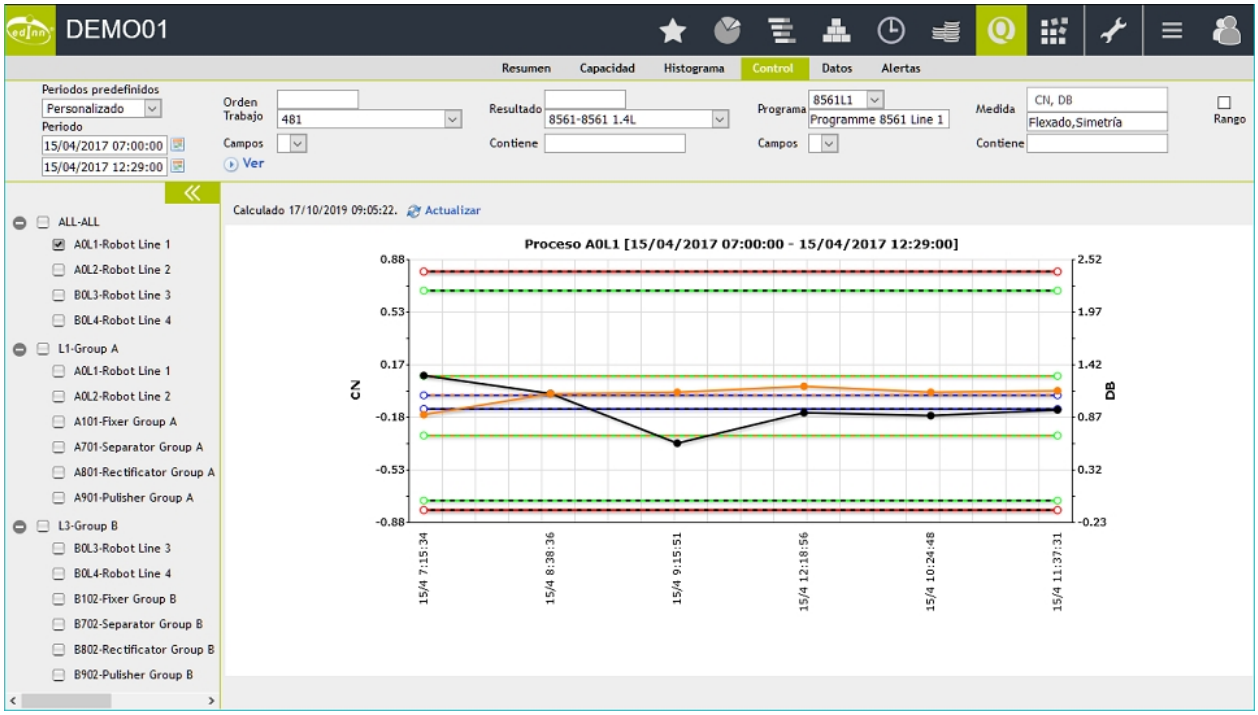
Much faster to show, dark mode and modernized

Now the edinn operation terminal is up to 50% faster to show. In addition, as black background visualizations have become the standard for many information screens, you have now available the dark theme; select it from Devices Configuration. As if this was not enough, we have also modernized the aspect with vectorial icons which are shown perfectly in all resolutions.

Create dashboards with 1 click: they are faster and with better appearance

Create a complete dashboard with the basic information, with just 1 click from Processes Configuration, and send it to the desired terminals (devices). This version only incorporates 1 template, but in future versions we will incorporate more. We need to know if this functionality is useful for you and if you need more templates of dashboards.

In addition, now widgets are a 10% faster and consume approximately a 15% less memory in the server. This, together with the Big Data & AI Module which in some cases accelerates widgets to last up to 1% of what they used to last, and to the performance improvements which we applied in previous versions, make of edinn dashboards and reports one of the most powerful tools in its market.

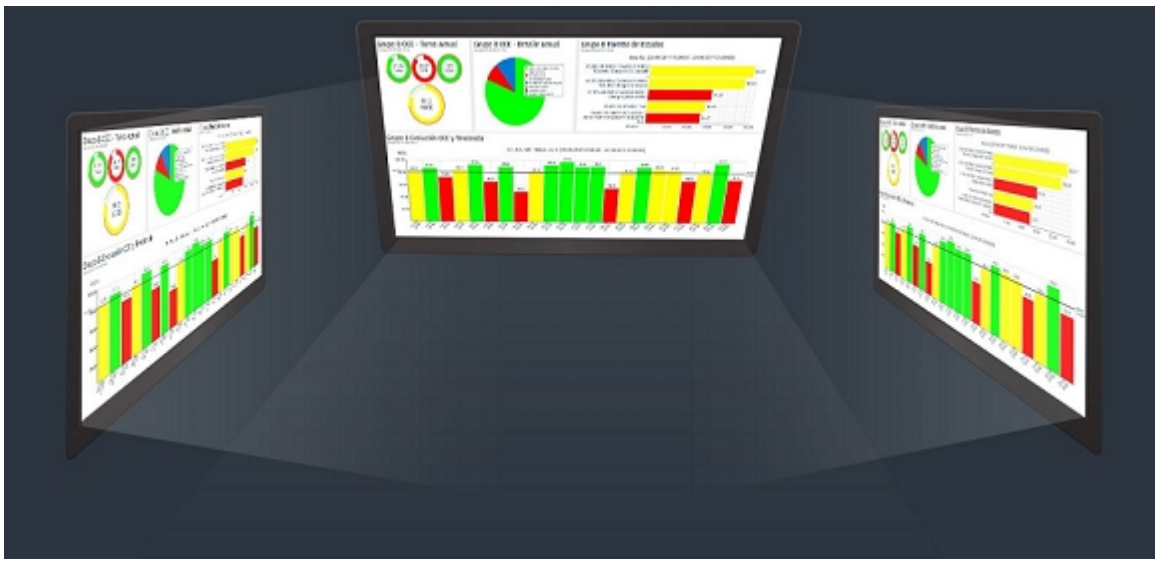


More IIoT (Industrial Internet Of Things) power

Graph any variable that you need to control and compare it visually with any other. You have new signals in the monitor and improved graphics. All that together with internal improvements for more power on IIoT.

Version 2019-01

Discover what is new in this 2019-01 version. Please read the Revision notes to know all the changes in detail and to check if important tasks should be performed for the upgrade.



Your operation terminals... Your visual plant! *

The new screen saver feature in the terminal, not only protects the screens, but will also show your favourite dashboards in full screen, even when no one has logged into the system.

How? From the reports, create dashboards and associate them to devices by clicking on the title of the tab, or create the dashboards automatically from the processes configuration. Then, activate the screen saver in your configuration of devices.

* Requires some installed components in the operation terminal, like Microsoft Internet Explorer in their most updated versions, and further configuration.

Dashboards on the terminals!

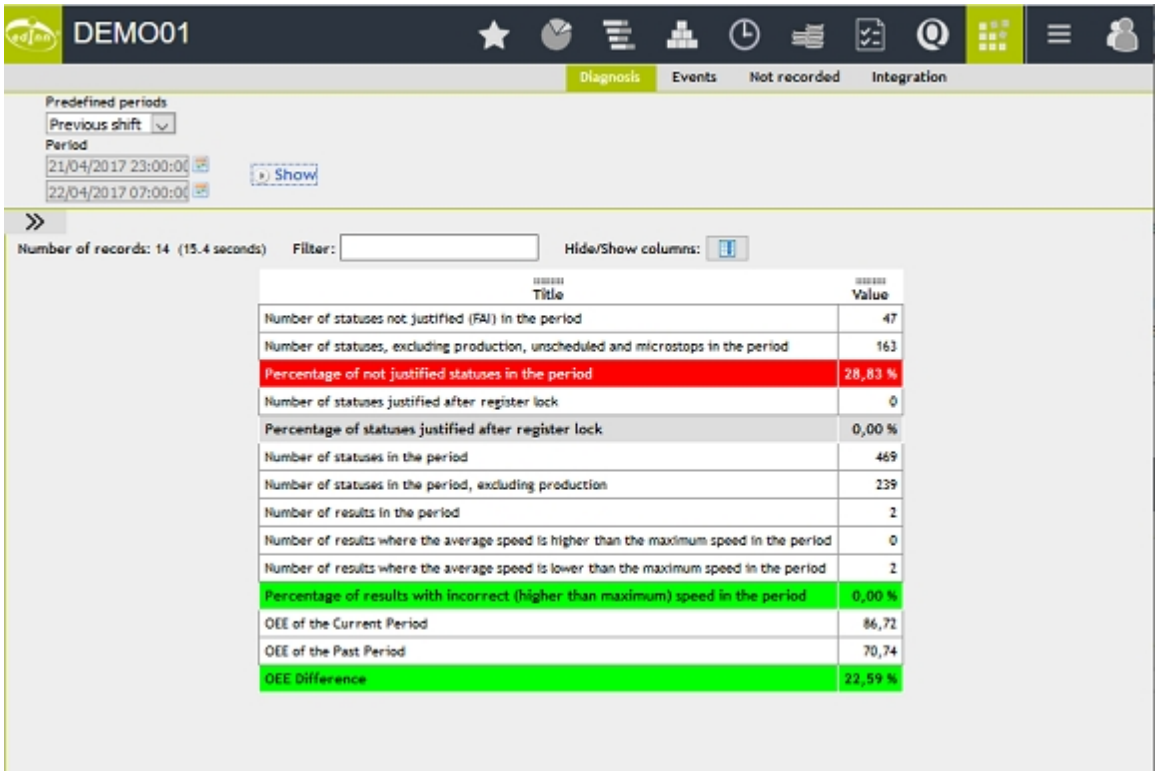
Now you can view, from your operation terminals, the dashboards that you need. Made with up to 41 different widgets!
To do this, you just have to double click on the tab of the dashboard from the web reports.

* Requires some installed components in the operation terminal, like Microsoft Internet Explorer in their most updated versions, and further configuration.

Visually atractive dashboards and yet even faster, thanks to Big Data*!

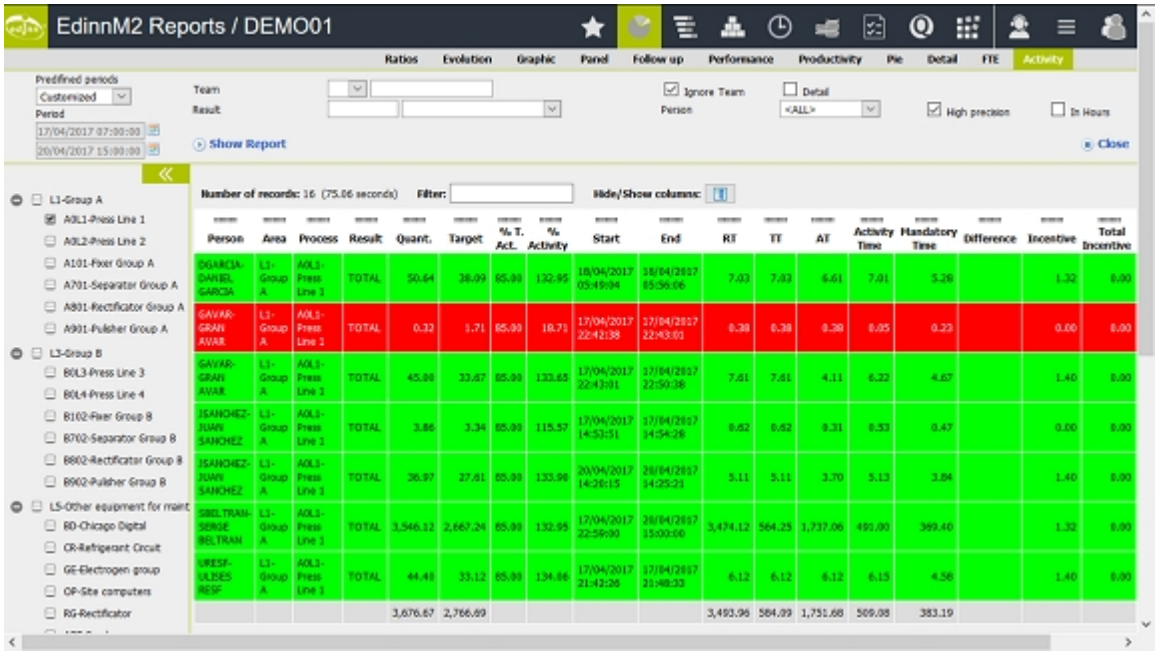
We did it again: dashboards are more attractive and yet even faster. In some cases up to 90% less time, or even less.

* Could require the Big Data and Artificial Intelligence Module.



Is your organization using edinn correctly? Check the new Diagnosis Report

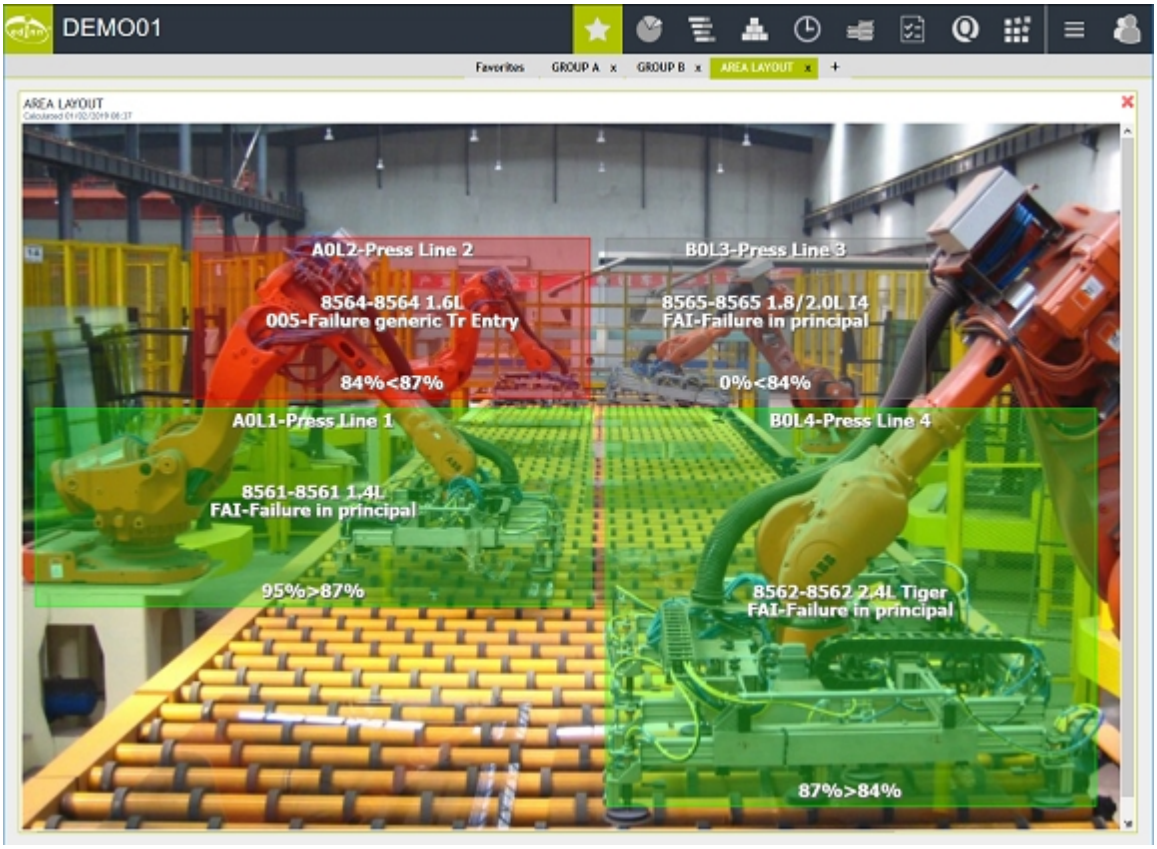
The new Diagnosis report shows the first 3 indicators to determine if your organization is using edinn correctly.



Improved activity ratio calculation

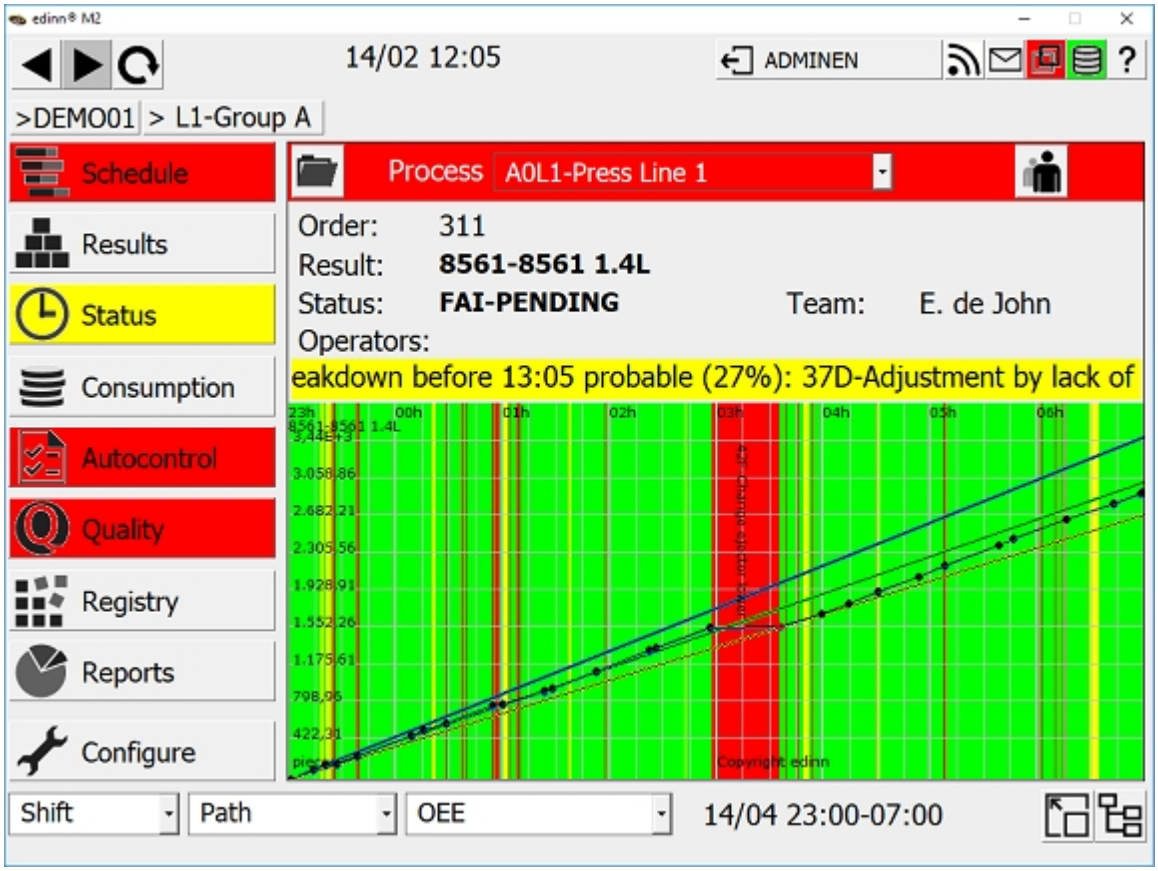
Calculation of the incentives is more precise since the percentage of activity ratio can now

be configured in addition to per state, per process.



Visualization with augmented reality

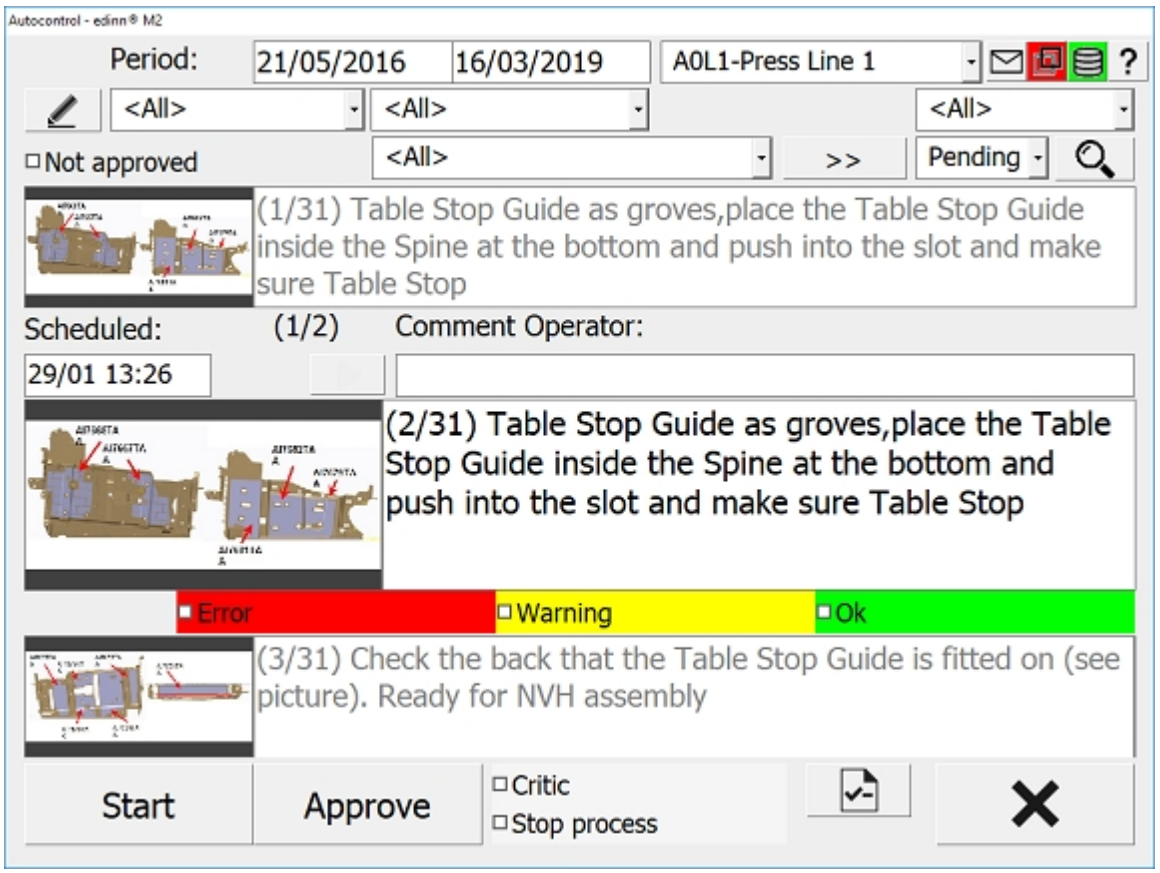
Visualize all the information of your processes with live real time images in the Analysis, Ratios report.



Better failure prediction with Artificial Intelligence and Big Data*

We have improved our technology, and now the neural network allows to predict failures with 4 hours of advance.

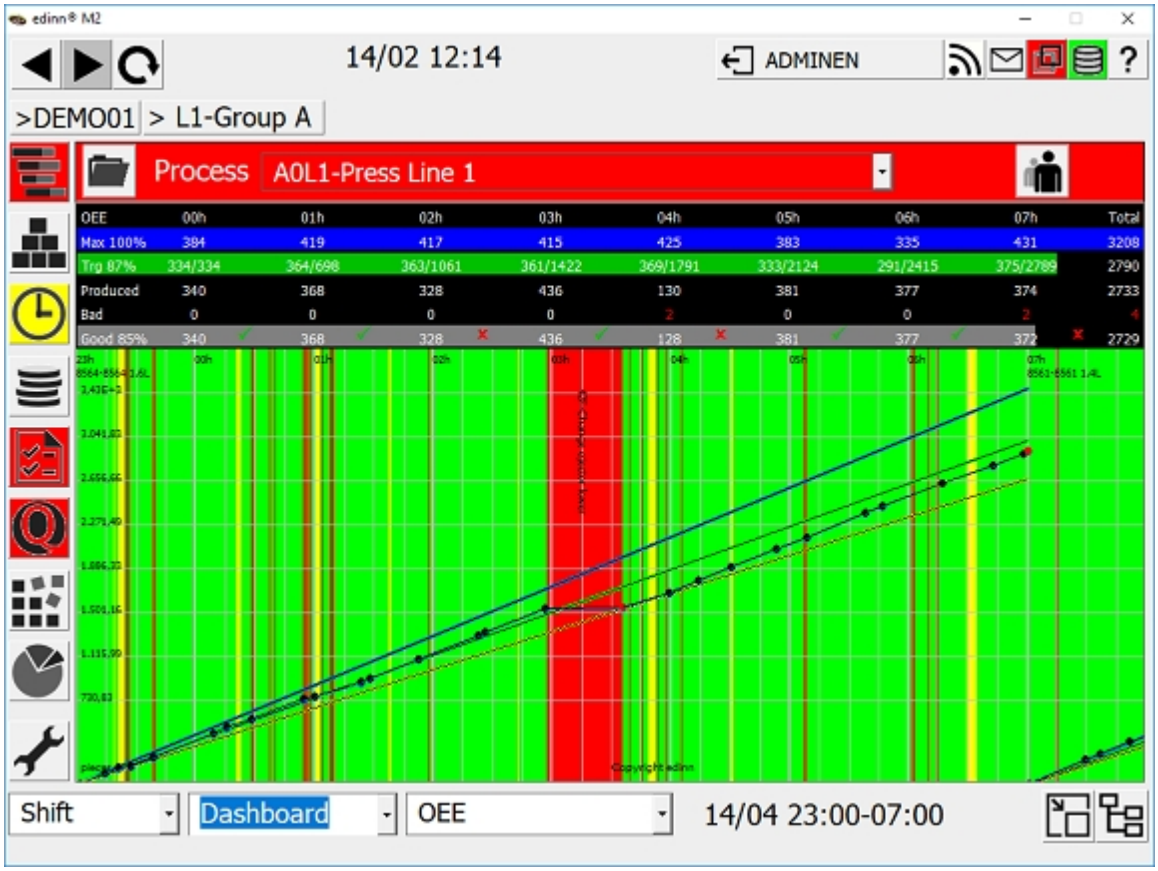
* Requires the Big Data and Artificial Intelligence Module.



Renewed Autocontrol*

Optimized follow-up and management of autocontrol* tasks with improvements in the display of alerts and step by step images and explanations.

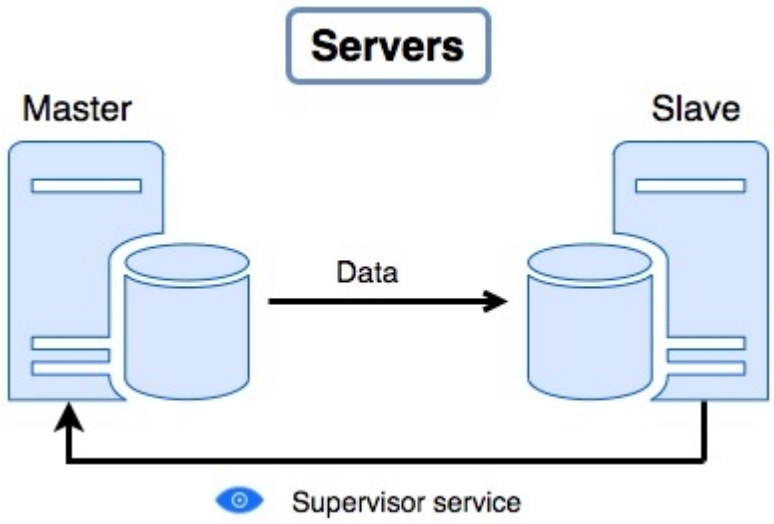
* Requires the Autocontrol Module.



More ready than ever to work with Big Data*

Higher speed in data queries and the possibility to carry out different broader and more complex dashboards developing with the API.

* Requires the Big Data and Artificial Intelligence Module.



Automatic Recovery

For mission critital installations, where the edinn server can never be down, you can now have Automatic Recovery which will help you to reach the highest availability of your edinn server.

Basic Concepts

What is Industry 4.0?

Industry 4.0 refers to real-time data exchange enabled by the interconnection of all the elements present in organizations, especially humans and machines, through the Internet of Things (IoT), creating a digital virtual copy of the real world.

It is is known as the **4th Industrial Revolution,** the next phase of digitisation in manufacturing. This new revolution is occuring because of a greater, more intensive and extensive use of computers, automation and electronic data communication, which can dramatically improve not only factories, but also all types of organizations.

Source : https://edinn.com

What is efficiency?

To fully understand how the edinn platform **helps to increase total efficiency**, we need to define 2 concepts: Process and Efficiency.

EFFICIENCY is defined as:

Efficiency = Results / Consumption

Where:

- **Results:** are the valid **outputs** of the process, e.g: *good finished products, results of tasks...*
- **Consumption:** are the total **inputs** of the process, e.g: *raw materials, energy...*

We consider these **types of efficiency**:

- **TOTAL EFFICIENCY:** when the total, results and consumption are analyzed. For this, the **OPCE** standard ratio is used.
- **PRODUCTION EFFICIENCY:** when only the results component is analyzed, measuring them in terms of Availability, Speed and Quality. For this, the **OEE** standard ratio is used.
- **CONSUMPTION EFFICIENCY:** when only the consumption component is analyzed. For this, the **OCE** standard ratio is used.
- **ENERGY EFFICIENCY:** when we only analyze energy consumption.

WHY IT IS SO IMPORTANT:

Efficiency can be considered for processes, groups of processes... or an entire organization. In fact, the economical results of a company, its yearly profits, are nothing different than pure Efficiency.

If a process is not performing at 100% efficiency, we say it has **losses**.

As you can see, **improving** efficiency means directly improving **profits**, because you are producing **more good results** and you are consuming **less resources**: *money, time, energy, materiales, etc*.

This has enormous direct benefits, as generates:

- Higher **economical benefits.**
- **Less impact** on the environment.

But there is also an additional interesting effect. In the analysis of consumption efficiency, one can detect steps or procedures that generate excessive consumption, which a deeper analysis may determine that the process should be adjusted. This will improve production

efficiency which again will imply an improvement in consumption efficiency.

Therefore, although it may not be a direct conclusion of improving efficiency, in general, it has been confirmed that more efficient organizations:

- Produce with less time to market
- Produce better results with higher quality
- Are more profitable.

What can affect efficiency?

Efficiency can only be affected by **4 factors**:

- **AVAILABILITY**: the process stops and it is not working due to **failures** or **idle** periods OR there were **no results** during the time that could have been used for production.
- **SPEED**: results are not obtained in the **expected time.**
- **QUALITY**: results are not good enough, so they are **discarded** or **reworked**.
- **CONSUMPTION:** more energy or resources than necessary are used to produce the results.

What is OPCE?

The ratio created by edinn to measure **Total Efficiency** is called OPCE (*Overall Production and Consumption Effectiveness*).

OPCE measures both the **Productive** and the **Consumption efficiency**. It is useful to analyze the 2 factors (OEE and OCE) in one sight. It is defined as:

$$OPCE = OEE * OCE$$

Meaning:

- 50% OPCE means that the process could double its total efficiency by:
  - Producing the same but consuming half.
  - Consuming the same but doubling production.

- A combination of the previous.

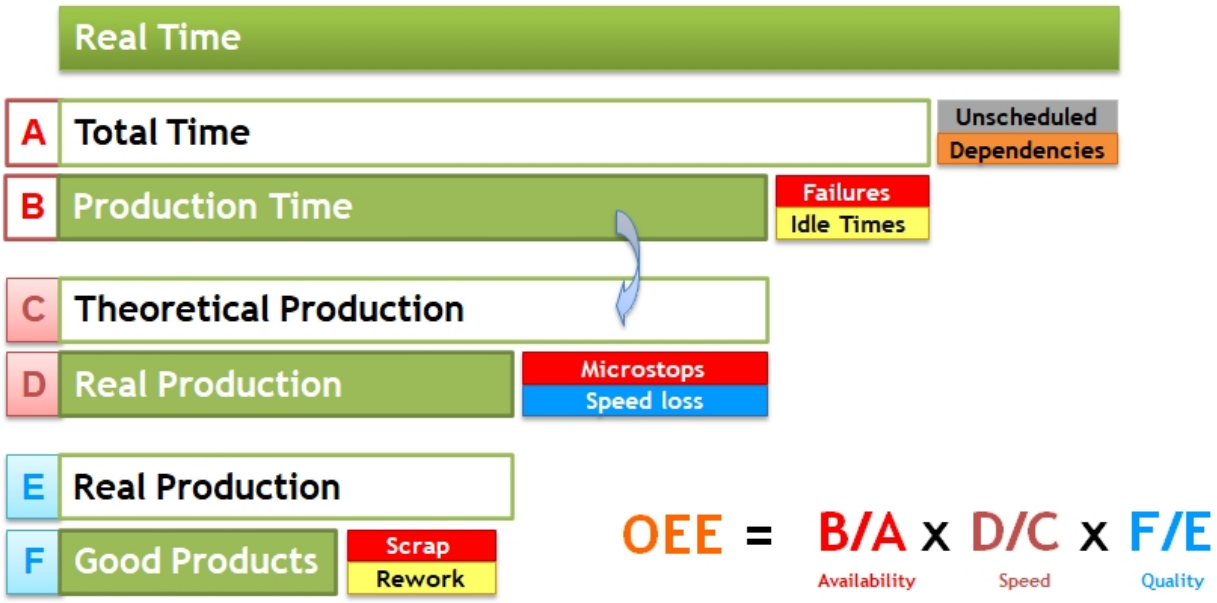
- 100% OPCE means that the process can neither consume less nor produce more.

What is OEE?

The international standard ratio to measure **Production Efficiency** is called OEE (*Overall Equipment Effectiveness*).

> **NOTE**: With regard to Work Management, for information on how the **availability** and **speed** ratios are calculated in a way more appropriate for processes which are persons, please view 'operation time' in processes configuration.

OEE CALCULATION:



The advantage of the OEE, compared to other ratios, is that it measures, in one ratio, all fundamental parameters regarding **productive losses**: Availability, Speed and Quality.

These three ratios compose the OEE and, by analyzing them, it is possible to know what **losses,** until 100%, are caused by:

- **AVAILABILITY**: the process stops and it is not working due to **failures** or **idle** periods or there were no results during the time that could have been used for production.

- **SPEED**: results are not obtained in the **expected time**.

- **QUALITY**: results are not good enough, so they are **discarded** or **reworked**.

AVAILABILITY CALCULATION:

$$Availability = B / A$$

Where:

- **A. Available time =** total time that the process **could have been producing** = [natural time] - [not scheduled time to produce] (such as holidays, lunch, programmed maintenance tasks, weekends...).
- **B. Running time =** time the process **has being producing** = [available time] - [failures and idle times]

SPEED CALCULATION:

$$Speed = D/ C$$

Where:

- **C. Theoretical production =** number of products that **could have been produced** = [running time] * [nominal production speed].
- **D. Real production =** number of products **produced** = [theoretical production] - [microstops and speed loss]

QUALITY CALCULATION:

$$Quality = F / E$$

Where:

- **E. Real production =** total number of results produced.
- **F. Good production =** total number of **good** results produced.

What is PE?

The **Productive Efficiency** (PE), is a OEE variant, which does not consider idle statuses.

This ratio is used to compare the number of failures and preventive process maintenance, without considering idle times because these can be caused by result changes, cleaning, etc.; and this is not always the responsibility of the operators. Therefore, for some companies, PE is a better ratio to evaluate the performance of the operators.

The PE ratio is a quick approximation to measure performance of personnel better than

OEE, but for a ratio which is objective and fair for this purpose, please refer to the Activity Ratio.

The most similar ratio in the ISO 22400 standard would be the Net Equipment Effectiveness Index (NEE), although the latter only ignores preparation waits (setup).

Be careful not to confuse this ratio with the concept of Production Efficiency.

> **NOTE:** for more information about idle time, please see: What is a status?

What is OCE?

The ratio created by edinn to measure **Consumption Efficiency** is called OCE (*Overall Consumption Effectiveness*).

OCE measures the **consumption efficiency** of all resources, including **energy efficiency.** It is defined as:

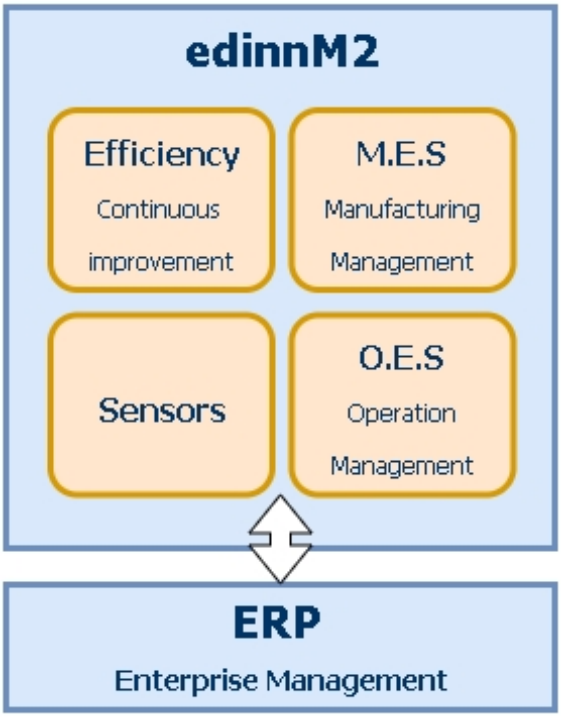
OCE = (Maximum Cost - Real Cost) / (Maximum Cost - Minimum Cost)

Meaning:

- 50% OCE means that the process could double its consumption efficiency by producing the same but consuming half.
- 100% OCE means that the process cannot consume less.

Manufacturing Execution System (MES)

A **MES** (*Manufacturing Execution System*) is a system that manages and monitors **work-in-process**, including both manual and automated tasks, generating **production reports**.



The **edinn® M2 system** operates in any sector and constitutes a complete MES and OES edinn® M2 system.

Edinn is a **Premium Solution Provider** member of **MESA International** in order to guarantee that its MES solution complies with the most demanding **international standards**.

The **integration** of edinn® M2 with any ERP is realized through the integration module using **international ISA-95 standard**, **XML** and **web services**. With this module, you can use your own ERP and change in the future, protecting your investment.

Functionality of edinn® M2 MES and OES are:

- View and Manage the Schedule of Production
- Stock and Resource Management
- Production Data Collection
- Maintenance Scheduler
- Quality Assurance
- Time and Activity Managment
- Access Control
- Productive and Energy Efficiency Improvement

Time and Activity Management

Time and activity management is convenient to:

- Eliminate all inefficiencies and make the organization more profitable and competitive.
- Reduce delivery times.
- Allow incentive payments that motivate employees and maximize performance.
- Production control.

The edinn® M2 system has exclusive functionality on this area.

Main functionality of edinn® M2´s time and activity management is:

- The system is able to determine if a user has been working and during what time, based on production results.
- Easy to use, no need for experience in computer science.
- Resource (person) presence and performance control.
- Activity ratio report of each resource (person), independently of processes, products or tasks performed.
- Possibility of production breaks and to carry out various tasks at the same time, including various processes at the same time.
- Control and possible prohibition of presence of a user in several processes.
- Possible expulsion of a user of a process.
- Management of working calendar.
- Manual and automatic time data entry.
- Connection with salary and incentive software systems.

Production Data Collection

The edinn platform allows fully automated as well as manual and intelligent data collection with real-time data processing. It is 100% configurable, precise and can handle a multitude of systems (PLC´s and manual processes) simultaneously.

Functionality of **edinn® M2´s production data collection** is:

- Available for any type of process: through PLC, manually or through computer

vision.

- Acquisition is 100% automatic and can be configured for all production data.
- Automatic detection of time usages (stop, production, microstop, dependence, etc.) of processes.
- Capturing of context signals (pressure, temperature, etc.) to detect problems and continual improvement.
- Capturing of reasons of stops, incidences, etc.
- Compilation of data to calculate international standard ratios (OEE, MTBF, etc.)
- Provides information for Maintenance, Proceedings and SPC tools of edinn® M2 as well as for ERP and other systems.

Lean

Lean is a management philosophy focused on he reduction of the 7 types of "waste". Let us see these types of wastes and how edinn® M2 uncovers and helps to reduce them:

1. **Overproduction**: edinn® M2 includes view and manage of the scheduling functionality which allows to see in real time overproduction.
2. **Time of delay**: edinn® M2 includes a complete status and time usages analysis.
3. **Transport**: edinn® M2 includes a consumption module and a complete status analysis. Both will uncover transportation times.
4. **Excess of processing**: edinn® M2 includes a complete status and time usages analysis.
5. **Inventory**: edinn® M2 includes Stock and traceability functionality as part of its included MES.
6. **Movement**: edinn® M2 includes a complete status and time usages analysis.
7. **Defects in manufactured products**: edinn® M2 includes quality improvement tools.

Eliminating waste, quality improves and production time and costs will be reduced.

Lean is basically obtaining the correct things at the correct place, at the correct moment, the correct amount, reducing waste, being flexible and open to change.

One of the key principles of Lean is Continuous improvement - reduction of costs, quality improvement, increase of productivity and sharing information.

For more information: please consult the wikipedia.

Kaizen

Kaizen (Japanese for "improvement" or "change for the better") refers to philosophy or practices that focus upon continuous improvement of processes in manufacturing, engineering, supporting business processes, and management. When used in the business sense and applied to the workplace, *Kaizen* refers to activities that continually improve all functions, and involves all employees from the CEO to the assembly line workers. By improving standardized activities and processes, kaizen aims to eliminate waste.

edinn® M2 is completely focused on continuos improvement, so it is considered an essential tool for *Kaizen*. *Kaizen* is based on continuous improvement, continually, day to day. edinn® M2 provides you with a 100% detail of losses, in real time, day to day.

For more information: please consult the wikipedia.

5'S

"5S" is the name of a workplace organization methodology that uses a list of five Japanese words which are seiri, seiton, seiso, seiketsu and shitsuke. Translated into English, they all start with the letter S: sorting, straightening, systematic cleaning, standardizing and sustaining. The objective is to obtain better organized, sorted and clean workplaces in order to improve productivity and an improved workplace.

edinn® M2 is the ideal tool to implement and maintain the 5S methodology, because it shows 100% details of the causes of productive losses. This makes it easy to check which tasks are not performed correctly (order, cleaning, etc.). Additionally, edinn® M2´s autocontrol controls if the procedures needed to maintain these disciplines are being followed.

For more information: you can visit this external link to wikipedia.

Quality (SPC)

There are **2 basic approaches** to quality from the system perspective:

- Qualitative approach: following procedures in case of a quality problem
- Quantitative approach: performing Statistical Process Control (SPC).

For the qualitative approach, edinn® M2 offers this procedure:

1. A quality issue can be detected by a quantitative analysis (SPC) or by the introduction of a user of a bad result in the form Results.
2. The user can inform to another user by using the "Comments" functionality, double clicking on any data grid. The user can select the type of user defined event that will allow further analysis.
3. The destination user can control what quality issues he has attended, by using the "View" functionality available at the Registry. He can also filter the search by selecting the user defined event.
4. The destination user can respond to the quality issue using the Registry form.
5. The quality supervisor can check, in the Events report, by selecting the type of the user defined event corresponding to the quality issue, if all quality issues have been addressed.

For the quantitative approach, SPC is edinn® M2´s tool to help to improve Quality.

SPC stands for Statistical Process Control which is a method to improve quality within a production process. Quality losses in a production process have a very high cost, because these products need to be reworked, discarded; or in the worst case, they reach the client damaging the organization's image and profits.

In order to become a defectuous product, it needs to not fulfil a quality variable. SPC is based on the measurement of these variables that indicate a product's quality, not only at the end, when it is too late, but periodically, during the production process, in order to know if the process is controlled within control limits and within limits specified by the client (in edinn® M2 the information is presented in green and red colour).

CPK is a ratio that indicates if the process is centred related to control limits and client specifications. Mathematically, SPC methodology is very complex. You need to consult bibliography (see below) if you want to know the exact formulas associated to SPC. edinn® M2 contains all SPC formulas and makes all calculations so the user does not have to do this. In one of the SPC reports it is possible to obtain automatically and in real time the main ratios (like CPK). Following edinn® M2´s philosophy, making SPC calculations with the system is very easy. Periodically, edinn® M2 will measure quality variables or will ask the user to do so by making blink the SPC button. If it does not receive SPC data, a justification is asked for why they are not taking data. Based on SPC data, automatically and in real time, a control graphic and histogram are generated.

For those values out of control limits, the user is asked to justify the steps he followed to "retake" the process. The SPC graphic is interactive, in order to let the user click on each point and see when measurements were taken, what justification was given, etc.

SPC Context

SPC context data refers to data that, although is not directly taken as SPC measures, it is

useful to understand them. Examples of SPC context data are: temperature, pressure, etc. Data that the organization does not measure for SPC purposes, but that the organization needs to have registered in order to perform analysis to understand the behaviour of the SPC data.

For more information on SPC, please follow the external links:

- Statistical Process Control
- CPK

What is MTBF and MTTR?

- **MTBF** (Medium Time Between Failures): is the medium time between each failure within a process, or in other words, the frequency of each failure. Tell us the most frequent failures within a process.
- **MTTR** (Medium Time To Repair): is the medium time until having repaired the failure. Tells us which are the most serious failures.

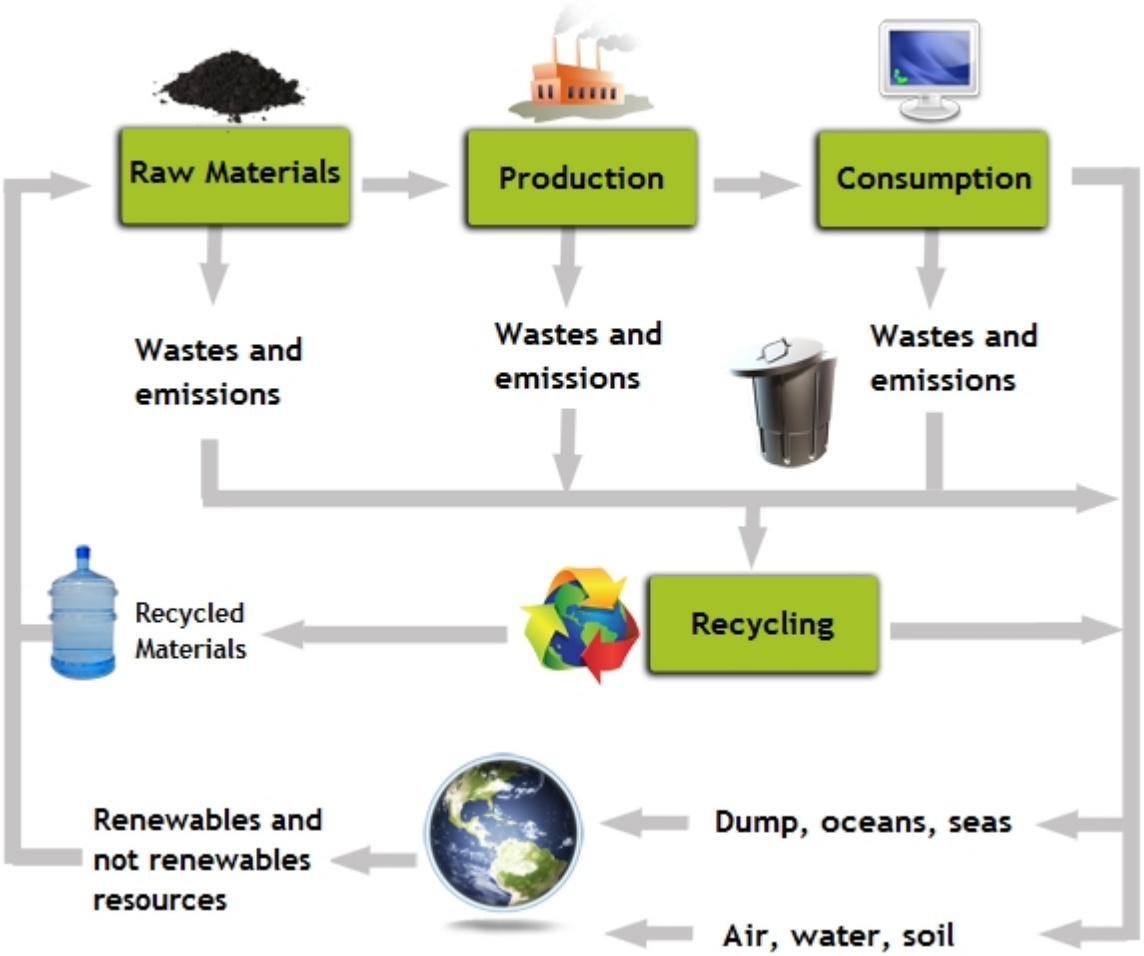
MTBF and MTTR are generally measured in minutes.

The edinn platform calculates the generic MTBF table (for all failures of each process), and the MTBF table depending on the current active result (shown as MTBFP). It is possible to configure the server for the period for which to calculate.

Carbon footprint

The edinn platform is able to determine the carbon footprint with high precision. It calculates all parameters that are necesarry for the carbon footprint: statuses of processes, real results obtained, raw materials consumed, energy consumed, personnel, stock consumption, etc.

To do this you just have to put the equivalent CO2 generated where this field is available, generally in consumption configuration, and then you will obtain this data in the consumption reports.



For more information: on how to correctly determine the carbon footprint, please refer to UNE-EN 16001 or ISO 50001.

What is OPC?

OPC is a **standard to read and write data from and to machines**, although it can also be used with any other device. For more information please visit the OPC FOUNDATION.

Why do you need the OPC standard for? and why do you need OPC Servers for? Because whenever it is possible, you will not have to type any data manually, since the edinn system is able to obtain it **automaticallly**:

- Therefore, type the necessary information manually, or configure the edinn system to obtain it automatically.
- The monitoring variables can be configured visually.
- Monitoring can be performed by "pull" (the system will obtain the data periodically) or "push" (the system will receive the data every time that the data

changes), which guarantees that no data is lost.

- The monitoring subsystem is autonomous, so it does not depend on the rest of the subsystems.
- The system automatically monitors persons, machines and other systems, whether software or hardware. For the monitoring, standards like ISA-95 and OPC UA/DA, as well as non-standard methods, can be used.

eCloud

The term eCloud refers to the edinn services for using the systems through internet, known as the "cloud".

This services mainly allow the user to:

- Use the system without the need of having local servers at his sites.
- Forget about backups, server or hardware maintenances, etc.
- Easily adjust the power of his edinn system to his needs, as the cloud can dedicate more resources to him.

Please see: new features of edinn.

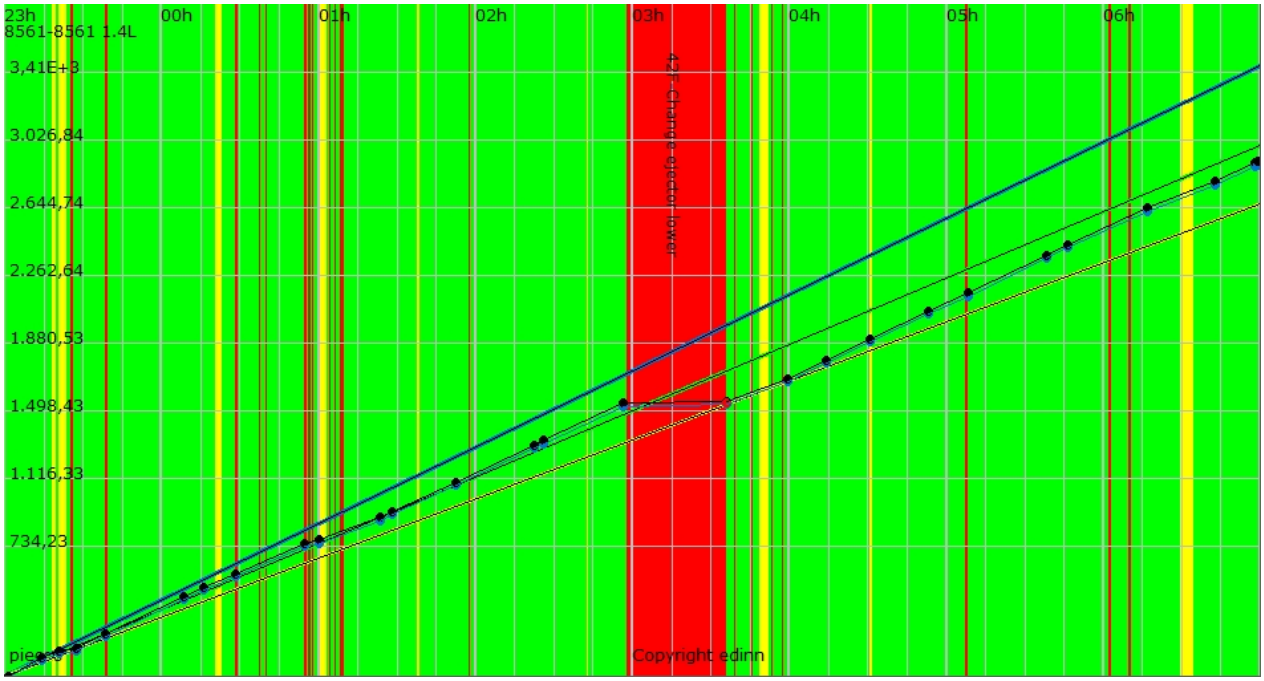
What is the "Path"?

The "Path" is a concept created and registered internationally by edinn®.

The idea behind it is that organizations need to see their correct "Path" of production to have the maximum possible information in order to quickly, easily and continuously improve.

There are different **types of paths.**

PRODUCTION PATH:

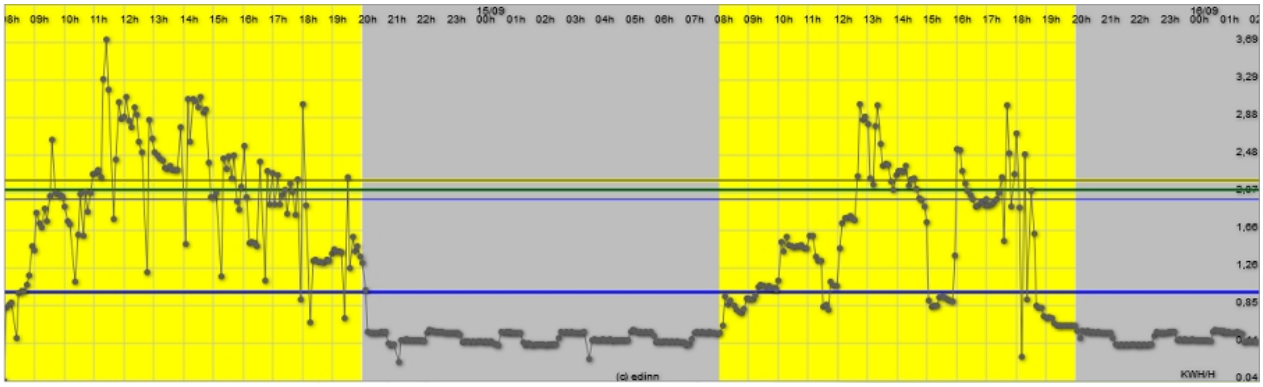


This "path" shows:

- **X axis:** shows the time passed.
- **Y axis:** shows the quantity of good results obtained.
- **Color areas:** are the statuses that the process has passed. Please refer to the color codes of statuses.
- **Blue dots:** it represents the production without taking into account speed losses (only the stops), which can be identified by comparing the black dots with the blue dots.
- **Black dots:** are real quantity of good results obtained. On the line that they describe, also red dots can be shown. This red dots indicate bad quality results obtained. Represents the productions taking into account the stops and the speed losses.
- **Blue line:** is the 100% production efficiency.
- **Green line** (above the orange line): this is the green target. If black dots (good results) are above this line, the process will be shown green.
- **Orange line** (at the bottom): this is the yellow target. If black dots (good results) are below this line, the process will be shown red.
- **Path:** is defined as the space between the upper blue line and the yellow/orange lower line. The work, or the production (good results) must follow the path defined by this upper and lower lines.

NOTE: the objectives are established by each client.

CONSUMPTION PATH:



This "path" shows:

- **X axis:** shows the time passed.
- **Y axis:** shows the quantity of consumption (kWh in the image).
- **Color areas:** are the statuses that the process has passed. Please refer to the color codes of statuses.
- **Grey dots:** are the real quantity of consumptions.
- **Blue lower line**: is the 100% consumption efficiency.
- **Yellow line** (at the top): this is the yellow target. If grey dots are above this line, the process will be shown red.
- **Green line** (below the yellow line): this is the green target. If black dots (good results) are below this line, the process will be shown green.
- **Blue upper line** (below the green line): that is the expected maximum of consumption.
- **Path:** is defined as the space between the upper green line and the lower blue line.

What is a shift?

Shifts are periods of time during which different groups of workers take up their posts.

Please take into account that:

- All the 24 hours of a day must belong to a shift.
- Shifts can not overlap.
- One shift must end at the same time that the next starts.

What is a process?

To fully understand how the edinn platform helps to increase total efficiency, we need to define 2 concepts: Process and Efficiency.

PROCESS is defined as a set of **activities** (process) performed by a set of **resources** (inputs) in order to obtain a **set of results** (outputs)

This definition allows to consider many things as a process: *an employee, a machine, any combination of persons and machines...* Some examples:

- A **solar plant** can be considered a process. But also each individual solar panel of the solar plant is also a process: they take the sun light as input, and generate electricity as output.
- A **house** or **building**: takes inputs (*electricity, water, etc.*) and generates outputs (*heat, light, etc.*).
- A **power generator windmill**: takes the wind as input and generates electricity as output.

As you can see, we define the limits of the process and the processes themselves based upon our interests. If we want a deeper analysis, we define lots of little processes, so that we have a detailed analysis of where the losses are. If that deeper analysis is not profitable, because it is enough to have a wider view, then we define processes as groups of processes, for example areas or groups of persons and/or machines.

What is an area?

An **AREA** is a group of processes, and therefore, for Work Management it is also a project. In edinn® M2, an Area is just a way of grouping diferent processes. Areas are fully virtual, so you can change them to match your needs.

Examples of areas are:

- **In a factory**: production lines.
- **A project**.
- **In any organization**: departments.
- **In a solar plant**: panel streams.

What is an order or task?

Before defining what an order or task is (sometimes expressed in edinn as **PN** [Production Number] or **Order Id**), we need to define what a client order is.

A **client order** is a group of **services and goods** that a client has ordered to our organization.

Once the client order must be produced to generate real services or goods, it needs to be converted into one or more different work **orders** or **tasks**. For the realization of the order or task, it must be assigned to one specific process.

Therefore, in edinn, an **order** or **task** is a **number of results, of a unique type,** that a process must generate in order to serve partially or totally **a client order**.

The edinn platform manages work **orders** or **tasks** with the Scheduling module but does not allow to manage **client orders**. This is typically performed by an **Enterprise Resource Planning (ERP)** system.

What is a result?

Results are the different elements produced by a process. Depending on the sector, they can also be known as products, parts or references, among other possible names. But in edinn they are known as results as, inside the same organization, some processes can produce services (cleaning, maintenance, etc.) and, because of that, the results word allows users to communicate better because is a common word for all processes, instead of products, parts or references.

In edinn, there are 3 types (shown in their usual colours):

- **Production**: all the results, including those which later can be discarded or reworked.
- **Discarded** or **scrap**: results which are discarded, and therefore 1 was counted as total and 1 was counted as discarded.
- **Rework**: results which are reprocessed, and therefore 2 were counted as totals and 1 was counted as reworked.

What is a status?

Status are the different time usages or statuses that a process passes during time. There are **5 types**:

- **PRODUCTION (shown in GREEN)**: the process is working to generate a result. This type of status **affects** the production efficiency of the process.
- **IDLE (shown in YELLOW)**: the process is waiting. This type of status **affects** the production efficiency of the process. Examples of idle statuses are: 'changing product' or 'cleaning'.
- **DEPENDENCE (shown in ORANGE)**: the process is waiting for another process. This type of status **does not affect** the production efficiency of the process. In other words, one process could be always in a dependence status, and have a 100% production efficiency. Examples of dependence are: 'waiting for results from another process' or 'there is not stock available to produce'.
- **FAILURE (shown in RED)**: the process is stopped because of a failure in the process. The process can not continue working unless some action is taken. This type of status **affects** the production efficiency of the process. Example of failure is: 'process is broken'.
- **NOT SCHEDULED TO PRODUCE (or unscheduled, shown in GREY)**: the process is not scheduled to work. This type of status **does not affect** the production efficiency of the process. In other words, one process could be always in a 'unscheduled' status, and have a 100% production efficiency. Examples of unscheduled status are: 'week end' or 'end of day'.

What is the activity ratio?

The **Activity Ratio** is a percentage ratio, **created by edinn**, which measures performance of a person who operates machinery. Its use is necessary because it is not correct to use OEE to measure performance of personnel operating machinery.

Let us see two examples to understand this better:

1. Let us imagine one person who produces very long series, with very few changes, using modern, fast and powerful machinery, which produce with high quality and which requires very little maintenance.
2. Let us imagine, on the other side, another person who produces very short series, with a lot of changes, using old, slow and not powerful machinery, which produce with low quality and which requires a lot of maintenance.

OEE is appropriate to measure performance of the modern and old machinery, but it is clearly not to measure performance of the persons operating in the case 1 and 2, as both persons are neither responsible of the machinery nor of length of the orders assigned to them.

Because of what was previously explained, it is necessary the **Activity Ratio**, which measures personnel performance with maximum independence of the context.

How is this solved in edinn?

By configuring a percentage of activity for every status of the process. For those statuses where the personnel is not responsible of, a zero percentage of activity should be configured.

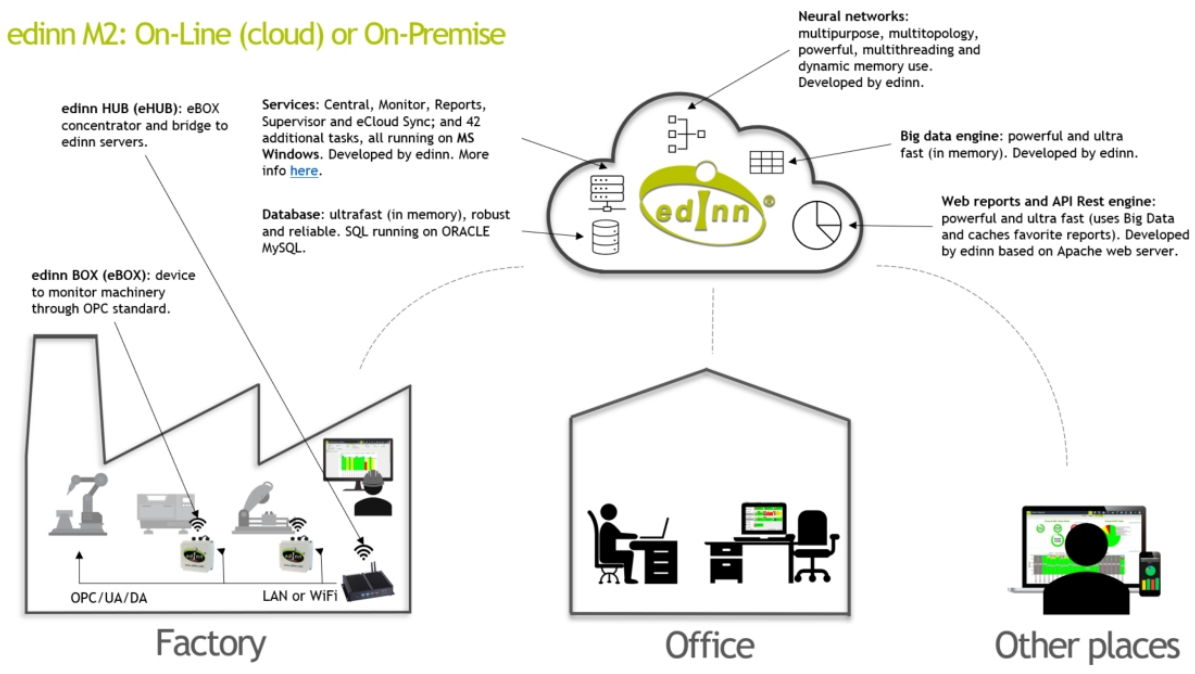
When a status has a maximum time, and after that time it is not responsibility of the personnel (for example with a change of reference), then it should be configured a zero percentage for the status which has maximum time and is not responsibility of the personnel, and a percentage different from zero for the status of excess.

Cuando un estado tiene un tiempo máximo, y a partir de ahí ya es responsabilidad del personal, por ejemplo un estado de cambio de referencia; entonces se configura porcentaje cero para el estado que tiene tiempo máximo y no es responsabilidad del personal, y un porcentaje distinto de cero para el exceso.

What is the edinn platform?

Please see the introduction for a definition of the edinn platform.

The following diagram shows the architecture of the edinn platform:



The edinn platform achieves Digital Transformation by focusing in two well known areas:

- Industry 4.0.
- Work Management.

Which are materialized in the following applications or modules that work in a fully

integrated way.

Industry 4.0

Industry 4.0 opens a great opportunity to improve all aspects of an organization, but unfortunately, it can also become a motivation to start projects which will not be profitable enough, or even not profitable at all. Today, we know that the difference is:

- **The people**: how users adapt, embrace change for better, learn, use and take the most out of the tools, is key for success.
- **The right technology platform**: in the hundreds of Industry 4.0 projects developed by our partners or us, every time a problem and a solution was found, it was incorporated into the edinn platform, so that users would not suffer from these problems and would obtain benefits faster.
- **Technology that is easy to use**: every time a user did not know how to perform something, we asked ourselves how we could improve the platform so that users would easily know how to proceed.
- **The right methodology:** the edinn platform was designed so that users are guided to incorporate functionality step by step, in the phases of a proven methodology, and allowing them to test it, at the minimum cost, before incorporating it.

Therefore, the platform is designed so that users naturally follow these phases:

1. **Interoperability**
   The first step in Industry 4.0 projects is to interconnect all the elements which are present in an organization, for example: machines, persons, systems (like ERP and MES) and sensors.
   Interoperability refers to connecting all items in the organization to be able to operate all of them remotely. This is also known as the **Internet of Things (IoT)**, where billions of devices have unique addresses and can transmit data to the cloud for processing, and be managed and controlled via applications. Thanks to increasingly miniaturized computers, affordable sensors, ubiquitous networking, and the increasing availability of “smart” devices around the world, example IoT applications will range from smart manufacturing to fitness-trackers and from smart homes to smart farming.
2. **Digitalization**
   The second step in Industry 4.0 projects is to digitize all the elements present in an organization to creat a virtual copy of it. This virtual copy of the organization will be useful to:
   - Navigate through all the elements and their associated data and information.
   - Interact with all the elements.

3. **Help to Improve**
   The third step in Industry 4.0 projects is to incorporate different tools which help to improve the organization. These tools range from measuring KPIs (Key Point Indicator) and ratios, controlling quality through Statistical Process Control (SPC), Maintenance, and up to whatever the need of the organization.

4. **Advanced exploitation**
   Once the previous phases are covered, at least partially, then more advanced applications and tools can be used or developed, like Big Data or Machine Learning.

In the experience based on hundreds of Industry 4.0 projects, following these phases is key for the success.

Profitability in edinn® M2 is mainly obtained because of these 2 reasons:

- **Improvement of production and consumption ratios**: this is achieved because both operators and management personnel achieve the reduction of losses and the increase efficiency, in some processes almost unconsciously. Although it can not be guaranteed in all cases, it has been repeatedly reported a minimum of 5% to 10% improvement in total efficiency in the first year of using the system.
- **Elimination of the time used to manually gather production data**, including calculations, reports, distribution, etc.

Edinn® M2 typically reaches break-even within one year. At edinn we can also provide you with tools by which you can make your personal calculations about edinn® M2's profitability.

Work Management

Work Management constitutes the control, management and improvement of any work, of whatever type, both intellectual and physical.

Work has been always very difficult to control, manage and optimize; whether made by people in the same space or, even more, if made remotely.

Any work is based on performing tasks. These tasks can be classified according to whether they are:

- **Recurrent**: they are repeated every certain time, for example, the administrative tasks.
- **Non recurrent**: once done it is not probable that they will be done again, for example, tasks inside a project.

To better know how edinn approaches this area, please see:

- Information about the Work Management module.

- How Work Management is done?

Work Management affects multiple aspects of the platform. See, for example, how it affects the OEE calculation.

Security

Security is a very important aspect of the edinn platform. The following security measures are taken:

- All passwords are stored and checked encrypted.
- Passwords can be forced to have a minimum of complexity: a minimum of 7 characters and at least 1 capital letter, 1 lower letter and 1 number.
- All data can travel encrypted between the user and the edinn system: please see https://ecloud.edinn.com
- The edinn software core executable files are encrypted.
- The edinn web software is periodically tested by external security experts and security patches are applied after assessment.
- All edinn cloud servers are provided by Amazon Web Services (AWS).
- [illegible]
- The edinn platform can be updated in a matter of minutes in all installations, what allows to keep operative systems up to date with all the latest security patches.

Modules

The edinn platform is modular, which means that functionalities can be added by activating modules.

In this guide we will briefly explain the available modules.

Production Control (OEE)

The **PRODUCTION CONTROL (OEE)** module helps the organizations to:

- Really know **what happens** at production.

- **Improve continuously** based on **real data**.
- **Trace** and **have information** on context variables to **understand** problems.
- **Increase the profitability of the operations:** our users reported increases of **10% in its efficiency (OEE)** in less than **one year** of using the system, with 100% detail what causes that increase, because it helps them to see in past and real time, and know what people and machines could improve its performance and efficiency.
- **Access at any time and place:** you will have the information in any device: PC, tablet or smartphone. Also you can configure Email notifications with most important alerts.
- **For any sector and size of company:** not matter in what sector you or your company works, or if your work is fully manual or automatic, the system edinn® M2 is already operating with success in variety of different sectors.

MODULE FUNCTIONALITY:

- **Simple, visual and interactive:** know in real and past time, with visual and graphic performance alerts, what **people** and **machines** are working on what **tasks** and **orders**, with what **results** and **performances,** which will motivate them to increase their performance and reduce their consumption. See in a quick way, through colors of **red**, **yellow** and **green**, when a machine or person is not making results and consumption **to be profitable**, compared with the **objectives** of the organization.
- **KPI's:** automatic calculation and display of **key performance indicators**: productivity (**OEE**), total efficiency (**OPCE**) and consumption (**OCE**): availability (**D**), performance (**V**) and quality (**Q**) and Mean Time Between Failures (**MTBF**), among others.
- **Detailed:** supervisors and management can in real and past time see with certainty of 100% their inefficiencies: what separates organization from productivity and total efficiency.
- **Notifications:** if you want, you can receive emails when certain events happened: when machine, person or group of them is not being profitable, when a user makes a comment for improve...In addition, users can receive emails telling them periodically which tasks performance has been achieved or not, asking for the latest justification.
- **Internal messaging system:** connected to email, for the **incidents** management and **suggestion** for improvements.
- **Interactive:** request a **justification** for the non-achievement of the objectives.
- **Create new fields of information associated** with the results and orders.
- **Zero papers:** access to electronic documentation or other applications linked to

any element of the system: *machines, incidents, maintenance and procedures, quality, etc.*

Scheduling

The **SCHEDULING** module helps the organizations to:

- **Automatically or manually Schedule, Manage** and **Optimize** tasks and work orders with an Advanced Planning and Scheduling algorithm.
- **Optimize delivery times and production:** check the tasks that are on time, those that will be delayed if you do not act, and those that are delayed. In addition, you will have estimated completion times based on the real performance (OEE) of your processes.
- Know the **work load** of each process and resource and detect **bottlenecks**.
- **Connect with your ERP:** leverage all your actual tools and capabilities for planning, and take advantage of exclusive edinn features for management, control and optimization.

MODULE FUNCTIONALITY:

- **Automatically or manually Schedule** tasks and orders considering different criteria with an Advanced Planning and Scheduling algorithm.
- **Know the need of stock and resources** according to the plan and things like replacement and expiration times and minimum stocks**.**
- **See the histogram of the needs of stock and resources in real time**, while you manually re-schedule**.**
- **Display screen** of the plan: employees can choose the **next task** from the schedule and will be able to see their **progress** compared to **production targets**.
- **Notifications:** if you want, you can receive a periodical email or notification and see, in **real** and **historical** time tasks that are on time, those that will be delayed if you do not act, and those that are delayed.
- Indicate the **resources** and **materials** used, as well as the **status** in which each of their **task and orders** are. This is useful to monitoring **stock** and **material levels**.
- **Real time connection to ERP:** synchronize all the planning information with other systems (*ERPs such as SAP, Peoplesoft, etc.*) and/or machinery through standards such as **ISA-95** and **OPC**.
- **Create new fields of associated information** with the results and orders.
- **Zero papers:** access to electronic documentation or other applications linked to

any element of the system: *planning, machines, incidents, maintenance and procedures, quality, etc.*

Work Management

The **WORK MANAGEMENT** module helps the organizations to:

- **Increase performance and motivation**: with edinn, we measure from the time worked, through the results obtained, to the fulfilment of commitment dates. In an easy and useful way, almost without noticing it.
- **Higher performance in telework, in real time**: thanks to greater visibility among the members of the organisation and in real time.
- **Results-oriented management**: say goodbye to annoying report sheets or timetables.
- **Greater confidence for supervisors**: the task and its results, the progress, the commitment date, the status and the workload are all seen in real time and in aggregate reports.

MODULE FUNCTIONALITY:

- **Create a process for each user** and configure the required daily operation time.
- **Add a photo of the person** for each process
- **Configure the time statuses** in order to generate the work calendar.
- Make the working team:
  - **Create the tasks and ask for them to be scheduled so that the supervisors receive an automatic notification** to check the times, finish dates, and characteristics, after which the supervisors can schedule the tasks so the workers may begin working on them.
  - **Start the tasks and insert the advancement of results for them**.
  - **Justify the not productive states** when the system requires it due to not having input a status, or failing to input sufficient results.
  - **Prefinalize the tasks** when they are finished, which will generate and send an automatic notification to the supervisors so they may revise the quality of the results of the given task.
  - **If you are a supervisor**:
    - **Insert low quality results**, when they have been detected in the

tests before finalizing a task.

- **Finalize the tasks** after reviewing the quality of results; or, return the tasks to the scheduled state so the workers may continue working on them until reaching the desired results.

- Visualize or receive **alerts**:
  - **Projects or tasks that are going to be delayed or have already been delayed**.
  - Personnel which should:
    - **Report results**.
    - **Justify their time spent working**.
    - **Perform periodic tasks or procedures**.
    - Other.
- Obtain **reports**:
  - **Indicators of personnel performance, including an adapted OEE**:
    - **Availability**: percentage of completion of their designated work hours.
    - **Velocity**: percentage of completion of tasks within their designated deadline.
    - **Quality**: percentage of high quality results compared to all given results.
  - **Projects and pending tasks**, to whom they are assigned, anticipated deadline and completion dates, the state of each one, percentage of advancement, etc.
  - **Total time dedicated per project, task, and state**.
  - **Summary of monthly hours** worked.
  - **Results** that have been generated for a given task.

Procedures and Maintenance Management (CSSM)

The **PROCEDURES AND MAINTENANCE MANAGEMENT (CSSM)** module helps the organizations to:

- **Optimize Maintenance** and related resources, reducing required times, allow to

follow Procedures to improve Availability, Speed, Quality and Consumption Efficiency.

- **Reduce maintenance costs:** our users in already highly optimized industry reported reduction in costs of maintenance up to 40%.
- In order to better understand this tool, we will give an **easy example**. Any of us likes to have a process we often use, a car, to be always available. Therefore, we perform an autocontrol based on:
  - Every certain kilometres (15.000, 30.000) we take it to the garage to have it checked.
  - In these checks procedures are followed.

MODULE FUNCTIONALITY:

- **Optimize maintentance by programming tasks and procedures:** schedule them to be periodically launched or at certain events, such as: production of a certain number of units, the process is in a certain status, a certain time has passed since the last maintenance, failure because of lack of greasing, on hold for change of mold, etc.
- **Prioritize** the tasks and procedures distinguishing those that are **critical,** and **set tolerances** to the tasks so the user can know them with advance and make available certain time margin.
- **Link** tasks and/or procedures to status of process (breakdowns, delays, etc.) to define and control how your organization manages the different situations that can occur.
- **Visual control:** by colours for oprerators and maintenance supervisors.
- Make **forecast** for **future periods** and then **simulate it** to know what maintenance tasks and procedures that can be carried out.
- Get **aggregate reports** and **details** to determine the level of compliance with procedures and maintenance tasks, as well as efficiency in its realization. We get to know the inter-process times in order to resolve failures, poduct changes, etc.
- Requires user **re- authentication** to ensure that certain supervisors guarantee that the tasks and/or procedures are made correctly.
- Activate the **workflow approval** of tasks and procedures in which supervisor have to approve the start of tasks or critical procedures.
- **Personalized:** we can specify those fields we want the user to be informed of while performing a task or in case of certain events.
- **Additional information:** add pictures, documents... about how to perform maintenance: tools, times, results, etc.

- **Notifications:** if you want, you can receive an email:
  - When a **machine stops** more than some time, so you can take advantage and perform preventive **maintenance,** which will allow you to gain a very valuable time and reduce costs.
  - When tasks or important procedures are close to become **Not Done**.
- **Create new fields of information associated** with the results and orders.
- **Zero papers:** access to electronic documentation or other applications linked to any element of the system: *machines, incidents, maintenance and procedures, quality, etc*.

Energy efficiency

The **ENERGY EFFICIENCY** module helps the organizations to:

- **Minimize consumption and costs of the operations:** our users continually report reductions of 10% in consumption ratio (OCE) in less than one year of using the system and that helped them to know in real and historical time what machines or people in organization could reduce their consumption, and find 100% of detail what causes them.
- **All information in real time:** forget the problem of expensive analyses of energy consumption, and then separately of the production. Integrated information and notifications by email, allow you to detect and correct any deviations in real time.
- **No limitation of types and variables of consumption:** a help to reduce any type of consumption, as for example: *the energy, materials, amortization of assets, damage or specific position, generation of CO2*.

MODULE FUNCTIONALITY:

- **Visualize**, **interactively** in real and historical time, from any device, which people, machines or groups of them, **are not being profitable** for your organization (because are not consuming the minimum), so you can request a **justification** to the responsible for that process. Also you can see the costs, the CO2 generated and saved, the evolution of all the values in the time...
- Supervisors and management can in real and historical time see with certainty of **100% their inefficiencies**: what separates organization from productivity and total efficiency.
- Indicate **consumption** of both, resources and energy that generates each person and machine, or instruct edinn® M2 system to get this information automatically, and then **compare** them with minimum and maximum objectives of your

organization.

- **Notifications:** if you want, you can receive a emails to **certain events**: *when any machine, person or group of them is not being profitable, or when a user makes a comment for improvement, etc.* or you can configure users to receive **periodic emails** telling them if an **over-consumption** has been produced or see it quickly through colors (**red**, **yellow** and **green**), asking for **justification** later.
- Learn about the reasons of over-consumption, according to machine alarms or the **justifications** from the users.
- **Zero papers:** access to electronic documentation or other applications linked to any element of the system: *machines, incidents, maintenance and procedures, quality, etc.*

Costs and Incentives

The **COSTS AND INCENTIVES** module helps the organizations to:

- **Know the actual costs as a first step to reduce them:** edinn® M2 system shows almost everything that happens in the operations. If economic costs are established to everything, the system will show virtually the real costs of all the operations.
- **Motivate you to reduce your costs:** with reports that are shown in red, all deviations in costs in more than one certain percentage motivate you to keep them under control.
- **Simple model form can include all costs:** direct or indirect, there is no limitation to the number of costs that can be considered in each process.

MODULE FUNCTIONALITY:

- Allows you to configure costs in practically all the elements of the system. Since the system registers and monitors everything that happens in production, it offers complete full cost of production.
- See in a **simple** way, through colors **red**, **yellow** and **green**, when a machine or person is not taking the minimal cost to be profitable and then request a justification for responsible process.
- Without any limitation in the number of variables of costs that can be considered and calculate for each process.

Quality and SPC

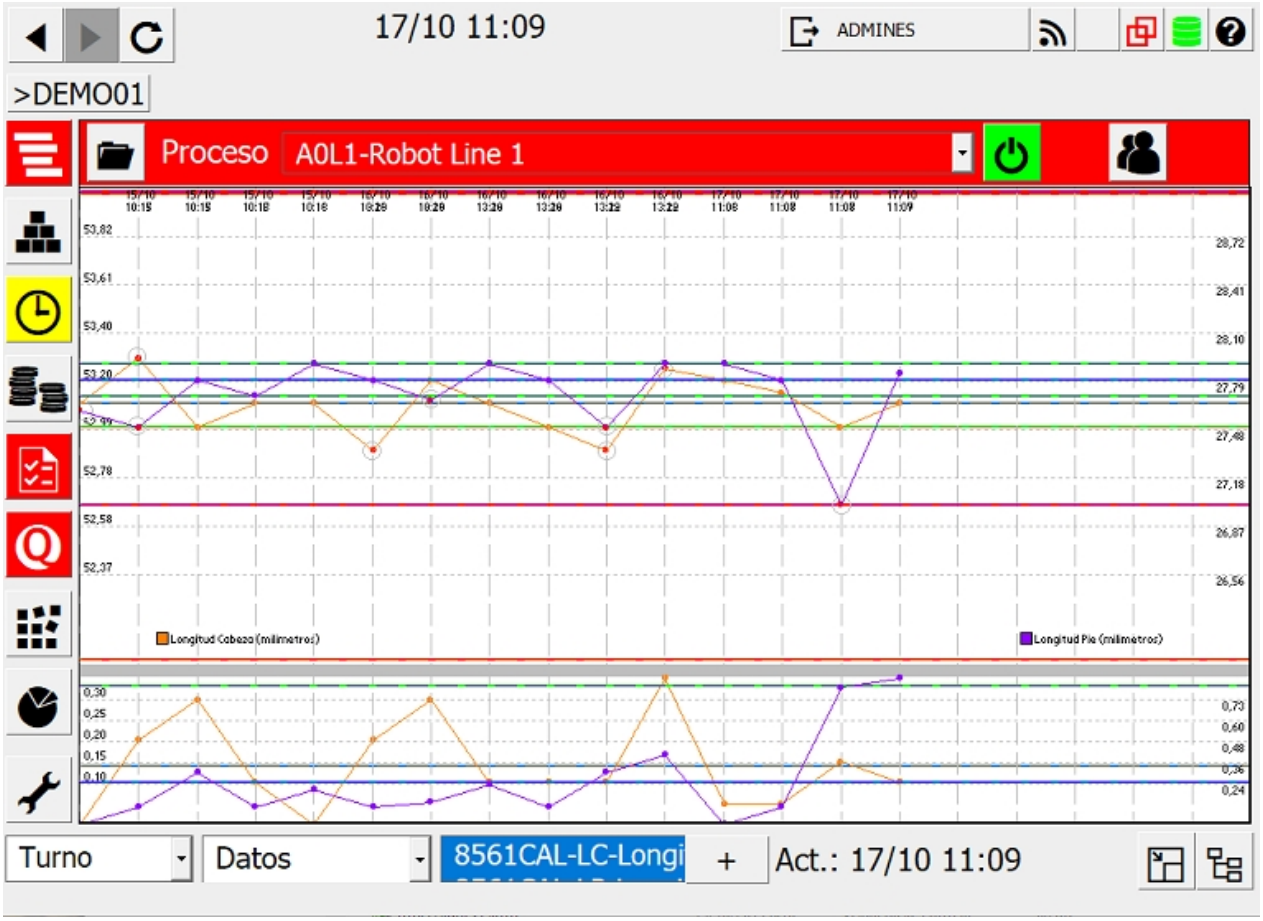
The **QUALITY AND SPC** module helps the organizations to:

- **Increase the quality of the processes:** with a tool that allows to measure automatically, manual or semi-automatic, the quality of the processes, without limitation of variables, in real time, or past, and following easily methodologies as SPC (Statistical Process Control).
- Overcome the most exigent **certifications** and **audits**: as they are exceeded every year by our users in industries, for example, from the automobile and power sectors.
- Register all the data automatically in a **visual form**.
- Improve the organization´s profitability and competitiveness.
- Reduce the possibility that defect products or services reach clients.
- Make **production more efficient** because there is nothing more inefficient then producing a product that will be rejected in the end.
- In some cases **improve production security**.

MODULE FUNCTIONALITY:

- Provide a **software tool** to automatically **capture production context data** (pressure, temperature, etc.), **quality parameters**, or perform Statistical Process Control (**SPC**)
- Define as much quality **variables** as needed.
- **Visualize**, in real and historical time, from any device, the **status and evolution** of the quality variables measured. Obtain and visualize automatically the **main quality ratios** such as: *CPK, Kurtosis, Skewness, Coeff. of Variance, etc.*
- Visually shows the evolution of quality parameters, to **detect and correct** problems as soon as possible, minimizing losses by non-quality.
- **Anticipates** and automatically **notifies** of no-quality situations: *ascending or descending points, over the average, etc.*
- Visually **notifies the user** when he needs to carry out quality autocontrol tasks or when he should take measurements of quality parameters. Also allows to **inform** the user on process quality and actions to take to correct the process and avoid non-quality situations.
- edinn® M2 system can **require users to justify** quality problems, and keeps all the log data electronically.
- **Interactive graphics:** get more information by clicking on a measure, to obtain the context information, such as: *quality variables, who took the measure, if there is a justification for the measure...*

- Compare 2 quality variables to **detect influences** between both.
- Can be **combined** with workflow, incidence management and task approval with quality verification.
- **Zero papers:** access to electronic documentation or other applications linked to any element of the system: *machines, incidents, maintenance and procedures, quality, etc.*



Stock and Traceability

The **STOCK AND TRACEABILITY** module helps the organizations to:

- **Manage the stock in real time:** helps to reduce and optimize to the minimum needed. Maintain information on where and how each element has been produced and consumed, and synchronize it to all the organization to work as clockwork.
- **Reduce stops:** because of lack of stock.
- **Easy and visual:** edinn® M2 system is more visual and easier to use than many management business systems (ERP) or Warehouse Management (WMS).

- **Multi-platform:** use the device that best suits your needs: PC, tablet or smartphone.

MODULE FUNCTIONALITY:

- Monitor the **material planning** and **flow** for production control.
- **Report your ERP** in real time what stock is consumed and generated, based on each result that is produced in every process or work order. Consult the **available stock**, readed from the warehouse in real time, if integrated with the ERP.
- **Detect deviation** from the forecast in the consumption and generation of stock in the **reports**.
- **Employees** will have the updated information of **stock availability**, where, in what batch, etc. They block each batch every time they use it, so other workers can not use it.
- **Visualization** of buffers between operations, material and product back tracing.

Personnel

The **PERSONNEL** module helps the organizations to:

- **Manage**, **motivate** and increase the **performance** of the teams**:** use the edinn system to motivate those teams who want to get the best performance (e.g. *the OEE*).
- **Calculate incentives with total accuracy:** the edinn platform is also a MES software, automatically captures data and makes accurately all the incentives calculations in the currency selected.
- **All situations are configurable:** edinn® M2 system will adapt to your management staff, [illegible]

MODULE FUNCTIONALITY:

- Easily manage **who**, **when** and **where** is working.
- Calculate the **performance individually or grouply** of the employees, since they can indicate on what machines are they working. Using the activity report, in real time, where you can know exactly by area, process, and by each resource (person), which results has been produced, in which machines, what was its performance and if it has some time without post of work.
- Obtain a system of **incentives** which provides a report that objectively indicates

the **exact amount** corresponding to each worker, in your currency (multicurrency).

- Available **FTE (Full Time Equivalent) report:** in real time by area, process and each resource (person), to know if the hours of work that the company is paying coincide with those who have worked and with the expected performance, both in production and in service. And in case this is negative, identify discrepancies.
- Set a **minimum** and **maximum number of employees** on each job position. When the maximum number of workers is exceeded, decide how you want to manage those workers, among many possibilities. For example, you can allow that one worker removes another worker, or that he needs to notify the other worker. The goal is to correctly manage employees and their associated costs. **Access control** is convenient to control:
    - When users or guests accede processes or areas.
    - When users of the system accede or modify certain items.
    - Permissions that each user has to consult, modify, approve, etc.

The edinn® M2 system incorporates a powerful access control system based on permissions and control functions.

Main functionality of edinn® M2´s access control is:

- Unlimited number of users.
- Access control with passwords.
- Individual files per user.
- User activity control, especially on data entry and information modification.
- Permission control by roles.
- Access by codes and magnetic cards.

Standard Integration

The **STANDARD INTEGRATION** module helps the organizations to:

- **Obtain information in real time, consistent in all systems without having to enter it more than one time:** the edinn® M2 system knows almost everything that happens in your production. Plug it into your current system to improve them and drastically improve your organization.

- **Some say that everything can be integrated, but we recommend that you protect your investment and reduce the initial costs and maintenance:** because edinn® M2 system use ISA-95, XML and web services that are more accepted standards.
- **Always available:** if communications fail, or if the other system is closed (example by closing accounting ERP), users will consult the available stock or performing any other function in the edinn® M2 system. Once when systems re-connect, the information will be synchronized again.

MODULE FUNCTIONALITY:

Connect your edinn® M2 system with any of your other systems, through standards and in real time.

- Exchange information automatically controlled, in real time and delayed, with any compatible system with the ISA-95 (http://www.isa-95.com/) standard, such as PeopleSoft and SAP.
- Exchange information automatically with any machine via the OPC standard (https://opcfoundation.org/).
- Manage without any problem falls in communications, the edinn® M2 system will continue to operate autonomously and it will transfer to another system all the [illegible] when the connection is restored.
- When some integration is not covered by the ISA-95 or OPC standard is necessary, it is possible to develop new integrations with development module which is incorporated or incorporate them into the standard edinn® M2 system as new integration modules.
- Allows you to launch external applications and documents in an integrated way from the user environment.
- Internal comments posted by users of edinn® M2 system, if desired can by passed on to the email.
- Export in one click any table data form report from edinn® M2 to Excel.

Artificial Vision

The **ARTIFICIAL VISION** module helps the organizations to:

- **Learns in Real-Time**: Let the module capture some labeled images, learn from scratch, and classify quickly and accurately using AI. Labels can come from Edinn.
- **No Maintenance**: There is no need to retrain models or perform periodic maintenance.

- **Patterns to Signals**: By converting visual patterns into standard signals, subsequent uses are limitless, as they are considered signals like the rest and can therefore be combined with other system modules.

If a camera can see it, you can **monitor** it, **control** it and **improve** it.

MODULE FUNCTIONALITY:

- **Simple and visual configurations:** allows to configure simply, step by step, the visual patterns that the system must monitor in real time and at high speed.
- When converting **visual patterns** in **standard signals**, further applications are unlimited since they are considered signals like the rest, and therefore, they can be combined with other modules of the edinn system.
- Optimized for **high sampling rate** and **pattern recognition** ultimately depending upon limitations like: speed camera, sending the pictures and processing the speed of the server.
- Possibility of **centralized monitoring** (in the central server) of all the images, or from **distributed**, for greater speed, directly in the process or using mixed solution.

Big Data

The **Big Data Module** helps the organizations to:

- Improve your processes by using recommendations and statistical analysis.
- Speed up the platform, obtaining, for example, reports and complex dashboards at a high speed.

MODULE FUNCTIONALITY:

- Makes available advanced reports with recommendations and statistical analysis.
- Prepares data so that multiple parts of the system work faster: reports, dashboards and other functionalities.

Failure Prediction

> **NOTE:** The Artificial Intelligence (Envision) module is currently in Beta version and subject to changes.

The **Envision Module**, which uses Artificial Intelligence, helps organizations to:

- Avoid high costs and problems of certain failures, knowing them before they occur with sufficient advance.
- Validate the success rate of the predictions, before notifying users.

MODULE FUNCTIONALITY:

- Sends notifications of certain faults, before they occur, so that they can be avoided. A button allows to check the success rate of predictions.
- If the success rate of predictions is high enough for your organization, allow the anticipated notification of the user to avoid the failure and achieve significant savings.
- Avoids up to 8% of all failures: according to tests performed with data of real users.
- Minimum maintenance: automatic self-training.
- Multipurpose: "intelligence" ready to help in different areas.

WARNING: As predicting failures is a future forecast, it can never be guaranteed with certainty.

Development Platform (API & UDL)

The **DEVELOPMENT PLATFORM (**API **&** UDL**)** module helps the organizations to:

- **Would you like that edinn system suits your need more specific or you want to make it more powerful?** Adapt it and improve it by yourself from your favorite development environment capable of developing ActiveX, or request us to do it for you. Always retain the code source of the developments. Develop any functionality, report or application you need, in case you cannot find it in the standard modules.
- **Training:** we have made many developments that are working today, and we can train you to do it so.
- **Multitude of real examples:** generate special reports, make a siren sound, stop machinery, control cameras, etc. If you are able to design it, you can probably do it.

MODULE FUNCTIONALITY:

- Resolve any specific needs that your organizations have, develop it by yourself or by your partners, retaining the code source of developments.
- Develop, inside your own organization or through one of your providers, any specific need.Use the development environment that is preferred.

- Get courses and examples of developments to take ideas. We made available many of real examples of applications.

For more information, please see the API and the UDL guides of this manual.

Community

Version **edinn Community** is automatically activated when the license field is empty in the console and only on servers which are out of the edinn cloud infrastructure. It provides free and permanent* license for 1 process and the Production Control, Big Data and Artificial Intelligence and Development Platform (API & UDL) modules. This, together with the integration with FlexSim, the leading process simulator, makes the edinn platform of Industry 4.0 possibly the best for partners, educational institutions, start-ups and companies.

Version **edinn Community** has these limitations:

- There are no additional modules available.
- Allows only working with 1 process.
- The wizard for companies creation cannot create companies in those servers which have already a company in Community version.
- In dashboards, it is not possible to copy a link from a widget and then open that widget separately from another browser.
- In the resources configuration, it does not allow to generate and copy the link to the web dashboards of the resource.

* For more information y to know additional conditions, please read the Terms and Conditions.

Performance

As time is money, especially in a productive environment, response and operation time for users has been a key priority in the development of the edinn M2 platform, which has led to:

- **The user experience, in terms of operating, opening windows or showing graphics or tables of data, is normally almost immediate (less than 1 or 2 seconds)**; meanwhile the edinn M2 servers are, in some installations:
  - Reading or writing hundreds of records per second in real time.
  - Updating hundreds of reports per minute.

    - Showing dozens of dashboards, refreshed and updated once per minute of even more frequently.
    - Storing hundreds of thousands of data records per day.
    - Managing hundreds of millions of data records in the archive.

- **Edinn M2 servers need to be comparatively much smaller**, in CPU and RAM, than those of other market solutions covering some of edinn's functionalities.
- **Performance ratios are kept almost identical during years**.

This is thanks to, among other characteristics:

- **In memory database**: as RAM memory is much faster than hard disk (typically up to 1000 times faster), all the data in the server runs in-memory, if the server has enough RAM. As data will grow with time, more RAM memory might be needed, although the edinn M2 system is clever enough so that un-used data will be replaced by needed data.
- **Indexed data**: all data is indexed so that searches are ultra-fast as they are binary.
- **Reports shown instantly (less than 1 second) to users**: favorite reports are automatically pre-cached by the edinn M2 Platform and therefore shown almost immediately to the user. In addition, this saves important server resources as the same report is never calculated more than once, unless manually requested by the user.
- **Optimized source code**: as a mature solution, the great majority of the edinn M2 Platform components have gone through multiple optimization iterations.
- **Auto optimization**: all systems degrade with time; therefore, auto optimizations must be performed periodically to keep performance. That is the case of the edinn M2 Platform, which periodically, and transparently for the users, performs tasks to improve performance, from archiving real time data to database optimization. For more information please read this document.
- **Multithreaded and buffered**: the edinn M2 Platform is composed of 64 software components running independently, not including the 2 external standard and very important components which are the APACHE web server and Oracle's MySQL Database. Apache and MySQL are also multithreaded. All this combination of threads makes the system very powerful and flexible to use all the available CPU cores. This also applies to monitorization which, in addition, communicates with the rest of the system via buffers. Thanks to this, edinn M2 has succeeded in projects where it had to be able to capture signals changing many times per second from dozens of processes; not losing any signal change.
- **Native executable files**: all the software components, which are critical in terms of processing speed, are native executables, meaning that they are faster as they run directly on the CPU, without additional software layers like interpreters or engines. Functionalities which are depending on human response, might be served by components running on interpreters and engines (like PHP and others).

- **Big Data**: the system automatically, and transparently to the users, creates and maintains data cubes (requires the “edinn M2 Big Data Module”) to speed up the whole system, from dashboards and terminals to reporting.

Basic guide

This basic guide explains the basic procedures to operate with the terminal win32 version, as well as the different windows that are displayed in the main window of the system.

> To configure the system, please see: *Configuration*.

> To obtain system reports, please see: *Reports*.

Interaction

The standard ways for a human to interact with the edinn M2 platform are:

- **The web reports**: to view (not modify) information.
- **The web wizard**: for the first basic configuration of a company.
- **The terminals, in all their different versions (win32, mobile, etc.)**: for the complete configuration and operation of a company.
- **The server console**: for the complete configuration of the monitoring and certain server-level parameters.

The standard way for a program to interact is:

- **The API**.

In addition to these standard ways, there can be other tailored developed (through the API and UDL).

Win32 Terminal

This section covers the win32 edinn M2 terminal.

1. Install the Terminal

In this document we will see how to install the edinn M2 terminal in its Win32 version.

WARNING: if you are using any **antivirus** that is not the Windows standard (for example **McAfee**), the following procedures may not work, and your antivirus will not warn you about it; temporarily disable it before continuing.

Follow these steps:

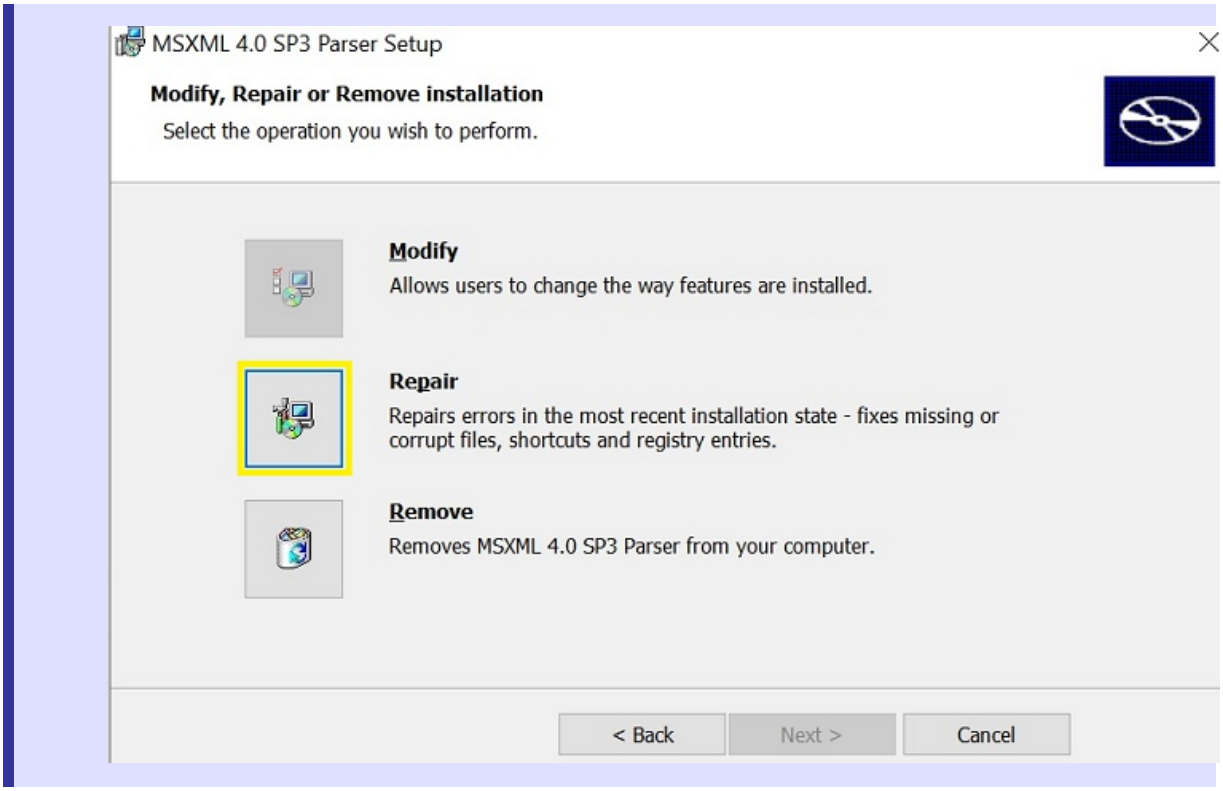
1. Download this file: /edinnM2/client/edinnM2_client_setup_win32.zip
2. **Select** the downloaded file and click with the **right button of your mouse** and **extract** the file into a **folder** in your hard disk.

WARNING: Do not double click on the downloaded file. As indicated above, instead, unzip the downloaded file as a new folder into your hard disk, and follow the steps as indicated previously.

3. **Once extracted into a folder**, enter into the new folder, **select** the file **Setup.exe**, click on the **right button of your mosue** and click on **Run as administrator**.
4. Installation should start. Follow the default steps.

Notes:

- If you are informed that some files that you are installing are **not newer than the ones you have**, answer "**Yes**" each time if you want to minimize the changes on the computer, or "No to all" for a quicker installation.
- If you are reinstalling the terminal, you will be asked if you want to repair or remove, select **Repair**, as shown in the image below:



After installing, you need to connect and update it.

After connecting and updating, you will be able to login to your company.

If you need help, please request support to edinn.

2. Connect and update

After installing the edinn M2 win32 terminal, **for it to work**, it must be **connected** to your server and company and **updated** to the version of that server. Please follow the next steps to connect and update your terminal.

Note: If that terminal has been recently installed and If you are not using the default edinn cloud infrastructure (ecloud.edinn.com), then you will need the domain name, or IP Address, and the Port of your server. Please ask your edinn system administrator for this information.

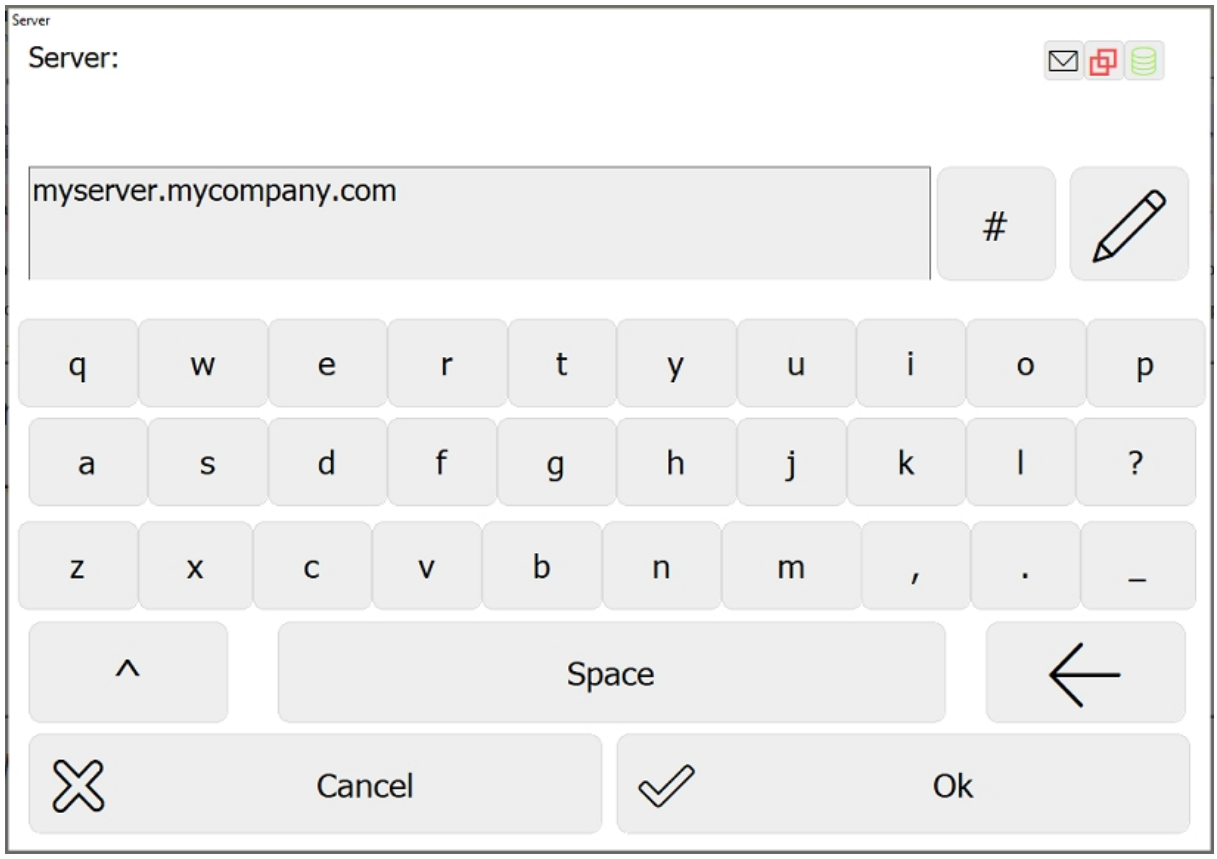
WARNING: if you are using any **antivirus** that is not the Windows standard (for example **McAfee**), the following procedures may not work, and your antivirus will not warn you about it; temporarily disable it before continuing.

WARNING: To be able to update your terminal, be sure to open it with **administrator permission**. To do that, open it with the **mouse right button** and select **Execute as Administrator**.

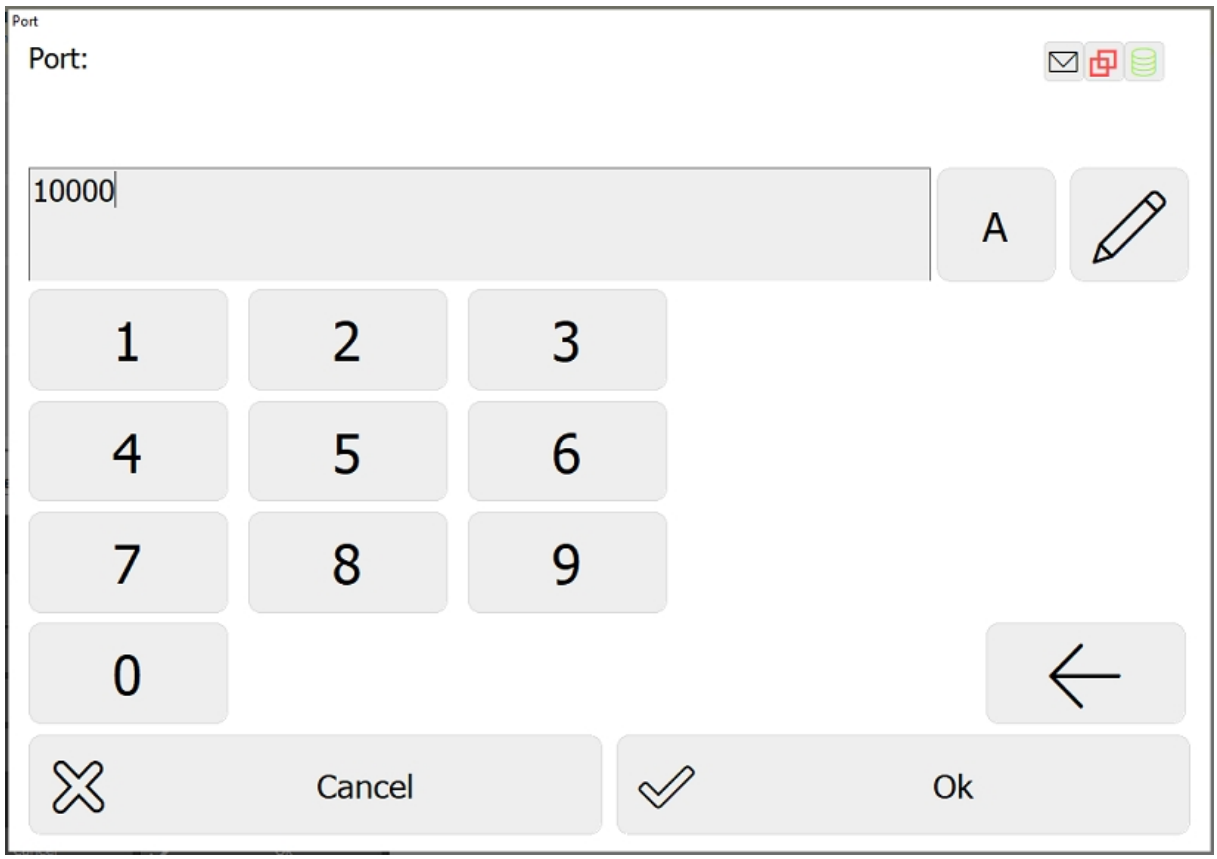
From the Login window of the win32 terminal, we can see at the top whether or not we are connected to our server ( ) and database ( ), since these icons will be on **green** (if connected) or **red** (if not).

To make the connection to your server and update your terminal, press the first icon, and fill the following fields. If you do not know these data, contact your technical support.

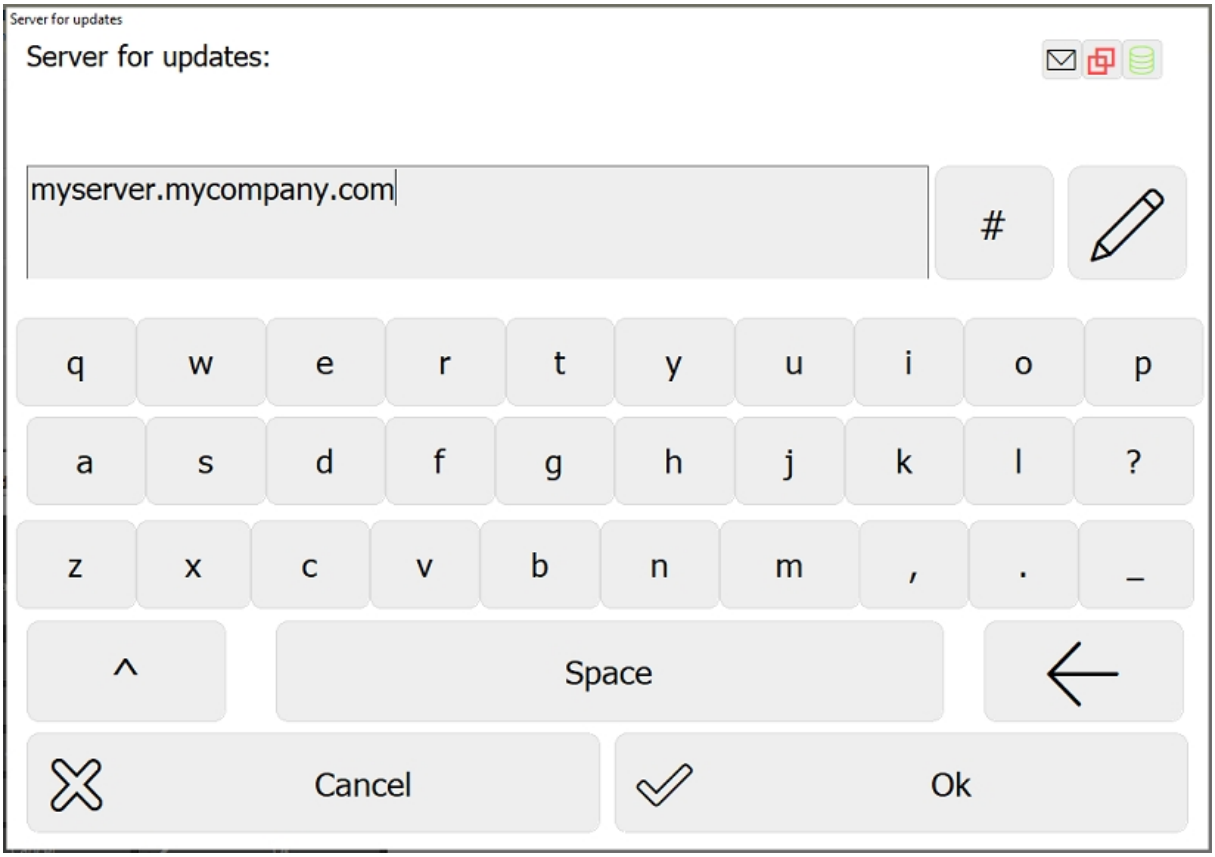
1. **The server IP:** type **your** server IP where edinn is installed:



2. **The server port (typically 10000):**



3. **The server IP for updates:** type **your** server IP where edinn is installed. It is recommended to type the IP address of your server instead of its name. To obtain the IP address you can open a Windows DOS session (Windows key and type CMD) and type PINT NombreDeSuServidor, which will show you the IP address of your server:



Then **answer Yes** when you are requested if you want to automatically update your terminal. If there is a different version in your server, your application should update automatically. If it does not update automatically, you can view how to update the terminal manually.

Note: if after updating, you cannot see the icons of the application, restart your computer and start the application again to check if you can see the icons.

3. Login

To open the operation terminal (win32 version) of the edinn M2 platform:

- Double click on the edinn M2 icon of your desktop.
- Or search "edinn" in the Windows search engine and execute "*edinn M2*".

Login screen: there are 2 types of login windows.

NOTE: to configure the type of login please see devices configuration.

WARNING: If your terminal does not connect, try first the procedure to connect and

update, second to reinstall, and if still not connecting, please contact support.

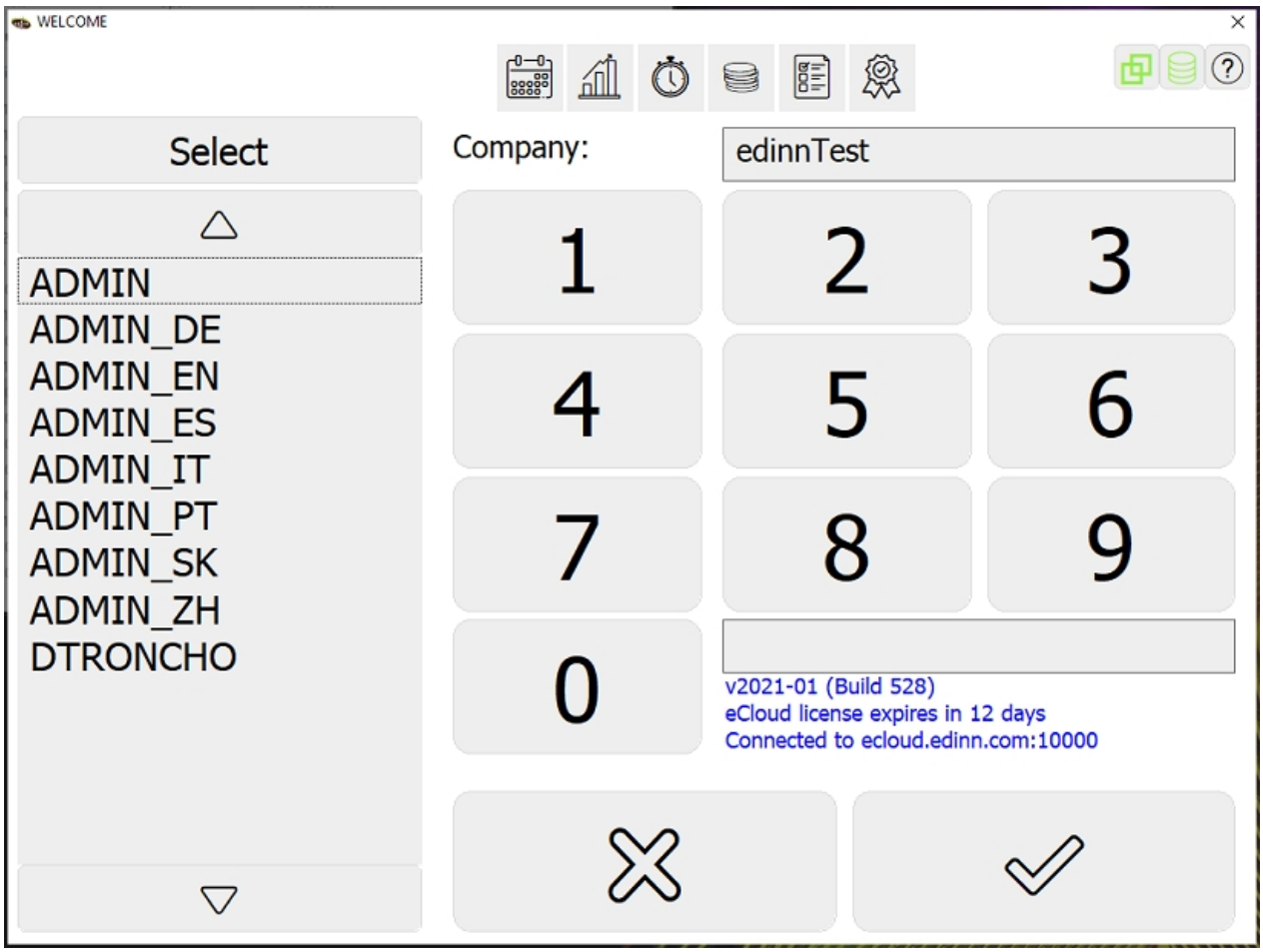
1. STANDARD LOGIN, recommended for terminals with keyboard:

Access requeriments:

- Type a valid organization: Id or Name.
- Type a user.
- Type a password.

NOTE: The 'Remember me' option allows fast and secure login without having to type user and password. It is only available when the 'Allow single sign on' option is activated in the server console.

2. EASY LOGIN, recommended for touch screens:



Procedure:

Type a valid **organization**: Id or Name, and

1. Select your user Id from the left list. The button **[Select]** will help you locate your user id on the list by typing the first letter.
2. Type the PIN password.
3. Click on the accept button.

This window shows, at the top, buttons which can blink to indicate that processes managed by this device, require attention on the following functions:

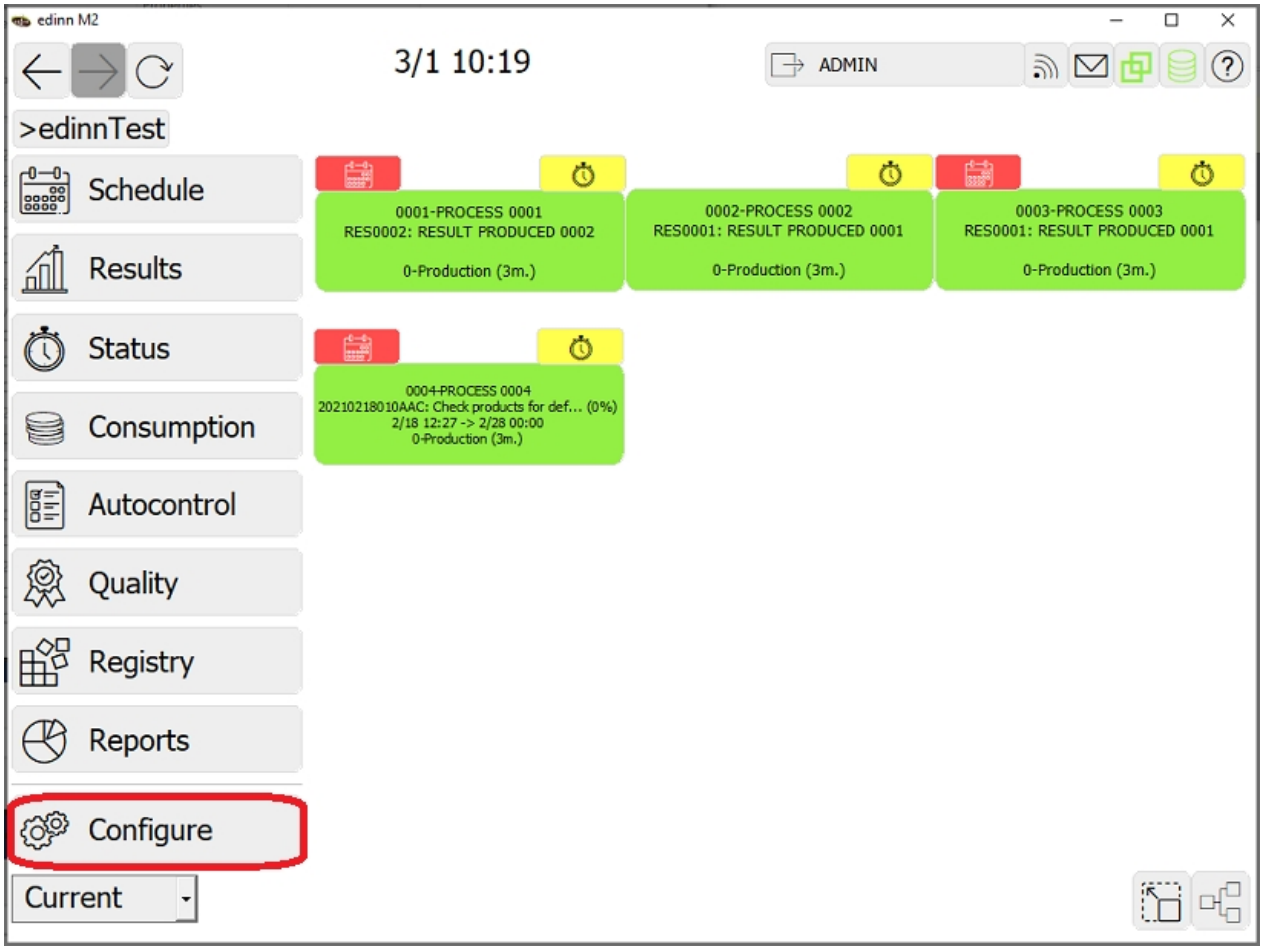
Schedule	Results	Statuses (Time usages)	Consumption	Autocontrol	Quality

4. Change password

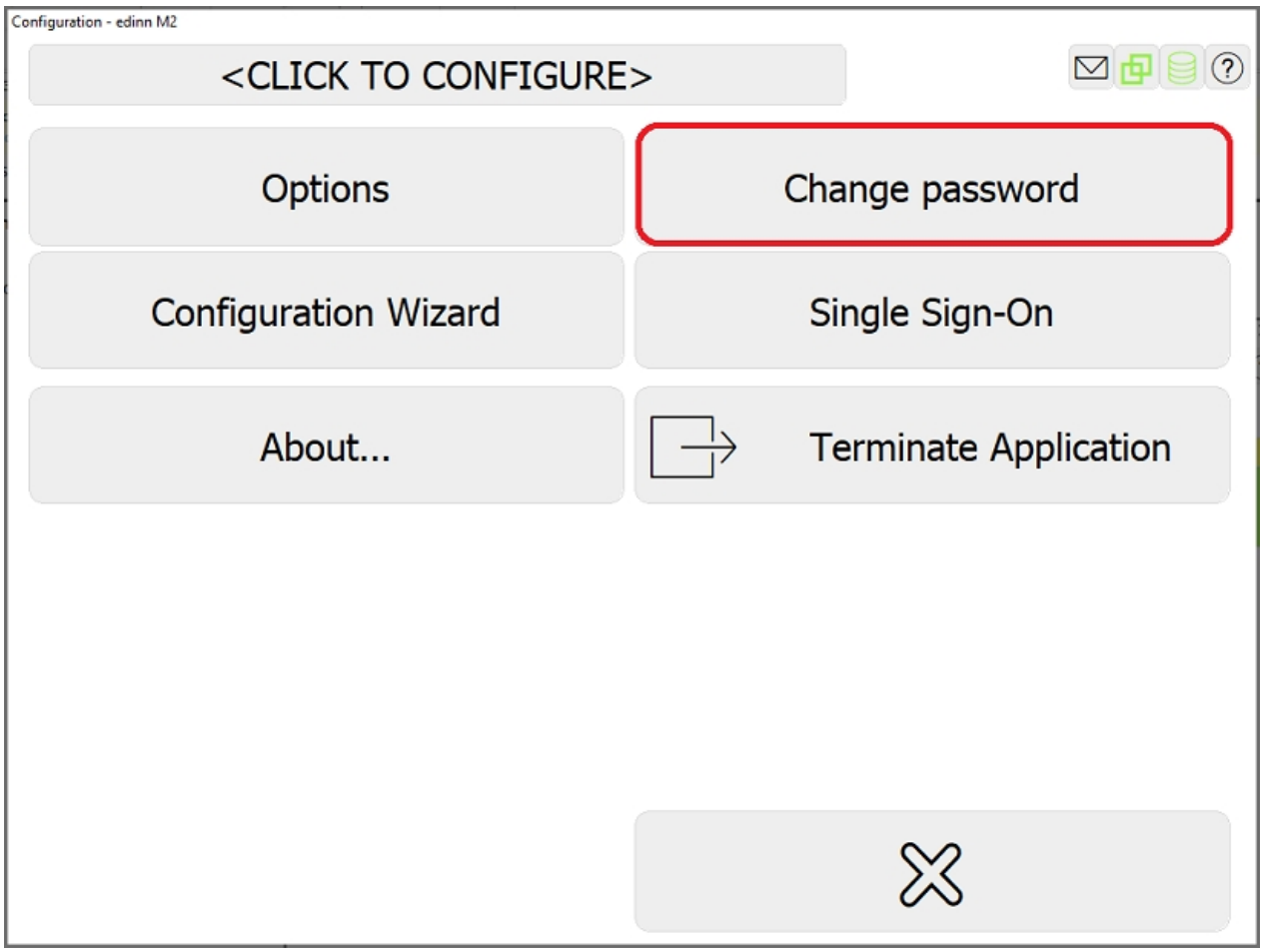
To change your password, follow these steps:

1. Access the configuration window

Press  to access the Configuration window:



2. Click on "Change password"



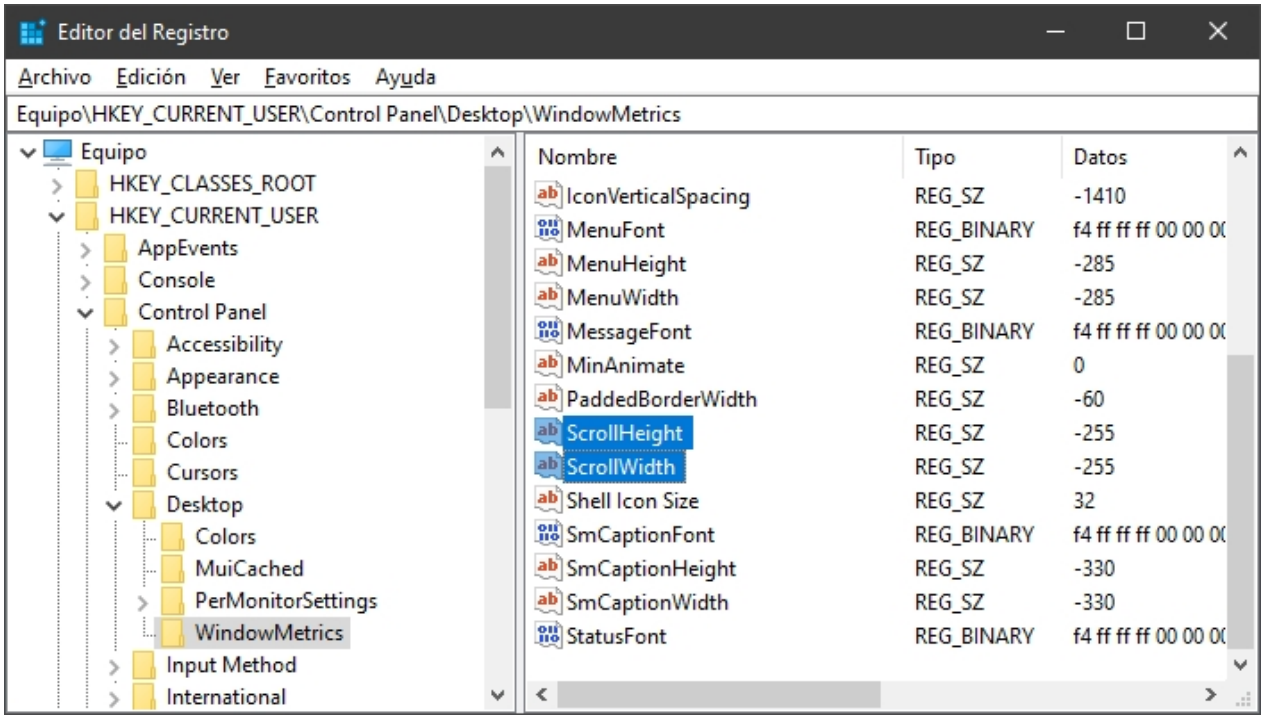
5. Scroll Bars

On touchscreen plant computers where keyboard and mouse are not used, the standard size of Windows scrollbars may be too small.

So that the operators can use them comfortably, you must change their size in the windows registry:

1. Access the registry settings using the regedit.exe program

2. Navigate to the "Computer\HKEY_CURRENT_USER\Control Panel\Desktop\WindowMetrics" section as shown in the image.



3. You will find two parameters ScrollHeight and ScrollWidth which indicate respectively the height and width of the scrollbars.

4. The default value of these is -255 which is equivalent to 17 pixels (it is the result of dividing by -15). Change the value to whatever you want it to appear (we suggest putting -510, which is equal to 34 pixels)

5. For the changes to take effect, it is necessary to restart the computer.

Please note that this change affects all applications.

Main

This is the main window of the edinn terminal (win32 version).

If you are an administrator, in addition to the options shown below, you have available, with the mouse right button, the following menu:

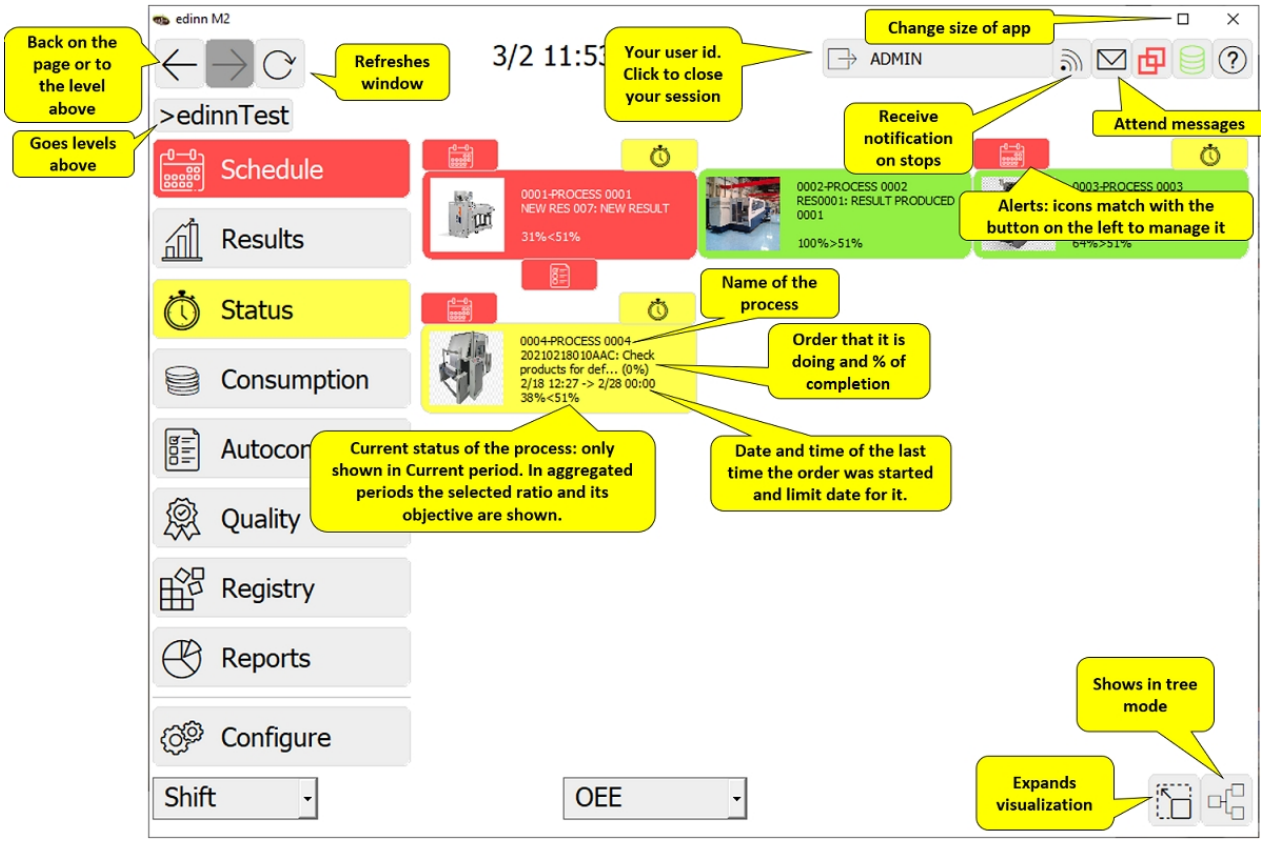
- **Create area**: creates a new area.
- **Rename area**: changes the description of the area.
- **Move area**:
  - Moves the selected area inside of another areas. Destination areas must be areas of areas or areas without processes.
  - Moves the orders/tasks and the processes or areas contained by the selected area to a new area. This action will permanently delete the

selected area.

- **Archivar/Desarchivar área:** archives or unarchives an area. When an area is archived it is no longer available for management with production orders.
- **Delete area**: deletes the selected area.
- **Assign processes**: assigns the selected process to areas or processes to the selected area.
- **Assign devices**: gives access so that the selected devices can see and operate with processes, areas and all their subareas and processes.
- **Assign resources**: gives access so that the selected resources can see and operate with processes, areas and all their subareas and processes.
- **Assign roles**: on a selected area, allows to assign only certain roles to the resources which can view the area.
- **Invite**: allows to invite, through email, new users. This option is only available if allowed in the server.
- **Operación time:** Allows you to change the operation time for a process when it is a person.

> **NOTE**: The administrators of areas will only have the menu in the areas of which they are administrators.

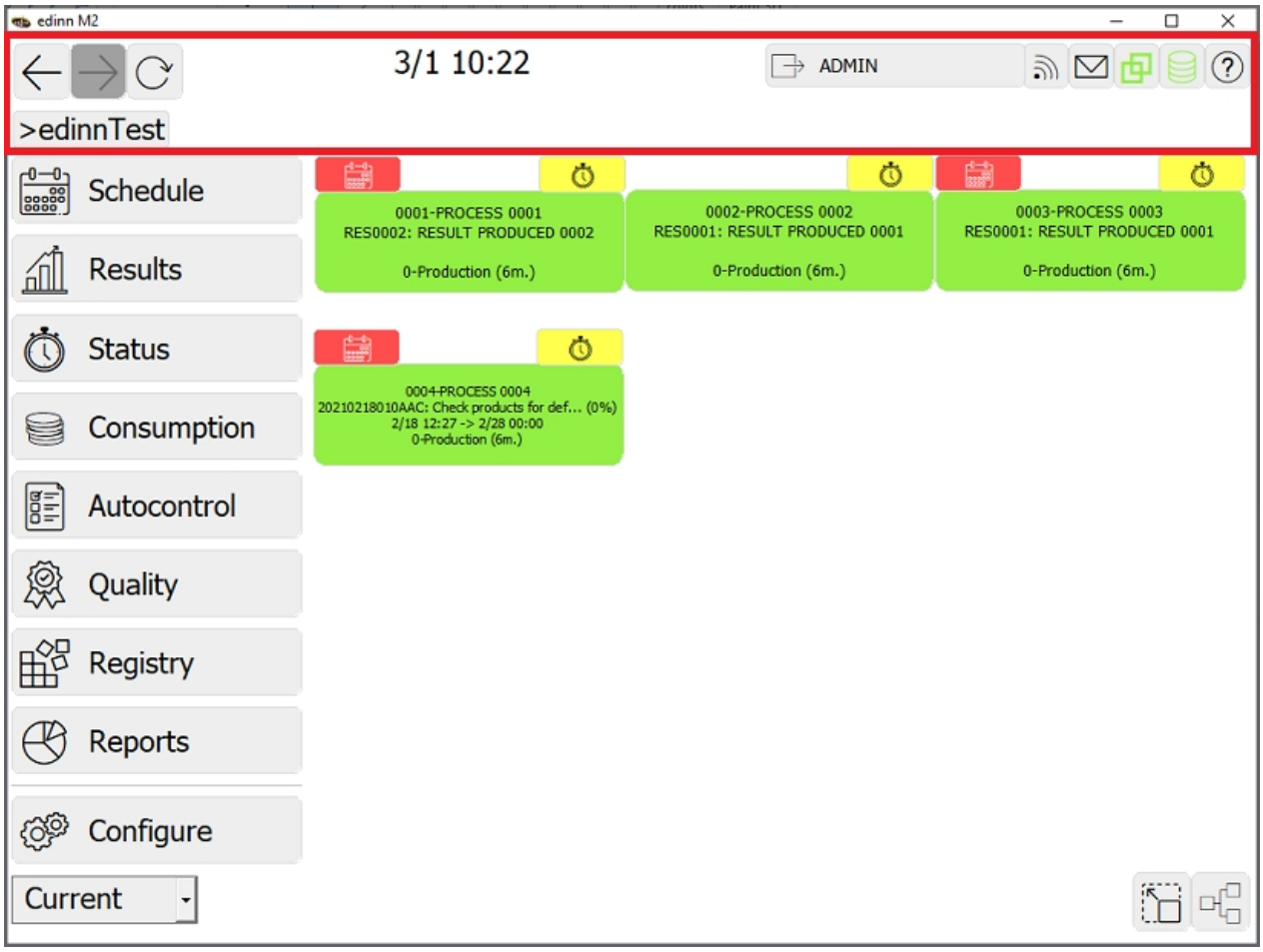
Let us see an example of visualization for a factory (in light theme):



And an example of visualization for an office (in dark bright theme):

Upper Area

In the Upper Area of the Main Window we have the following buttons:



- Arrows: allow you to navigate through the terminal.
- : refreshes the information shown in the window.
- : shows your user id. By clicking on this button you can close your current session.
- : the user has unread messages. By clicking on this symbol you can read the message.
- : if it is green, it means that the terminal is connected and synchronized with the edinn® M2 Central Server. If you can not see this icon or if you see it always in red, please contact your technical support. Click this icon in the Login Window to change your destination server. If this icon appears in grey, it means that we are in a closed period with respect to the interface system, and therefore

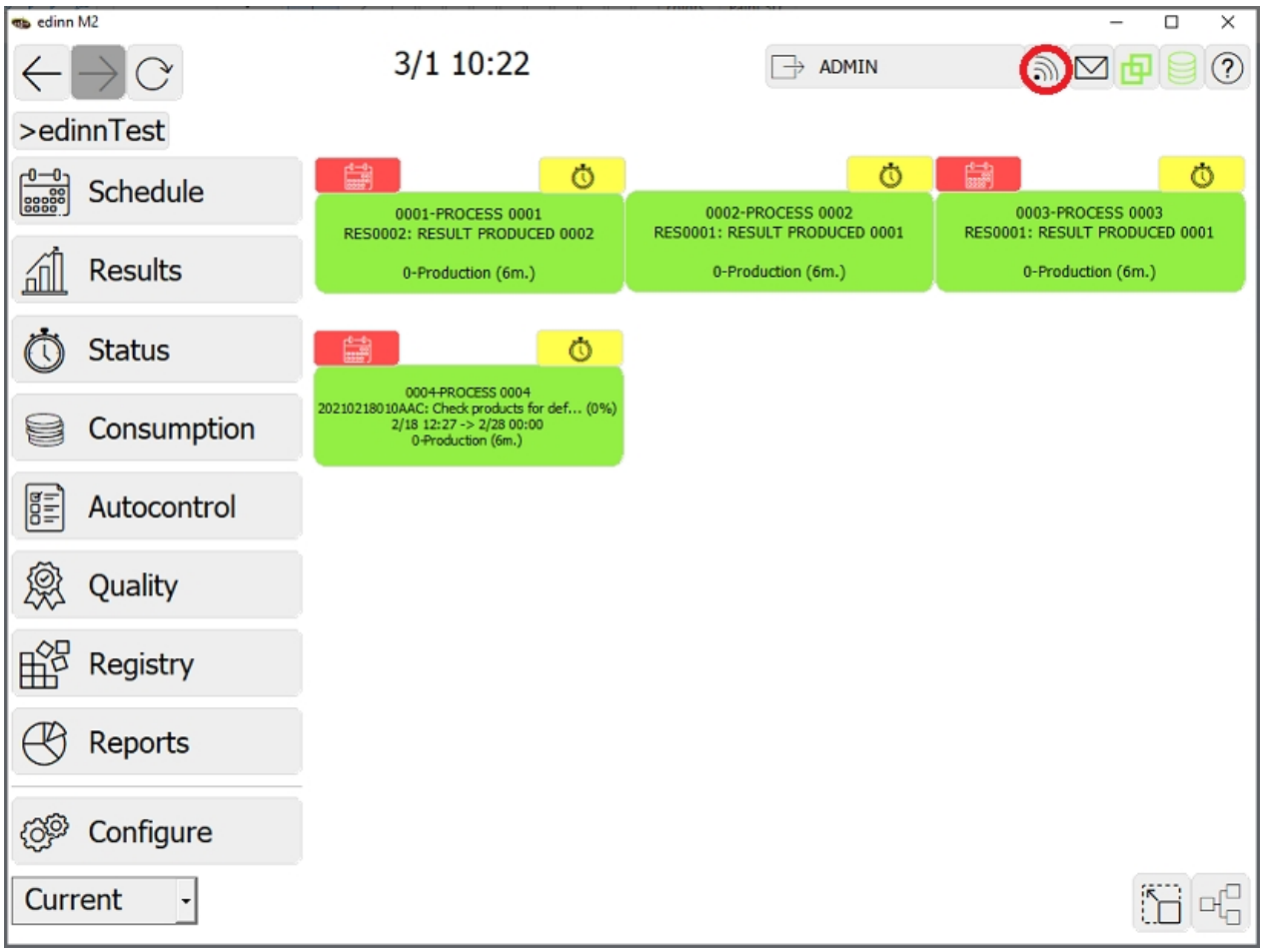
communication with the other system is off-line.

-  : The terminal is connected to the edinn® M2 Central Database. Click this icon in the Login Window to change your destination server.
-  : Provides access to:
  - This help, contextually.
  - On-line support.
  - The on-line list of available edinn partners, to cover other needs that you might have. This URL link can be customized in the Console, Recommendations tab.

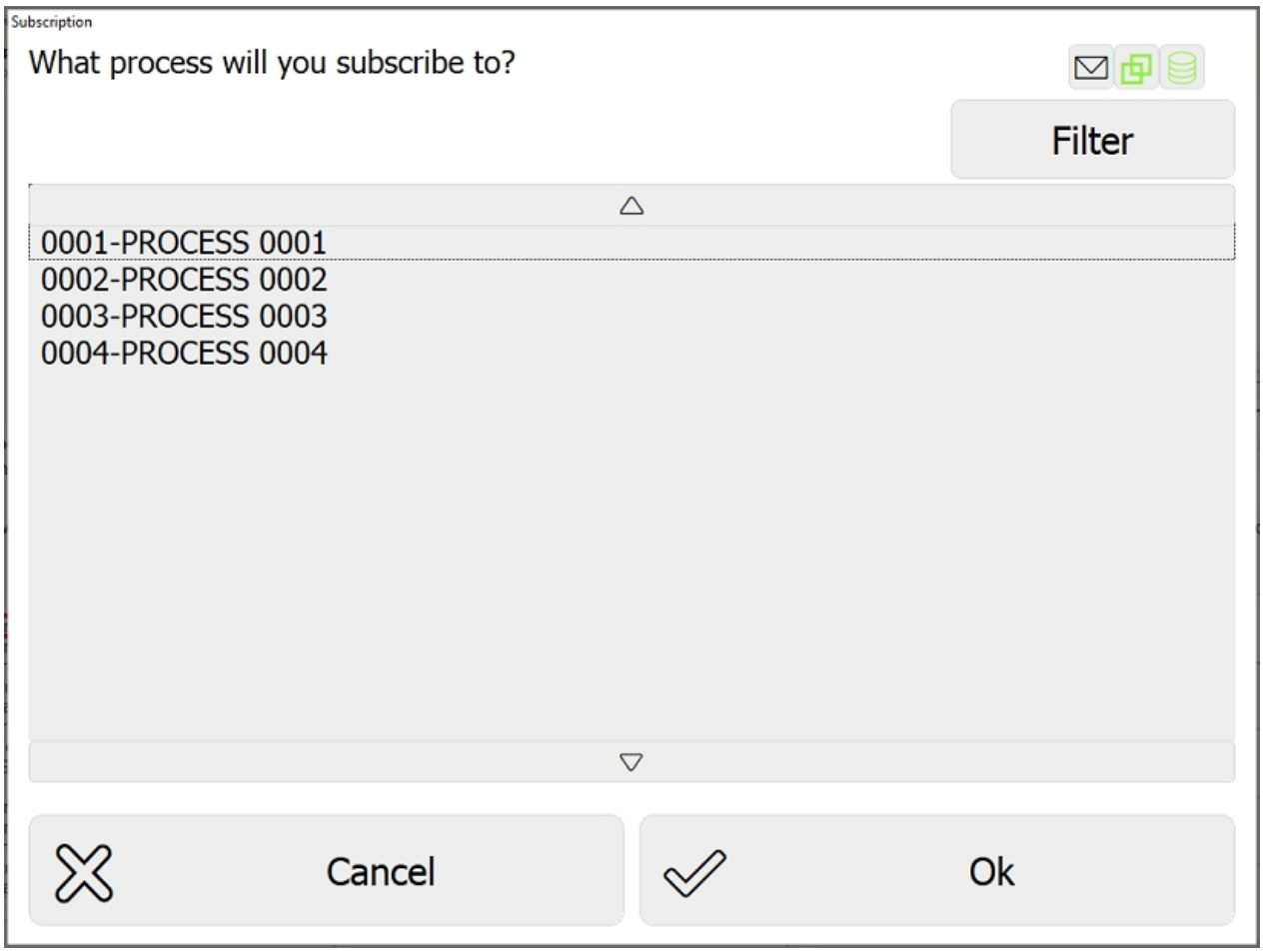
Subscription

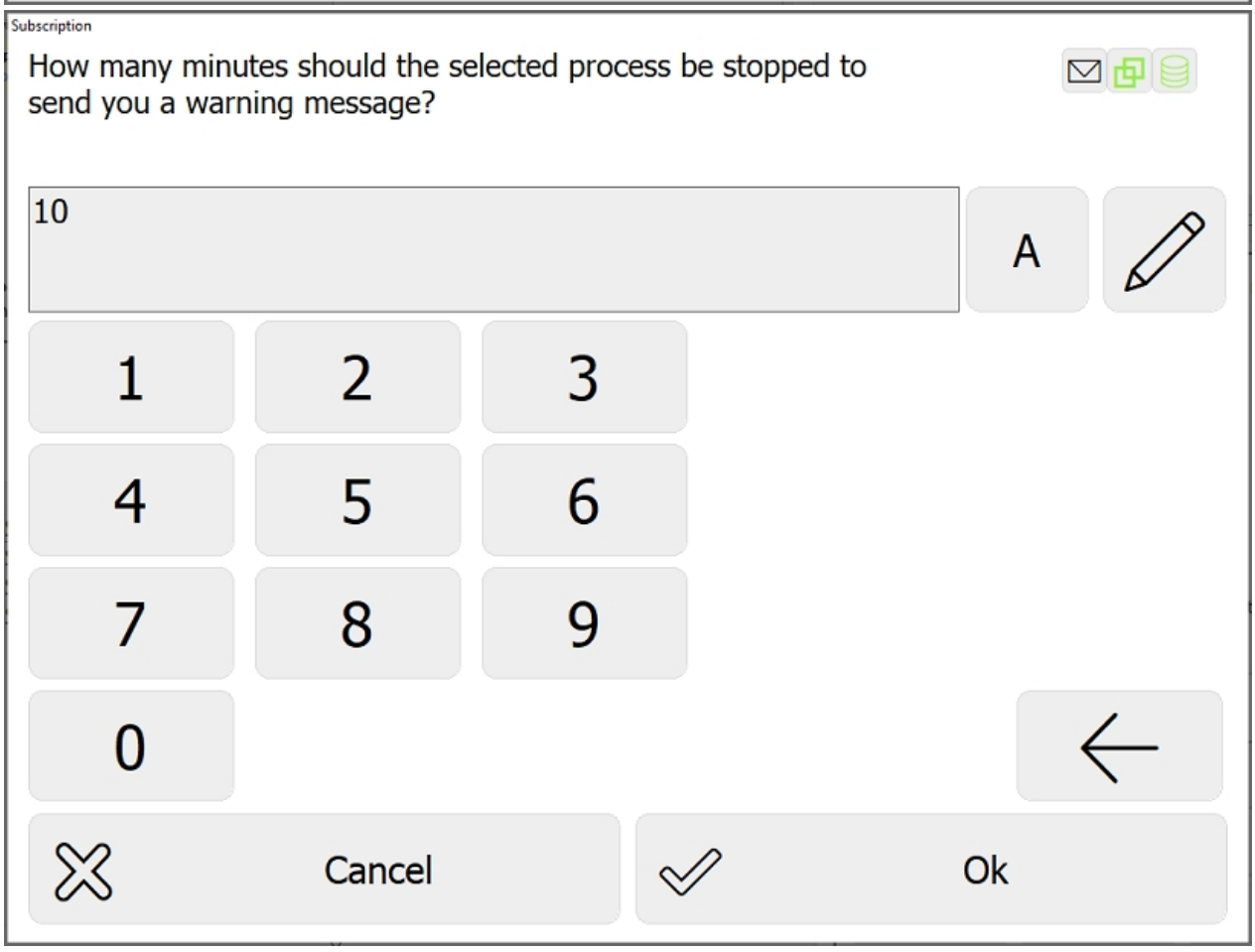
Main Window > button 

Utility: to be informed of when a process or processes are stopped.



To configure it, you must select a process from among those available, and indicate after how many minutes you want to be notified if the machine is stopped. This notification can be sent by an email to the subscribed user, or by the activation of a beacon (or any other type of output)

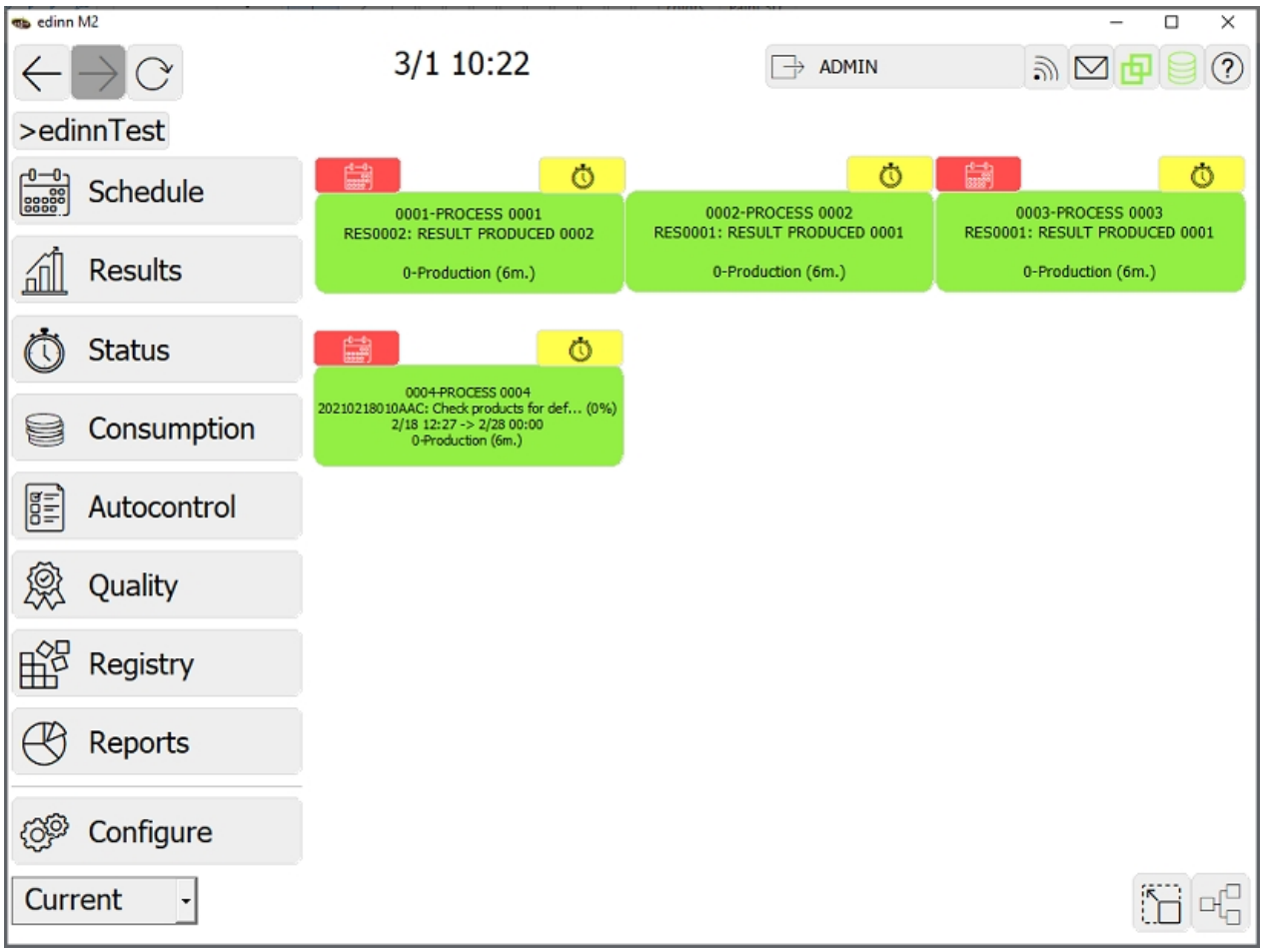

Subscription
What process will you subscribe to?
Filter
0001-PROCESS 0001
0002-PROCESS 0002
0003-PROCESS 0003
0004-PROCESS 0004
Cancel
Ok


Subscription
How many minutes should the selected process be stopped to send you a warning message?
10
A
1
2
3
4
5
6
7
8
9
0
Cancel
Ok

NOTE: when a user closes his session, all his subscriptions are canceled.

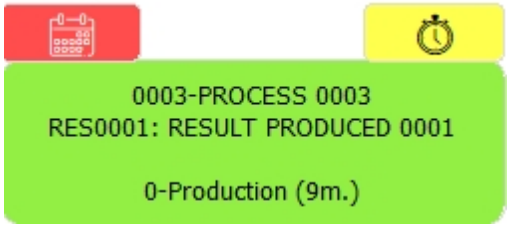
Main Area

In the Main Area of the Main Window we have the following:



Each of these boxes represent an area, by clicking each box you will see the processes that compose that area. In the upper part of each kind of box (area and process), you will see different kind of icons which represent alerts. Some areas could not show alerts if they do not have work orders in process, see the Scheduled option in the Areas configuration for more information.

AREA/PROCESS DETAILS:



There are 6 types of little buttons, corresponding to alerts, surrounding each process or area box:

- : There are tasks or orders that are about to be delayed (yellow) or are definitely delayed (red).
- : Results are expected to be generated.
- : There are not justified statuses.
- : There is an excessive consumption that requires justification.
- : There are pending autocontrol tasks.
- : Quality (SPC) data must be captured.

NOTE: except for the alert of delayed tasks or orders, alerts of areas will only be shown aggregating the alerts of the processes which are working in tasks or work orders of that area.

PROCESSES DETAILS:

The area or process boxes can have **little color buttons** at the top:

This buttons can be in red (require urgent action) or yellow (require action when there is not anything urgent to attend). If you click this button, it will take directly to the window where you can solve the issue.

In the upper part of the process' boxes, detailed information will be shown:

- : is shown when there is any additional documentation or a different application relating to the process.
- : ON /OFF button, is only shown when the user is allowed

to indicate to the system that this process is working or stopped. In the case of automatic processes, this button will appear if the process has an On/Off signal in the monitor. This information will be validated by the system and not accepted or even deleted if incorrect.

- : here you can assign resources, also know as IN&OUT. For more information, please see: "Assign personnel and team window"

Assign personnel and team

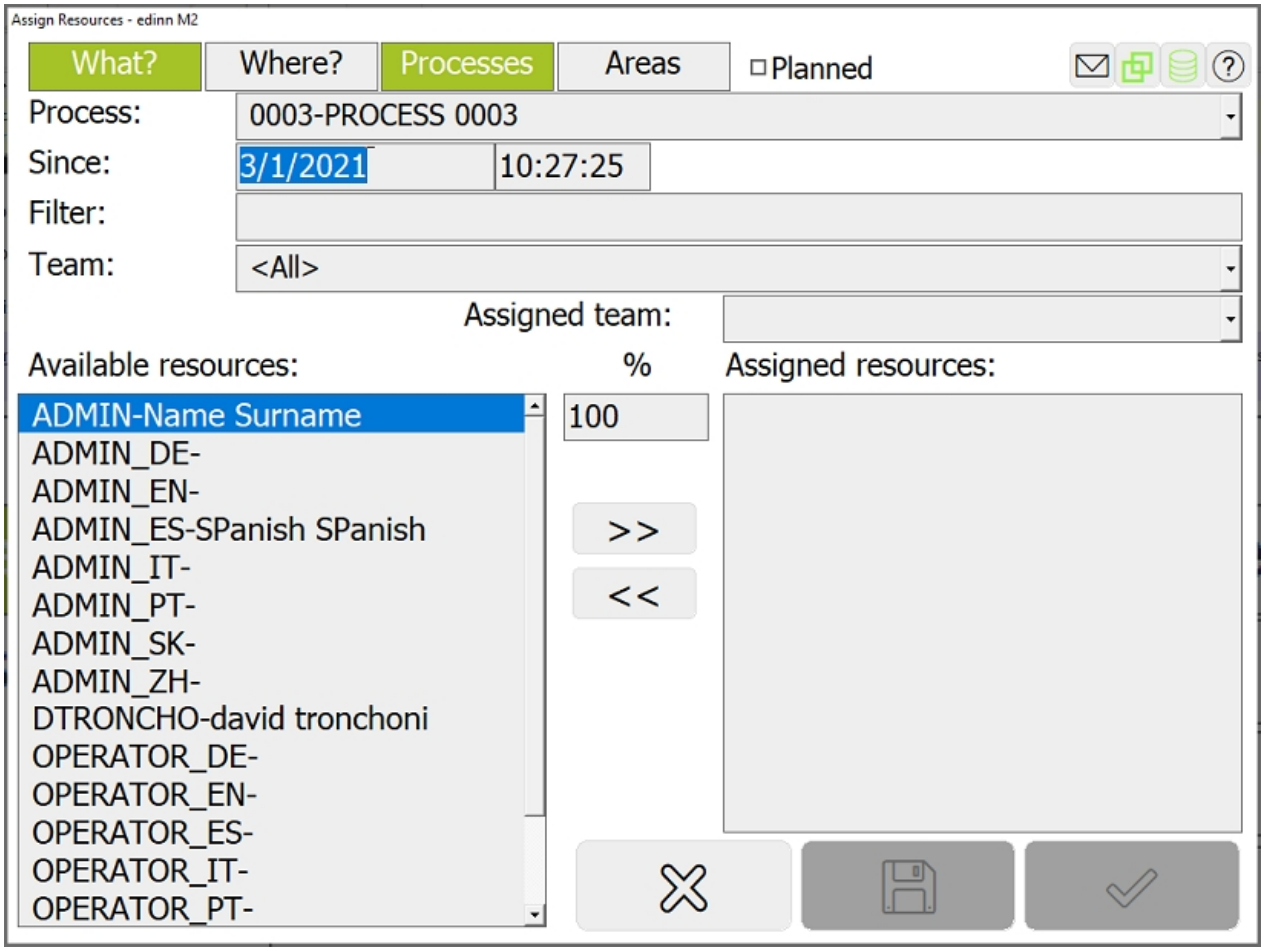
Main Window > In/Out button ( )

Utility: to indicate which persons or teams are working on which processes or areas. This window can only be accessed if the user has the In/Out role or the Multiprocess role.

You can also access this window from the Schedule window, but in this case only for the plan.

If the user is already working in this process, the **In/Out button ( )** will be shown in green.

> **NOTE:** on the process configuration window, administrators can configure the number of persons that are allowed to work in each process on each moment.If you try to IN to a process that does not allow more persons, the system will indicate this and pose different options.



At the top of the window, there are 4 selectors: What? Where? Processes and Areas. By combining them you can define:

- **What? + Processes:** which resources are working on a specific process.
- **What? + Areas:** which resources are working on a specific area.
- **Where? + Processes:** in which processes is working a specific resource.
- **Where? + Areas:** in which areas is working a specific resource.

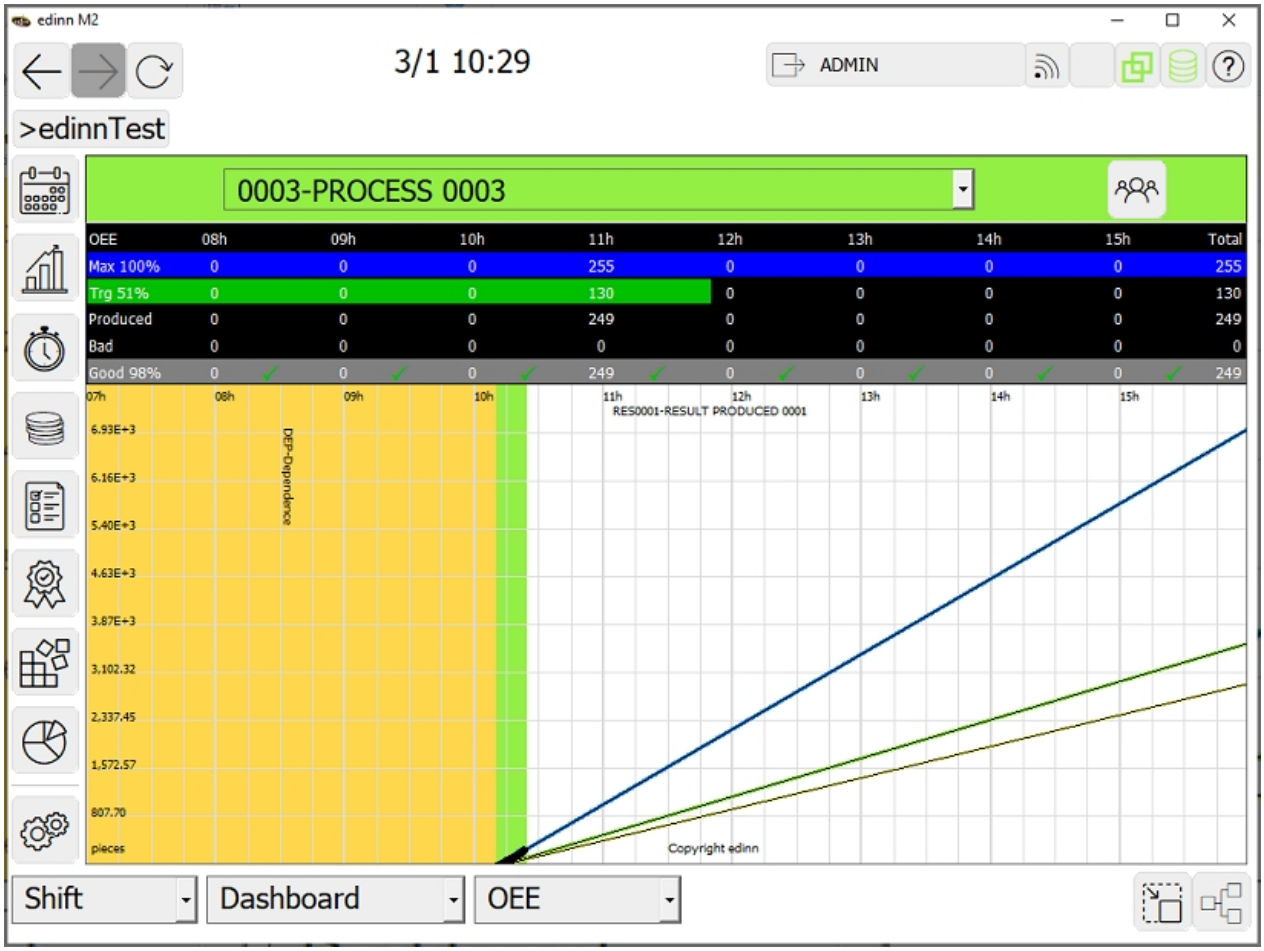
Detail is as follows:

- **Planned:** this field serves to store the assignment of resources as a planning. It is not active yet but it will be useful to compare the planning with the actual assignment.
- **Date and time**: on which the changes are going to affect. puedes poner turnos actuales, turnos pasados, y turnos futuros.
- **Filter**: allows filtering the Persons list.
- **Team**: choose the active team.
- **Available resources**: click the persons on the list that you want to indicate they are working in the process.

- %: activity percentage ratio.
- **Assigned resources:** to filter by teams.

Dashboard

The dashboard shows through a table and a graph the production data in real time, relating it to the OEE:

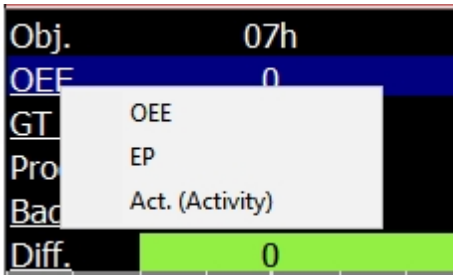


In the lower part the path of the process is shown. In the upper part the table shows different data classified by hours. The hours shown in the title of each column represent a time interval between the indicated time and the previous time (1 hour less). The fields shown by default are:

- **First row:** Pressing this button displays the menu with the following options:
  - OEE: Maximum quantity that can be produced at 100% of OEE.
  - EP: Maximum quantity that can be produced at 100%

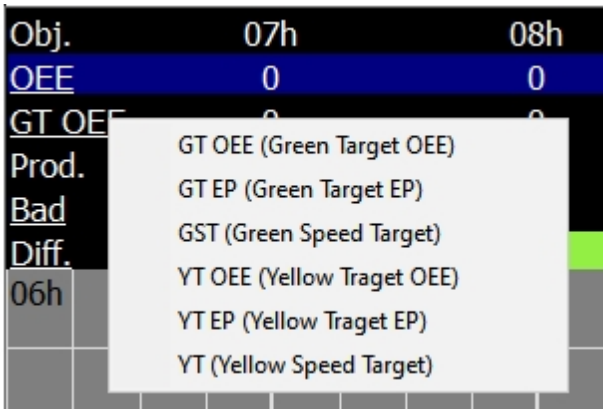
of EP (Productive efficiency).

- Act. (Activity): Maximum quantity that can be produced according to the activity ratio.

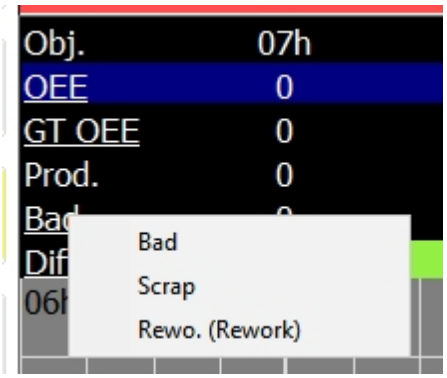


- **Second row:** Pressing this button brings up the menu with the options:

  - GT OEE (Green OEE Target): Green OEE target for that process. If the process has produced more than the amount specified in this field, it will be colored green.
  - GT EP (Green EP Target): Green EP target for that process. If the process has produced more than the amount specified in this field, it will be colored green.
  - GTS (Green Speed Target): Green speed target for that process. If the process has produced more than the amount specified in this field, it will be colored green.
  - YT OEE (Yellow OEE Target): Yellow speed target for that process. If the process has produced more than the amount specified in this field, it will be colored yellow.
  - YT EP (Yellow EP Target): Yellow speed target for that process. If the process has produced more than the amount specified in this field, it will be colored yellow.
  - YTS. (Yellow Speed Target): Yellow speed target for that process. If the process has produced more quantity than specified in this field, it will appear colored yellow.



- **Third row. Produced:** Amount produced. It includes good and bad quantities.
- **Fourth row:** When pressed, the menu appears with the following options:
  - Bad: Bad quantity produced, which is the sum of the amount of scrap plus the amount of rework.
  - Scrap: Amount of scrap.
  - Rework: Amount of rework.



- **Fifth row:** Pressing this button displays the menu with the following options:
  - Good: Good quantity produced.
  - Diff.: Difference from the target.

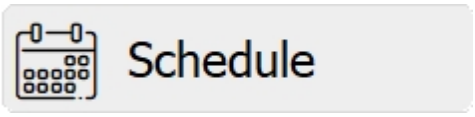
Additionally, through server configuration, the dashboard can be customized with the following options:

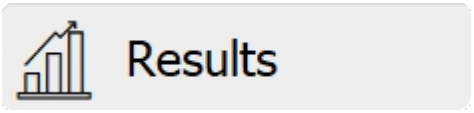
- **Accumulated**: With this option, instead of hourly data, the accumulated data from the beginning of the period is shown every hour and the total column does not appear.

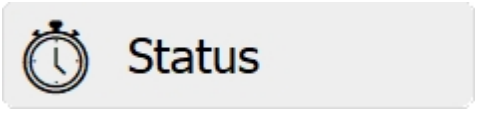
Main Buttons

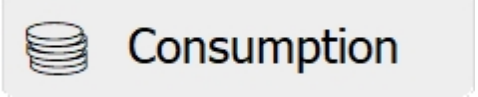
The buttons on the left allow us to visualize and manage the system.

> **NOTE:** to access you will have to choose first a process.

- 

: scheduled tasks and orders. If the button blinks means that an order is about to be delayed (yellow) or is definitely delayed (red).
- 

: results generated by the process. If the button blinks means that the user indicated that he was going to produce results in a certain time. That time is about to expire and the system has not detected any results.
- 

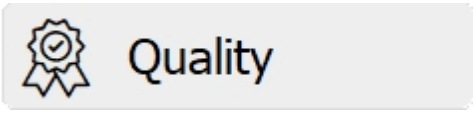
: statues by which the process has passed or is passing. If the button blinks means that there is a not yet justified failure.
- 

: consumptions performed

by the process. If the button blinks means that there is a not yet justified excessive consumption.

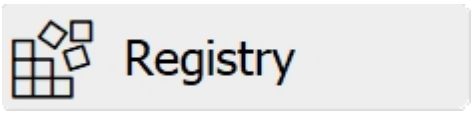
- 

: maintenance and procedure tasks. If the button blinks means that there are procedures or maintenance tasks to perform.

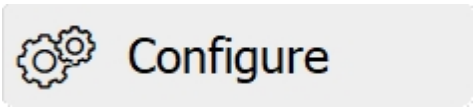
- 

: measurements to calculate the SPC (Statistical Process Control). If the button blinks means that the user must gather data from results in order to control quality (SPC).

NOTE: If you do not have the Quality license (SPC) active, the Quality button will flash when there are pending Quality-type Autocontrol tasks and when pressed, it will display the Autocontrol window filtered by Quality-type tasks, while pressing the Autocontrol button will show the Autocontrol window but excluding the Quality type tasks.

- 

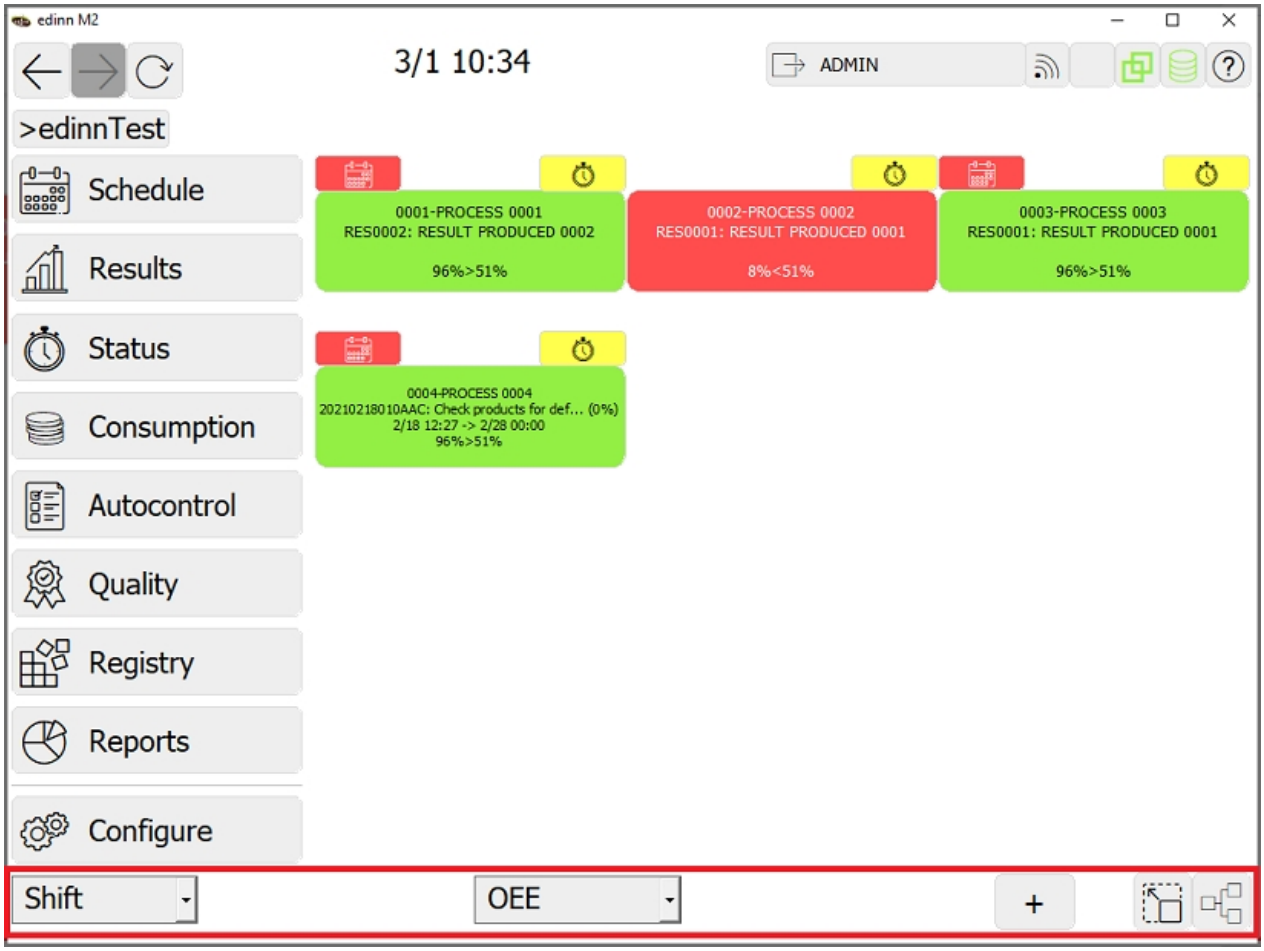
: different messages and events that occur in the system.

- 

: configuration of the system. Only available for administrators.

Filters and colors codification

The following will be shown on the display:



Selectors:

- **Period selector**:
    - **Actual** (as configured): shows the actual status that the process is going through, and how much time is in it. The colors divisions of the boxes of processes and areas shows the following:

	Production
	Idle times
	Stops or failures
	Not scheduled to work
	Not yet justified failure
	Dependence

    - **Rest of periods** (as configured): shows the aggregated status

in the period of the processes and areas. The colors divisions of the boxes of processes and areas shows the following:

	the process **has reached** all targets, according to the selected period and ratio
	the process **did not reach** the green targets, according to the selected period and ratio.
	the process **did not reach** the yellow targets, according to the selected period and ratio.

- **Type of visualization selector**: we can select between different types of visualization:
  - **Path**: shows the "Production Path" (internationally registered by edinn®). Indicates if the production process is being profitable, the scaled statuses through which the process has passed, the production targets, how much time is left before losing profitability, etc.
  - **Speed**: shows the graphic with the average production speed.
  - **Consumption**: shows the consumption. User must select the consumption type on the selector below.
  - **Path & Consumption**: shows the "Production Path", superposing the real and theoretic consumption graphic.
  - **CO2**: shows the graphic of generation of equivalent CO2.
  - **Cost**: shows the graphic of generation of costs.
  - **Ratios**: shows the ratio table, with the target ratios and values in red which are below the target.
  - **MTBF**: shows the table of the Mean Time Between Failures (MTBF).
  - **MTBFP**: shows the table of the Mean Time Between Failures (MTBF) for a specific product.
  - **Control**: shows the control graphic of the SPC or Statistical Process Control.
  - **Histogram**: shows the histogram graphic of the SPC or Statistical Process Control.
  - **In&Out**: shows the entrance (IN) and exit (OUT) of working personnel into the process.

  - **Team In&Out**: shows the entrance (IN) and exit (OUT) of working teams into the process.
  - **Dashboard:** customizable scorecard for each client.
  - **Reports Dashboard:** customizable scorecard by device and process. This scorecard can be configured and assigned from the reporting tool. Clicking with the mouse on the control panel will open it in the browser, which will allow us to consult the reports contained in it.

- **Ratios selector OEE, PE, MTBF, etc. / Measurement (SPC) selector / Type of consumption selector**: indicates to which ratio refer the colors of the “boxes” or which variable (SPC or consumption) we are considering.
- **Button "+":** Allows you to add an area (on the areas view) or assign a process to an area (on the processes view).

NOTE: some of these buttons buttons might not be accesible if you do not have the right role or if your organization does not have the right software license.

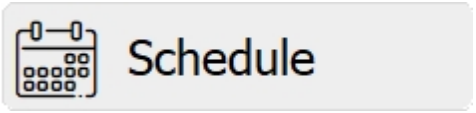
Schedule

Main Window > Schedule

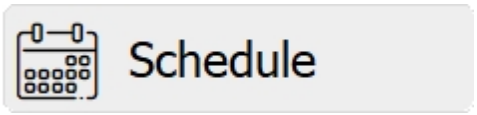
edinn® M2 has a scheduling module. This module allows:

  - To indicate, for each process, the results that need to be produced, its target quantity, its specified periods, etc.
  - That the working user knows the next order to be produced, what is the target quantity, what period, etc.
  - To know, in real time, the produced quantity compared to the scheduled quantity.
  - That the working user can automatically do a result change by starting a scheduled order.

The view schedule window shows the work orders of processes and areas. The view schedule window can be accessed from 2 origins:

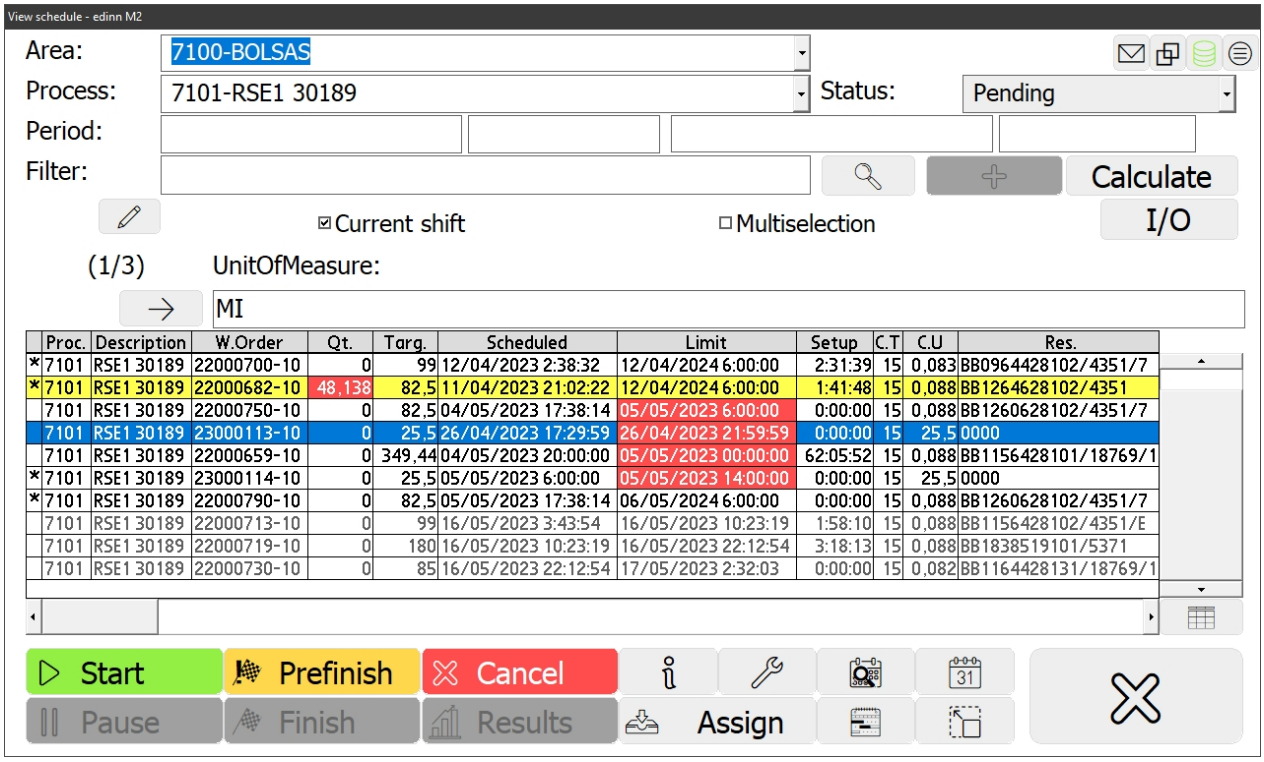
  - From the 

button on the main window.

- From the 

button on the Results Window.

The detail of the window is as follows:



Note that the **columns** of the table can be **hidden** or **shown** per user through the **icon** which will be shown in the **top right corner of the table** and in the Resources configuration.

Selectors and text boxes:

- **Area**: select the area you want to see its work load.

NOTE: Archived areas will not appear in the selector. To learn more about archiving areas see area configuration.

- **Process**: specify for which process you want to see its work load.
- **Status**: orders can be in different status. This selector filters the orders to be showed.
- **Since** (start date and time and end date and time): type the period to search the orders.
- **Filter**: text that will filter which orders will be shown. This filter will be applied to: Order Id, Results Id, Comments, 'Ref' field and Description of result.
- **Current shift**: in the table view, it shows the orders planned in future shifts with the text in gray color.

- **Histogram**: in the gantt view, shows the histogram of consumptions of the selected result.
- **Multiselection**: allows to select multiple orders and to modify them all simultaneously.

Buttons:

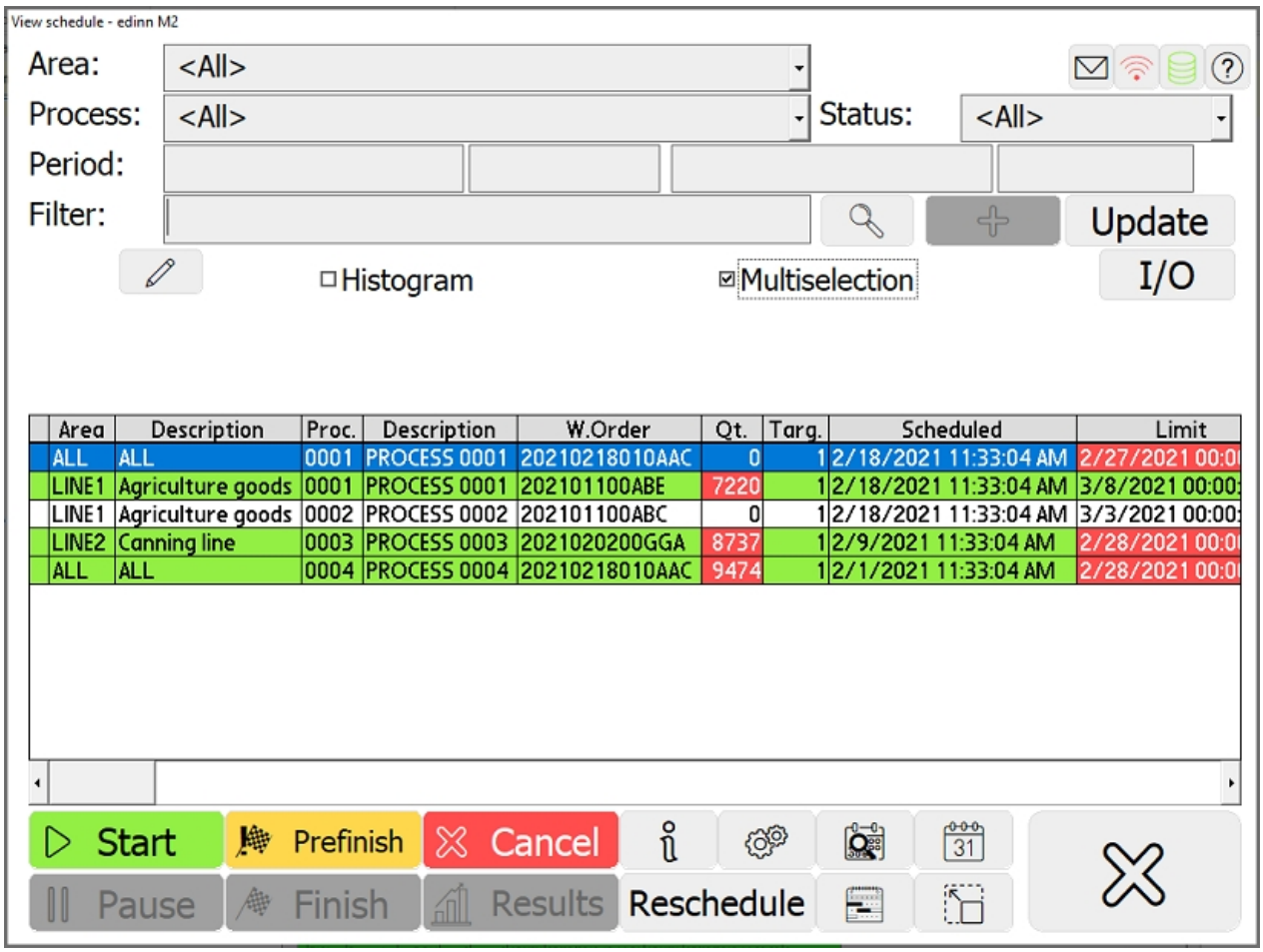
-  **Search**: updates the list of orders.
- **Update button:** recalculates the quantities produced in one order. There must be one selected.
-  **Comment:** Allows you to add a comment to an order or show a previously added comment.
- **I/O button:** goes to the Input/Output window which shows the inputs and outputs of the order.
- **Start button:** starts the order. The process is starting to work for the selected order.
- **Prefinish button:** the selected order is considered finished, although some checks should be performed to confirm this and pass the order to definitely finished, with the 'Finish' button.
- **Cancel button:** the selected order is canceled.
- **Information (i) button**: shows all the information of the selected order.
- **Edit button:** opens the window to change the schedule.
- **Pause button:** the selected order is going to be paused.
- **Finish button:** the selected order is considered definitely finished.
- **Assign button:** the selected order will be parcially or totally transferred to another process.
-  **Gantt**: view the Gantt chart:
-  **Maximize button:** allows to expand or collapse the data area of the window in order to show more or less orders.
- **Results button**: it allows access to the production records belonging to the work period of the order. If we had activated the Order detail button, and there would be inconsistencies, then it will show them.
-  **Order detail**: shows the different periods in which the work order has been

in progress and the states through which it has passed. For example, when and in what process was initiated, when it was paused, if it was assigned to another process, etc.

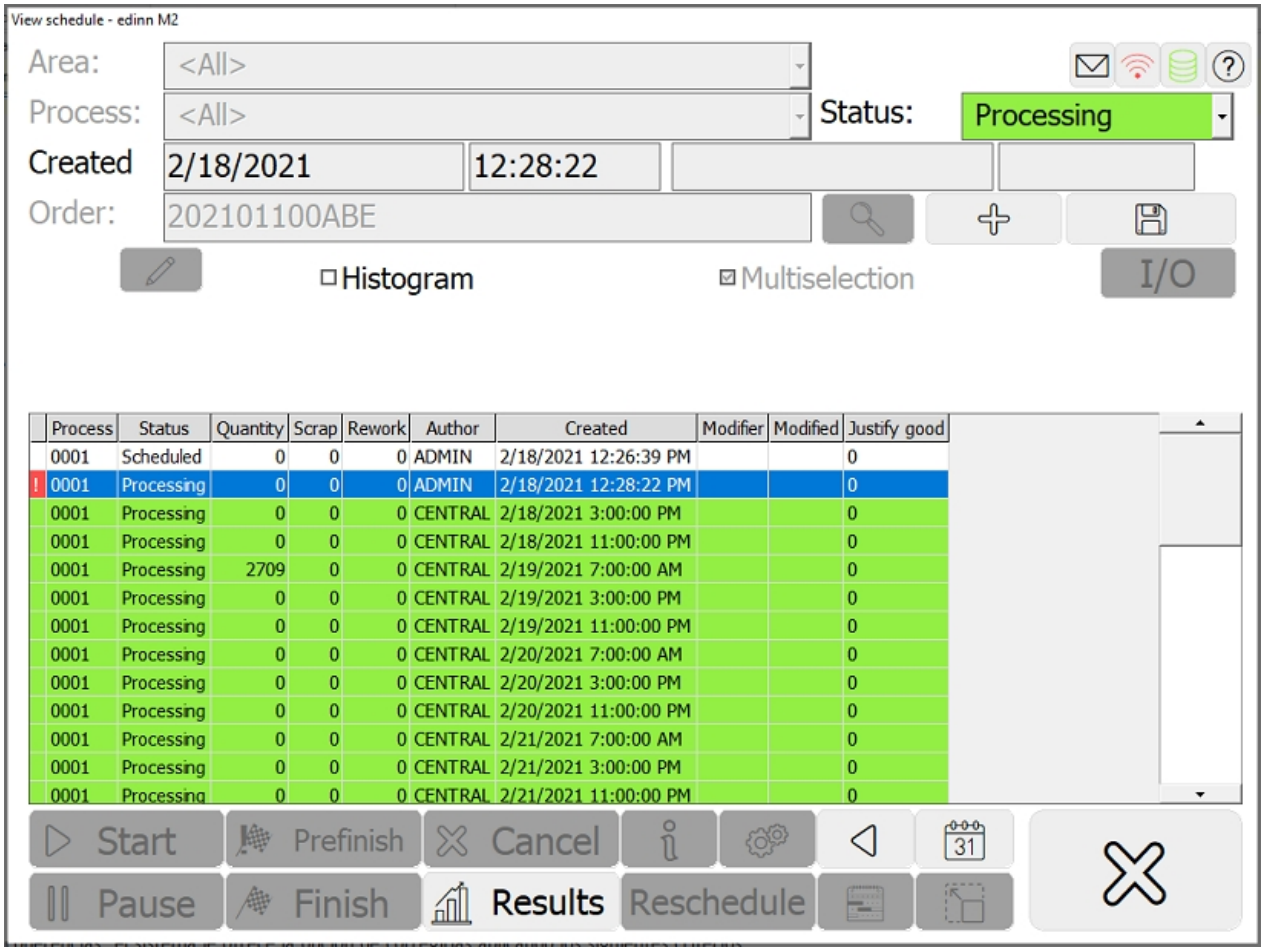
NOTE: You can use the CTRL key to select a range of records in the table, like it is normally done with the SHIFT key.

Let us provide further explanation about the Order detail functionality.

There may be inconsistencies between production records and work orders, for example productions that are not assigned to any order or assigned to an order different than the one that was started. The orders with inconsistencies will be indicated with the red background in the box of the quantity produced.

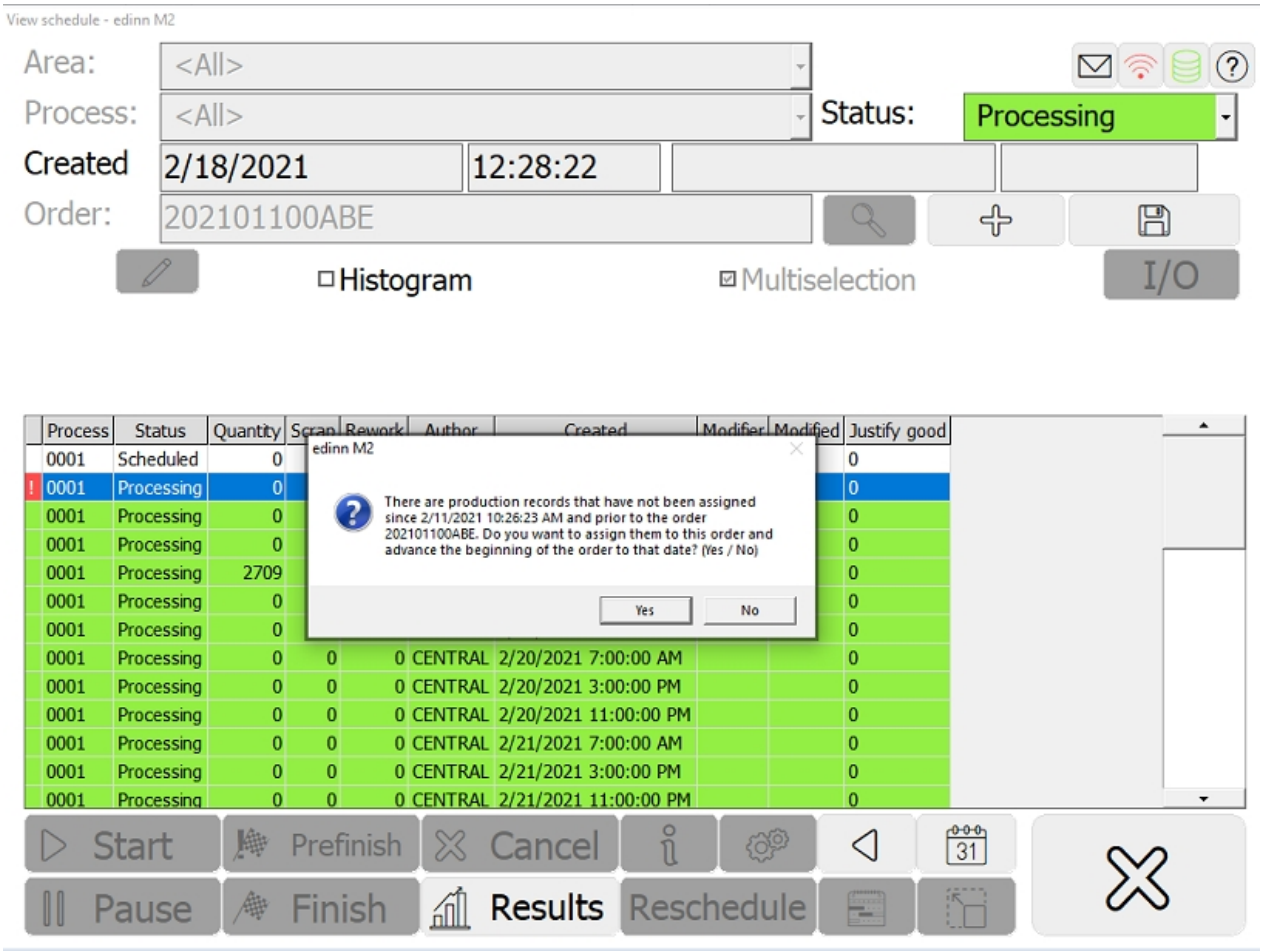


The periods of the work order where inconsistencies have been detected will be indicated by the red background in the first cell of the table and an exclamation mark.



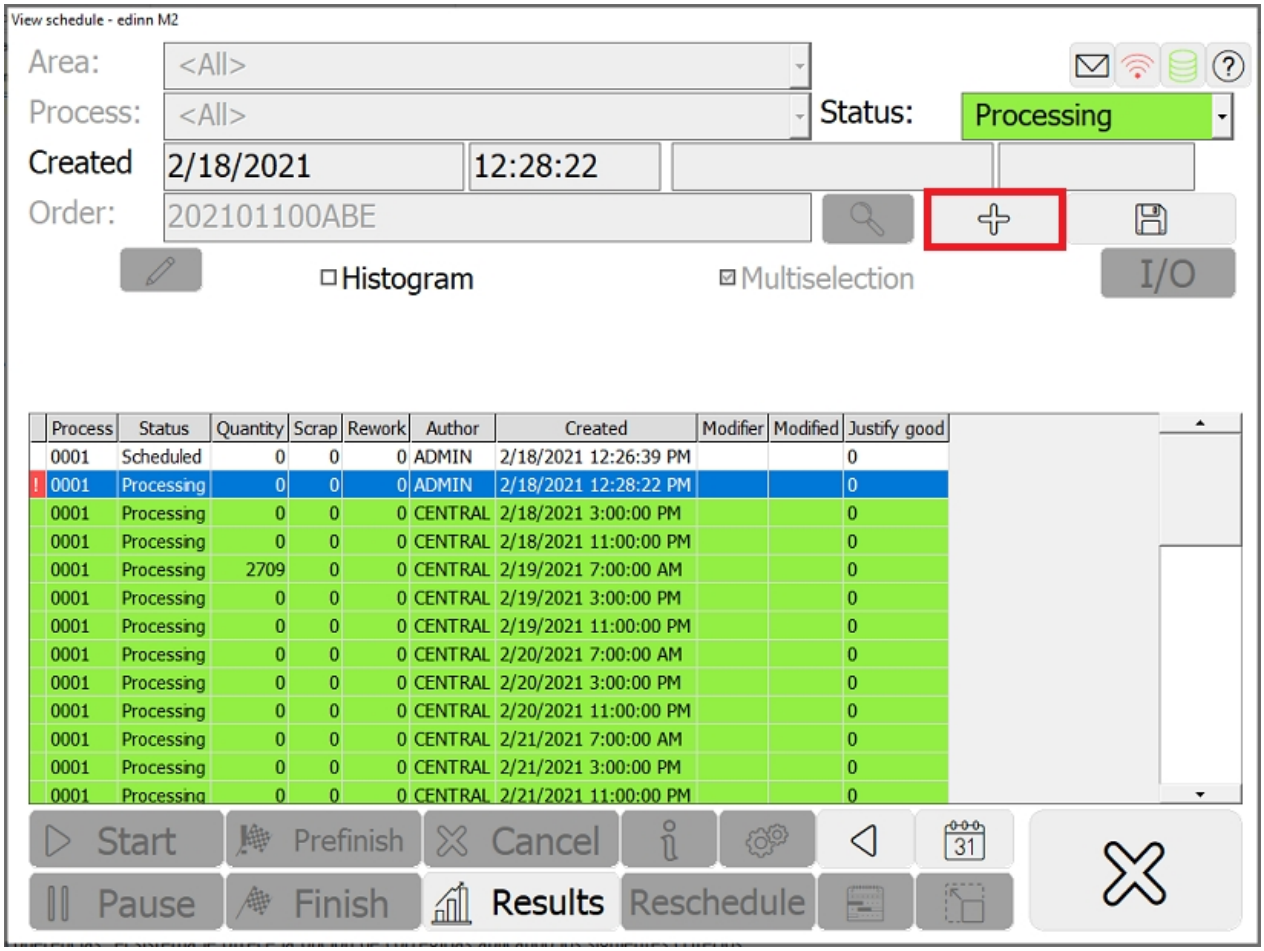
Selecting the period with inconsistencies, the system offers you the option to correct them by applying the following criteria:

1. If there are unassigned records to a order, just before the start of one, these records will be assigned to that order and its start time will be advanced.
2. If there are records assigned to a order, just before the beginning of said order, its start time will be advanced.
3. If there are records without assignment to any order, during the period of execution of one, records will be assigned to said order.
4. If there are records assigned to an order in the execution period of a second order, records will be assigned to the second order.

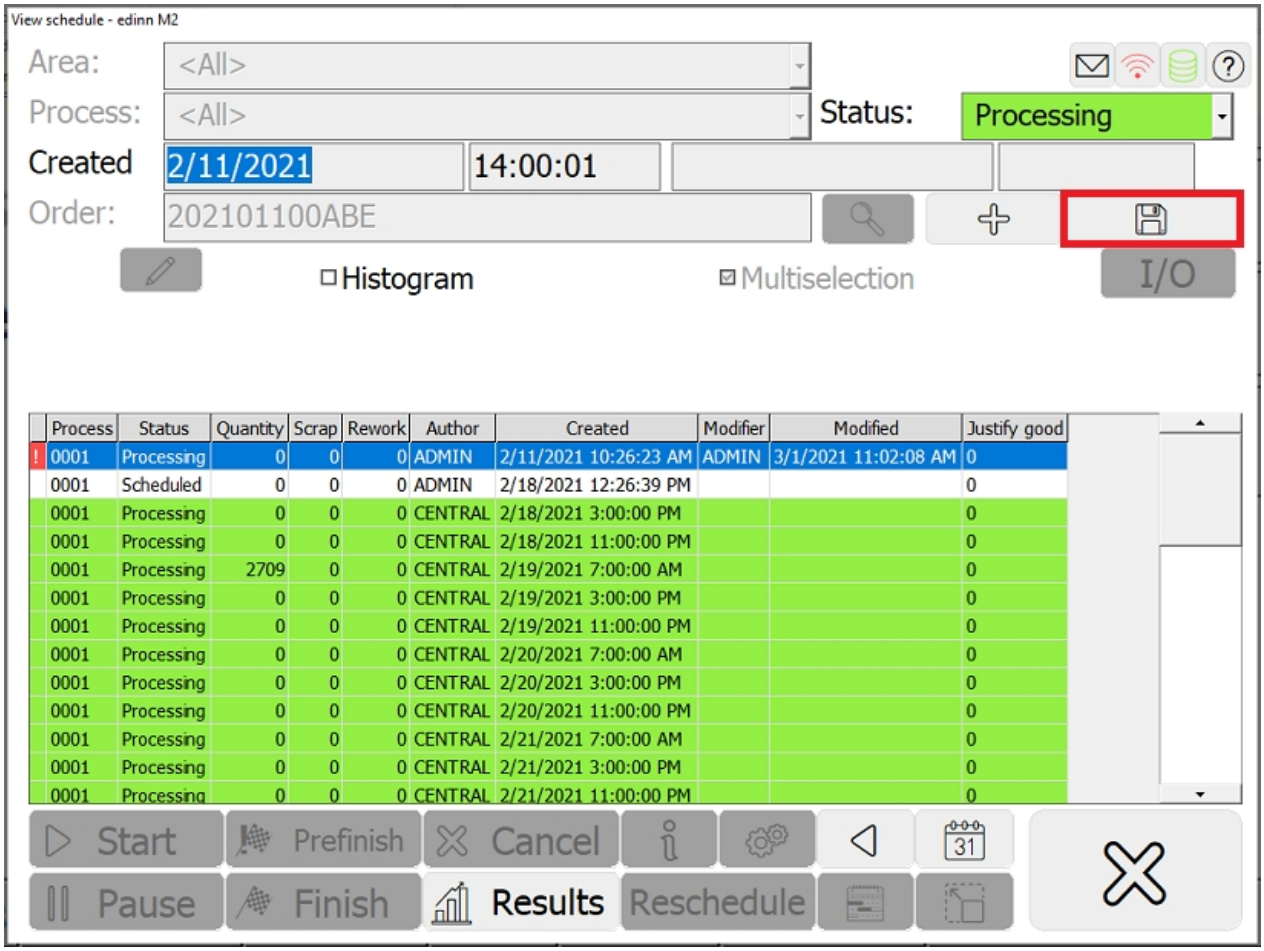


You can insert a new period between two others to resolve inconsistencies. For example, if the user did not restart an order after pausing it, you can insert an order reset by entering the date, time and status of the order (for example, in process) and clicking the "+" button (more).

This functionality is restricted by these conditions: you can only insert periods between two others. You can only insert periods with status in progress or paused. You can not insert a status equal to the previous or the next.

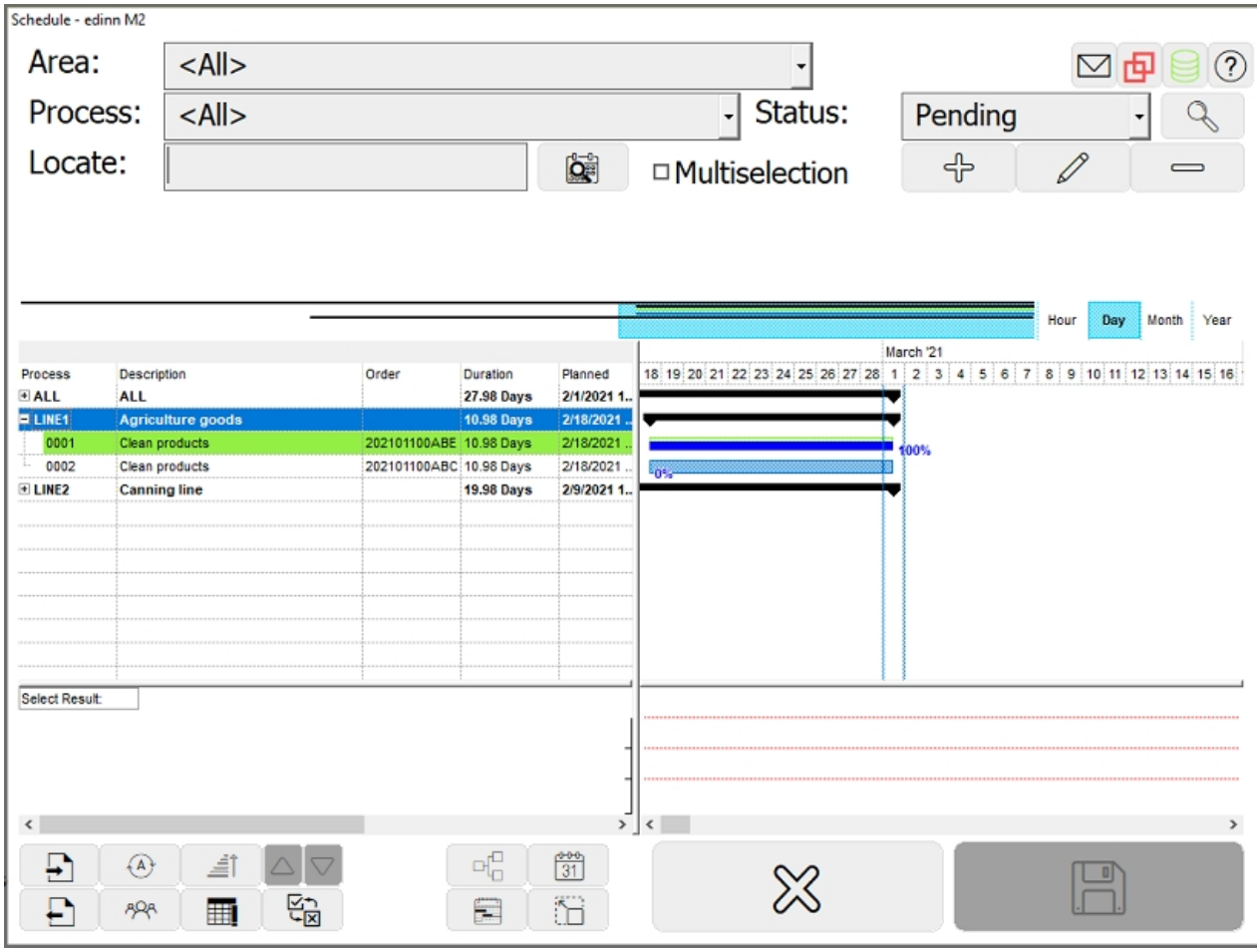


You can modify the time a period starts or the status. To do this, select the record you want to change, enter the new date and time, or the status you want to assign, and click save. This functionality is restricted to the conditions already mentioned.



Edit schedule

Main Window > Schedule > Edit



Detail is as follows:

- **Area and Process**: selects the area and process for which we want to see the scheduled orders.

NOTE: Archived areas will not appear in the selector. To learn more about archiving areas see area configuration.

- **Status**: allows to select the status to search.
- **Search**: refreshes the data grid.
- **Locate**: locates in the table or gantt.

Buttons:

- imports new work orders from the clipboard, from project files (.xml, .bc3) or Excel (.csv).

Note regarding import from CSV

The mandatory fields (ordered according to the Excel columns) are: Area, Proc., W.Order, Qt, Targ.,

Scheduled, Limit, Setup, C.T*, C.U*, Teardown, Res., Calc, Rec.
*In C.T and C.U if the entered value is 0 , it will inherit the values configured in PSR Relations.

In the "Predecessors" and "Relation" fields, enter the working orders codes (separated by commas) with which you want to relate the working order and the relation between them, respectively.
The relations types are:
0- Finish-Finish
1- Finish-Start
2- Start-Finish
3- Start-Start
4- None

-  exports work orders to Excel (.csv)
-  automatically schedules the selected orders. First performs an adjustment based on their dependencies and second, only for the tasks/orders in pre-scheduled status, an advanced schedule. Please consider also that the calendar for the availability of the processes is indicated by scheduling statuses. For this, please review the notes and warnings of statuses configuration.

Note regarding priority

The automatic scheduler will prioritize plans which:

1st) Have less delay, considering the scheduled date and times with respect to their limit date and time.
2nd) Are finished sooner, with respect to the current moment.
3rd) Have a lower WBS field, according to their alphanumeric order. Therefore, you can use the WBS field to prioritize between clients or other criteria.

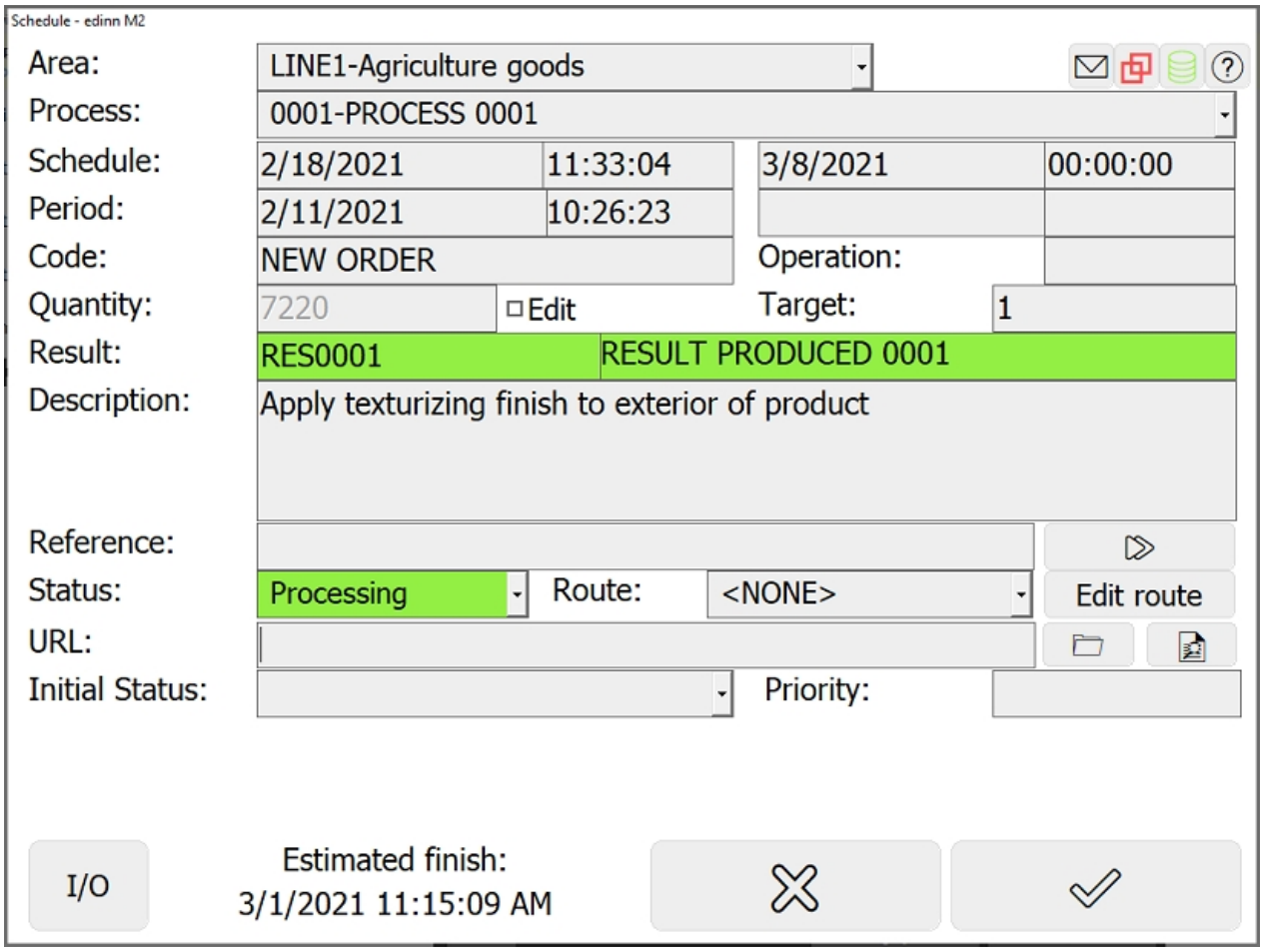
For example, if you have clients type A, B, C, etc.: you could the A in the WBS field of those orders of client A, and so forth, and for production plans where the 1st and 2nd previous criteria are equal, then the scheduler will prioritize plans of client A versus B.

-  changes the priority (WBS) of the selected orders.
-  changes the priority (WBS) of one order based on its order in the table grid.

Note: This function can be used in table view and only if the table grid is ordered according to the "priority" column.

-  assigns resources. For more information, please see: "Assign personnel and team window".
-  shows a table with the issues of the orders.

-  changes the status of the selected orders.
-  shows the processes in a tree structure for a faster access.
-  shows the calendar calculated by the system based on the statuses configuration or the statuses which really occured.
-  shows the schedule in a gantt chart or  table grid.
-  maximizes the table or the gantt.
- Press  to create a new order:



- **Area**: select the area to which the order will be assigned.
- **Process**: optionally, select the process, inside the selected area, which will perform the task/order.

Note: until an order / task is assigned to a process, there will not be any alarms about delays in the

win32 terminal.

- **Schedule**: date and time to start and to end (**limit**) of the order.
- **Period**: real period in which the order has started and finished.
- **Order**: id of the order. By means of an option within the server configuration, this field can be automatically dumped to the order field of the result when the scheduled order is started.
- **Operation**: id of the step inside the ordern
- **Quantity**: results produced until now.
- **Edit**: indicates if the QT field should be updated by clicking on accept in this window. If not, it will not be modified. This is to avoid, if the order is being processed, that when changing something in the order, we mistakenly change the produced quantity until now.
- **Target**: quantity that should be produced.
- **Result**: result code to produce.
- **Description and Reference**: free texts.
- **Status of the order**:
  - **Prescheduled**: the order (not the work indicated by the order) is being prepared.
  - **Scheduled**: scheduled to be produced.
  - **Processing**: being processed.
  - **Paused**: paused.
  - **Prefinished**: finished, but waiting for approval to pass to finished.
  - **Finished**: finished.
  - **Canceled**: canceled or will not be started or contined.
- **Status (time usage)**: used to read the corresponding fields of the process-status-result relation. It can be configured to be requested to the user. The list of statuses comes from the PSR relation.
- **Route**: the following processes through which the result will go, or if the user tries to assign the order. Please see routes configuration. You can choose a route from the list of templates of routes and it will be copied to as the route of this order. You can then change the route by using the button **Route**, explained below.

- **Edit Route:** to configure a specific route for the order.

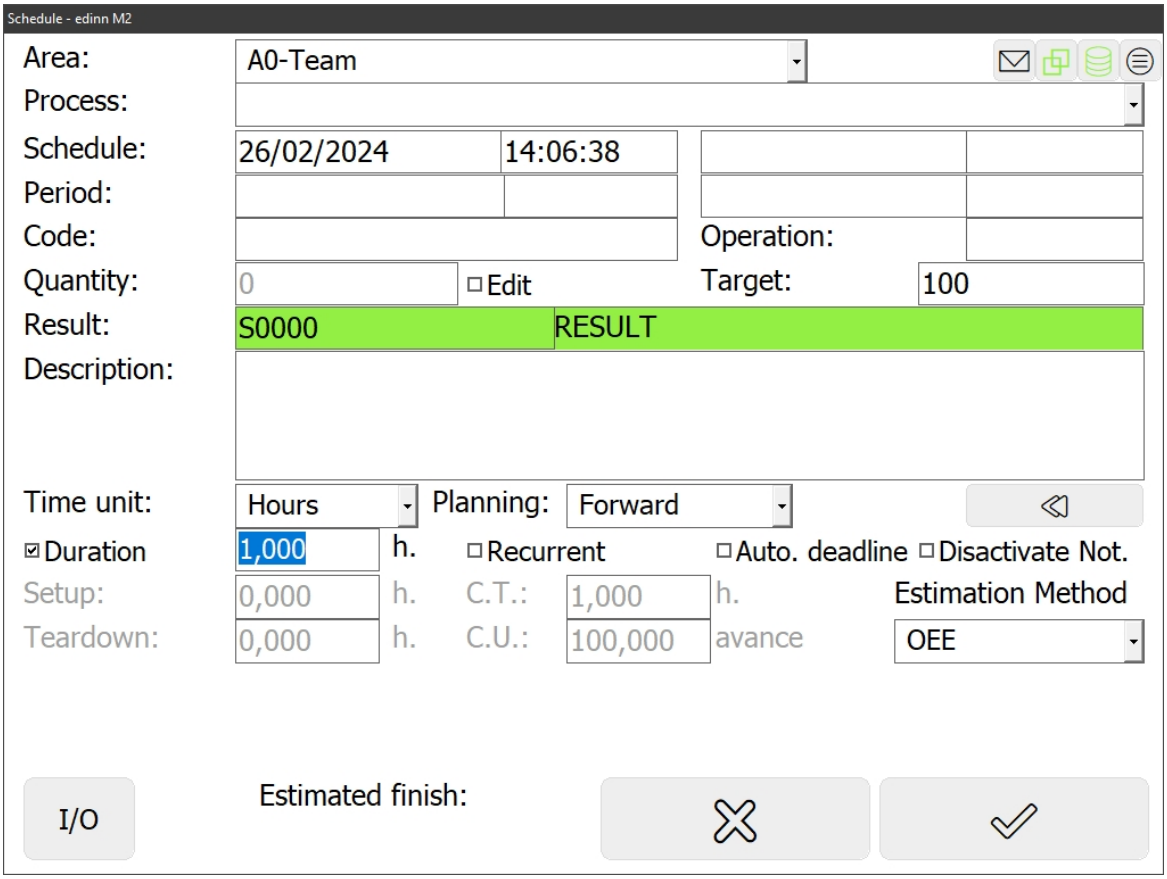
Note regarding the routes: When saving the changes of this order, you will be asked to automatically generate the operations of the route. If answered yes, the system will generate an order with different operation code, according to the route, for each process indicated in the route.

If inside a route, in the step of an operation, a process has been selected, then the automatic scheduler (see buttons above) will take as possible processes all the processes which have that same result as PSR Relation.

If inside a route, in the step of an operation, an area has been selected, then:

1) For the creation of the operations (steps of the route), the system will select the first process of that area which is not in an unscheduled status.
2) The automatic scheduler (see buttons above) will take as possible processes for that operation only those which belong to that particular area and which have that result as PSR Relation.

- **URL**: document or application that will be opened when the user clicks on the  that appears after having clicked on the order, on the view schedule window. Please see how to configure URLs. URLs can be selected from  and its syntax can be cheched from the button  .
- **Initial status**: the process will automatically enter in this status when the order will be initiated.
- **Priority (WBS)**: value to prioritize the orders.



- **Time unit:** Indicates the unit of the time fields on this form from the following: seconds, minutes, hours, and days.
- **Planning**: if this order should be automatically scheduled forward of backward.
- **Recurrent**: indicates if the order is recurring. If marked, when starting the order, if the quantity if greater than the target, then it will understand that it must restart the order and will put the quantity to 0. All recurrent tasks have also marked the Auto. deadline mark.
- **Auto. deadline**: indicates if deadline date must be calculated automatically. If marked, deadline date will be calculated continuously, based on the times of the order, while the order in in a pending status and, as the last time, when the order is started.
- **Deactivate notif.**: prevents production to be notified on this order. This is useful when there are more than one operation and you want the initial operations to be only to report inputs and outputs, and only in the last operation report results. In this example we would mark this in all operations except in the last one.
- Time parameters. Any change in these fields will automatically update the limit date and time seen above:
  - **Duration:** Indicates the total duration of the order, if checked, the rest of the time parameters will adapt to the time indicated in this box. If not checked, this box will be the one that will adapt to the values entered in

the rest.

- **Setup**: expected time in seconds to prepare the order.
- **C.T.** (Cycle Time): time in seconds per cycle of production. Please see Maximum speed.
- **C.U.** (Cycle Units): number of units expected per cycle of production. Please see Maximum speed.
- **Teardown:** expected time in seconds to dismantle the order.

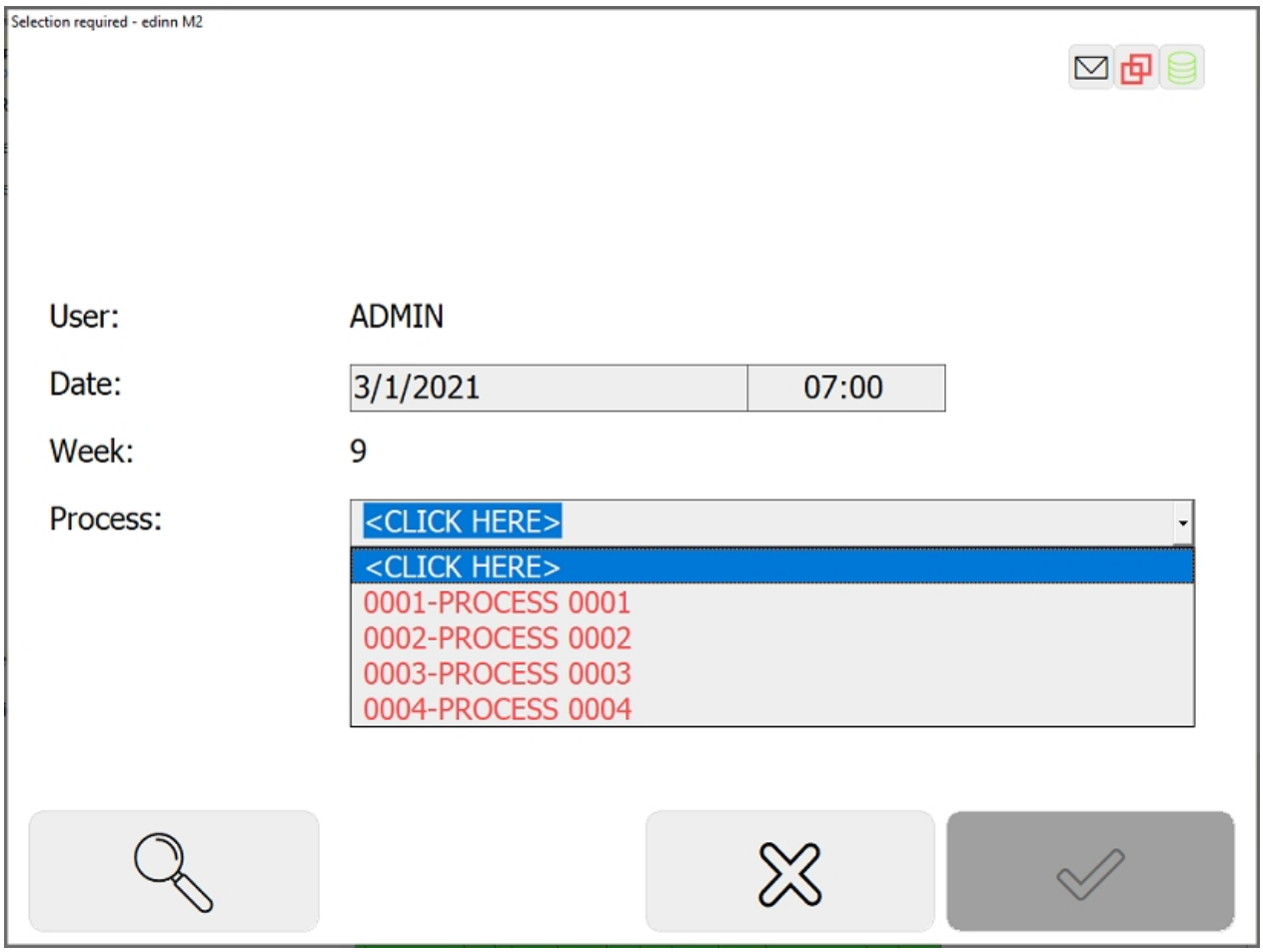
- **Estimation method:** Indicates the criteria to establish the estimated time. It can be by OEE or by nominal capacity.
- **I/O button**: for configuration of the expected inputs and outputs of the order. Please see *inputs and outputs window*.

Results

Main Window > Results

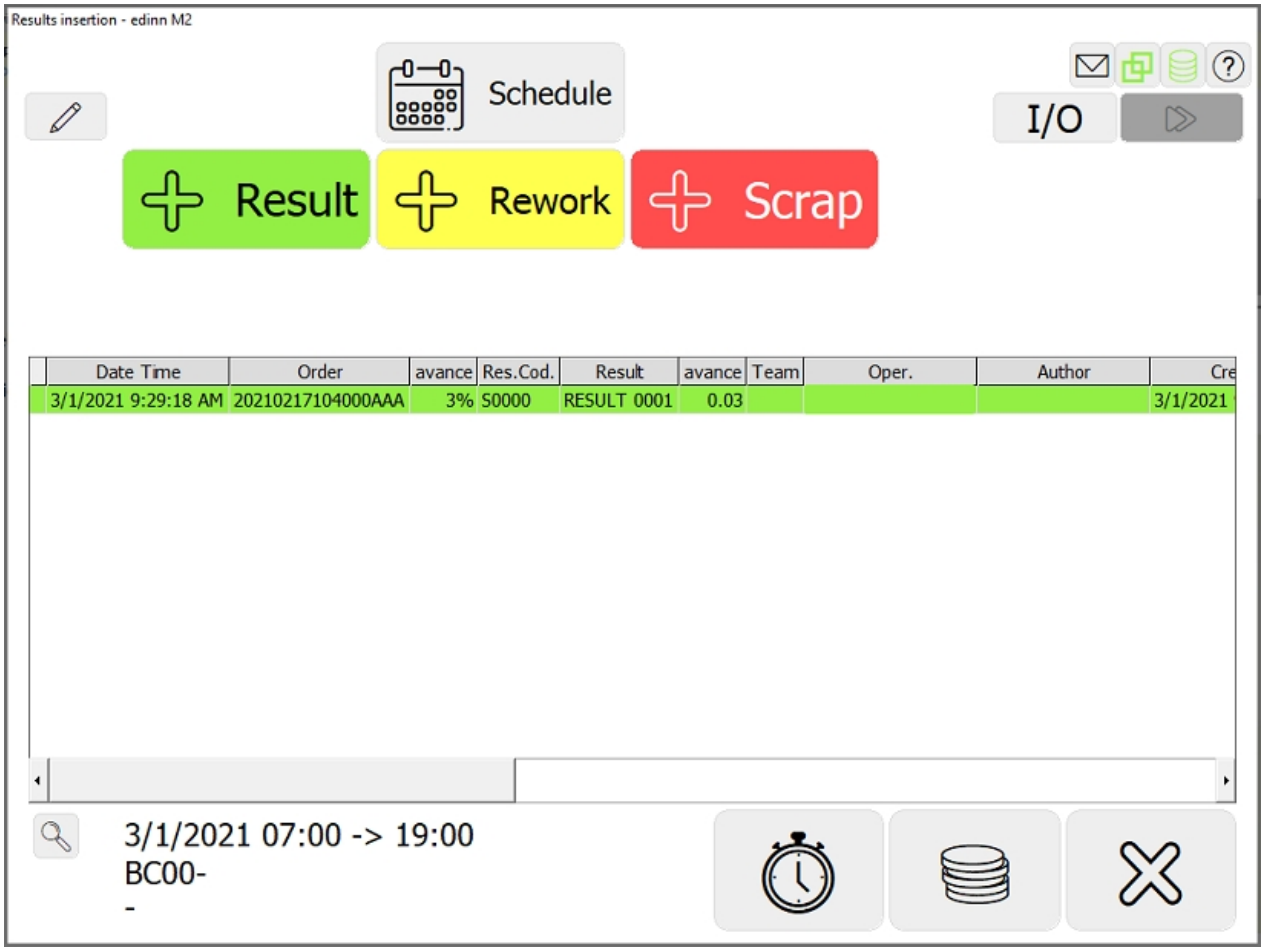
Utility: to view and indicate the various results a process has generated, either manually or automatically.

If the process is not selected before clicking on the Results button, a selection window will be shown:



The user who has the Multishift role will be able to "Locate" any data of any other period (button at the bottom left) and change the date and time.

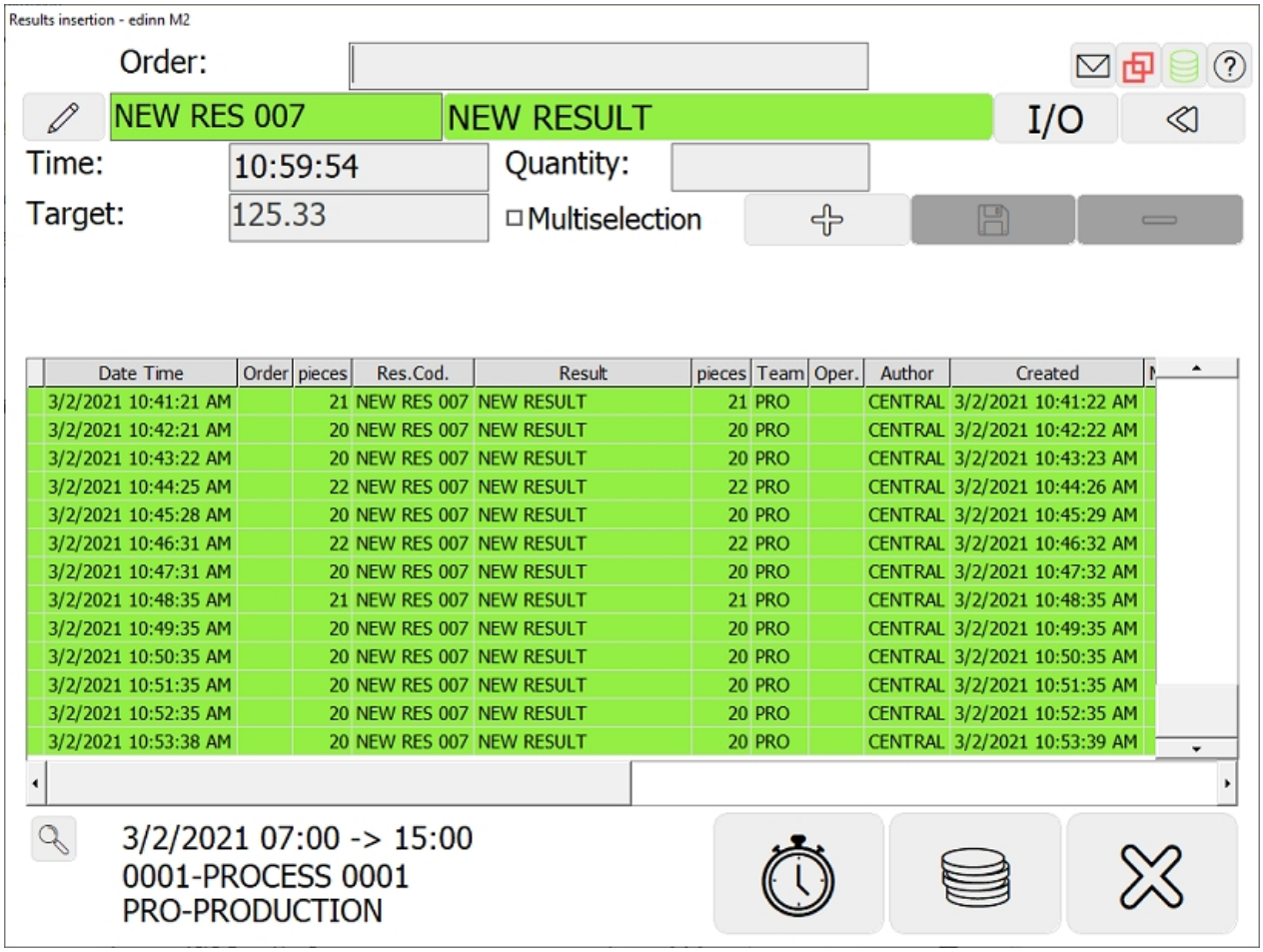
Once we have selected the desired process, we have to click on the button  . The following window will be displayed:



This window has the following buttons:

- **New**: to change a result. From this moment on, the system expects a different reference of result from this process.
- **Schedule**: it takes us to the Schedule window. For more information, please see: View schedule.
- **View paused**: allows to view the paused results.
- **I/O**: takes us to the *Input/Output window* which allows us to view and manage the inputs (elements consumed by the process) and outputs (elements generated by the process).
  - : allows to go to the advanced editing functions. The fields of the window are:
  - **Order**: the order the process is working on.
  - **Reference of result fields**: the code and description of the result the process is producing.
  - **Time**: hour or time.
  - **Quantity**: quantity.

- **Target**: maximum target quantity to be produced, considering only the production time between the last record and the indicated time.
- **Multiselection check**: allows to select multiple lines in grid.



  - **Insert Good, Insert Rework, insert Scrap**: these buttons only appear when we have the necessary role. These buttons are used to indicate that a new result has been produced (in the case of the green button), or to indicate that a previously generated result, has been confirmed to need rework (because of quality reasons) or has been discarded (scrap, because of quality reasons).

At the columns of the grid, at the first column, a " * " symbol indicates there is a user comment for that record. You will have to double click in order to go to the comment.

By clicking on the lower left box you will go to the period and process selection window explained at the top of this document.

NOTE: You can use the CTRL key to select a range of records in the table, like it is normally done with the SHIFT key.

WARNING: Some functions, like for example deleting production registers of which you are not the author, or changing the date times, or type of product, can only be used if the administrator assigned you these roles in the personnel configuration.

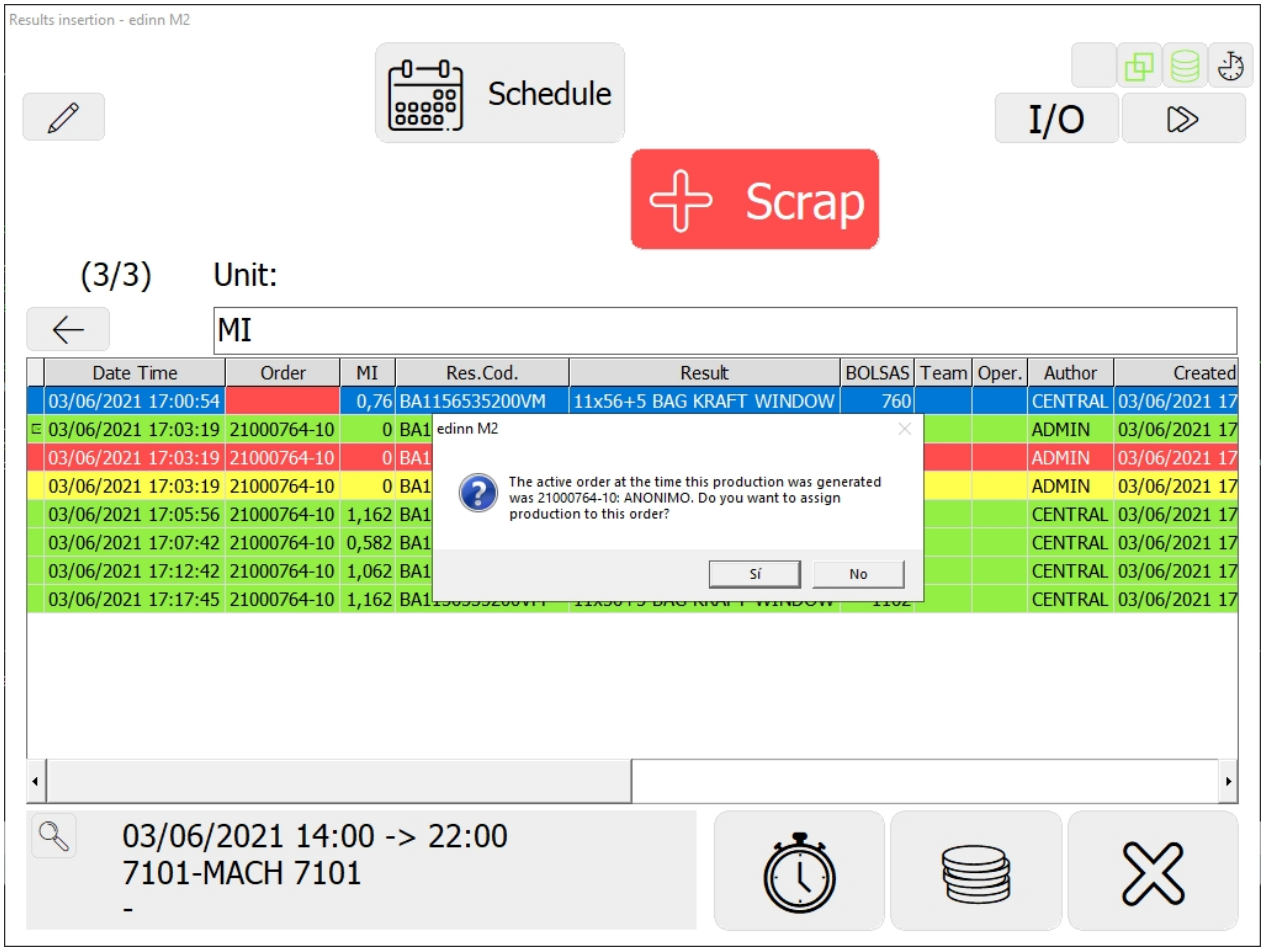
Results with schedule

Additional information about the results window when working with planning.

When the production of the process is limited to planned work orders, there must be no productions without an assigned order, so when a production does not have an order, the corresponding cell appears in red even though the row is selected.

If the cell is clicked, the system searches for the order that was active at that moment, shows it to the user and asks if they want to assign that order to the production record. If the user answers "yes", the system will modify the registry.

This function is complementary to the one explained in the planning window and requires the user to have permissions to modify a production record.



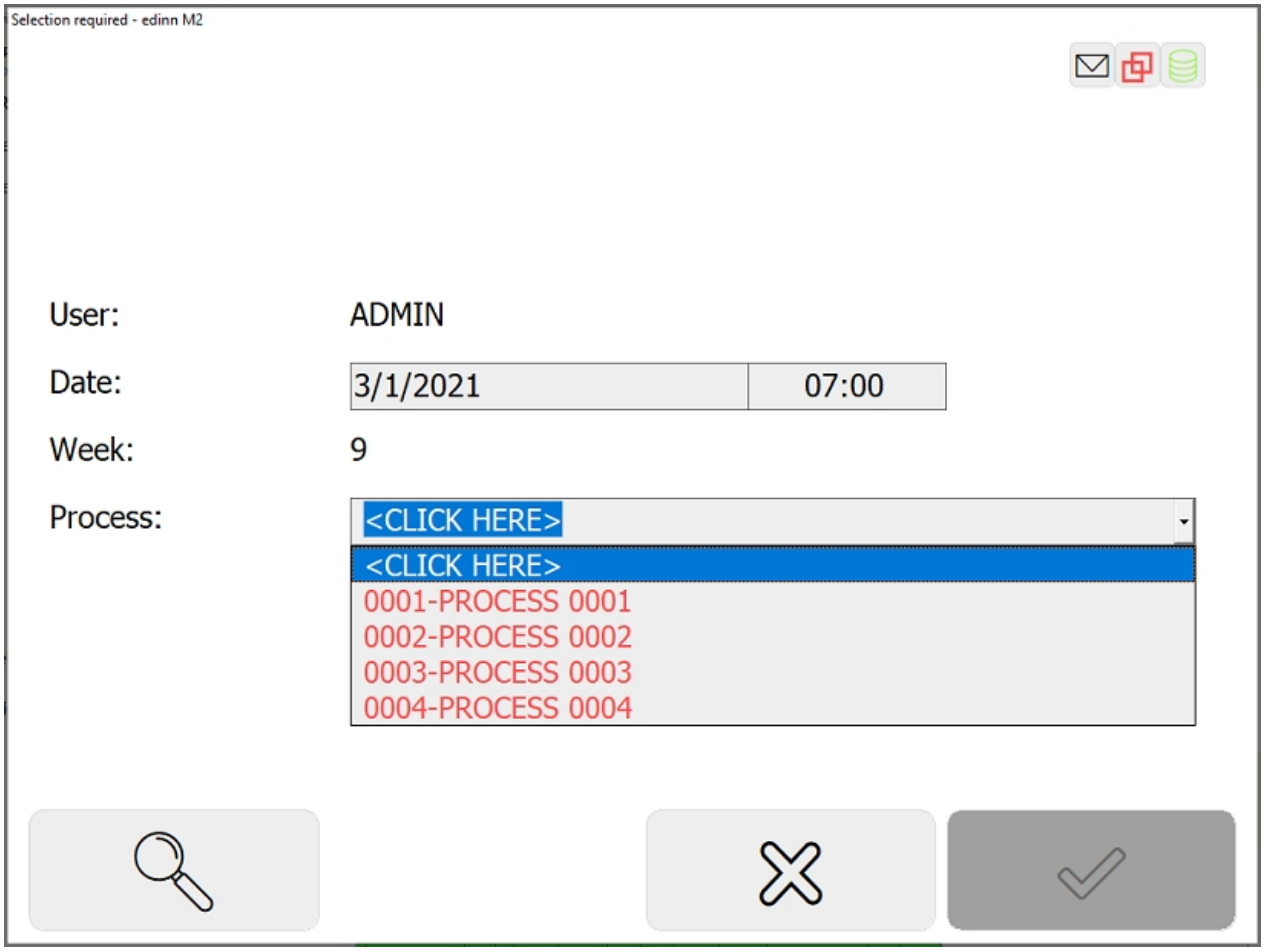
Procedures

How to insert a result

This procedure details the steps to insert a new result or production. To insert scrap you can also view this video.

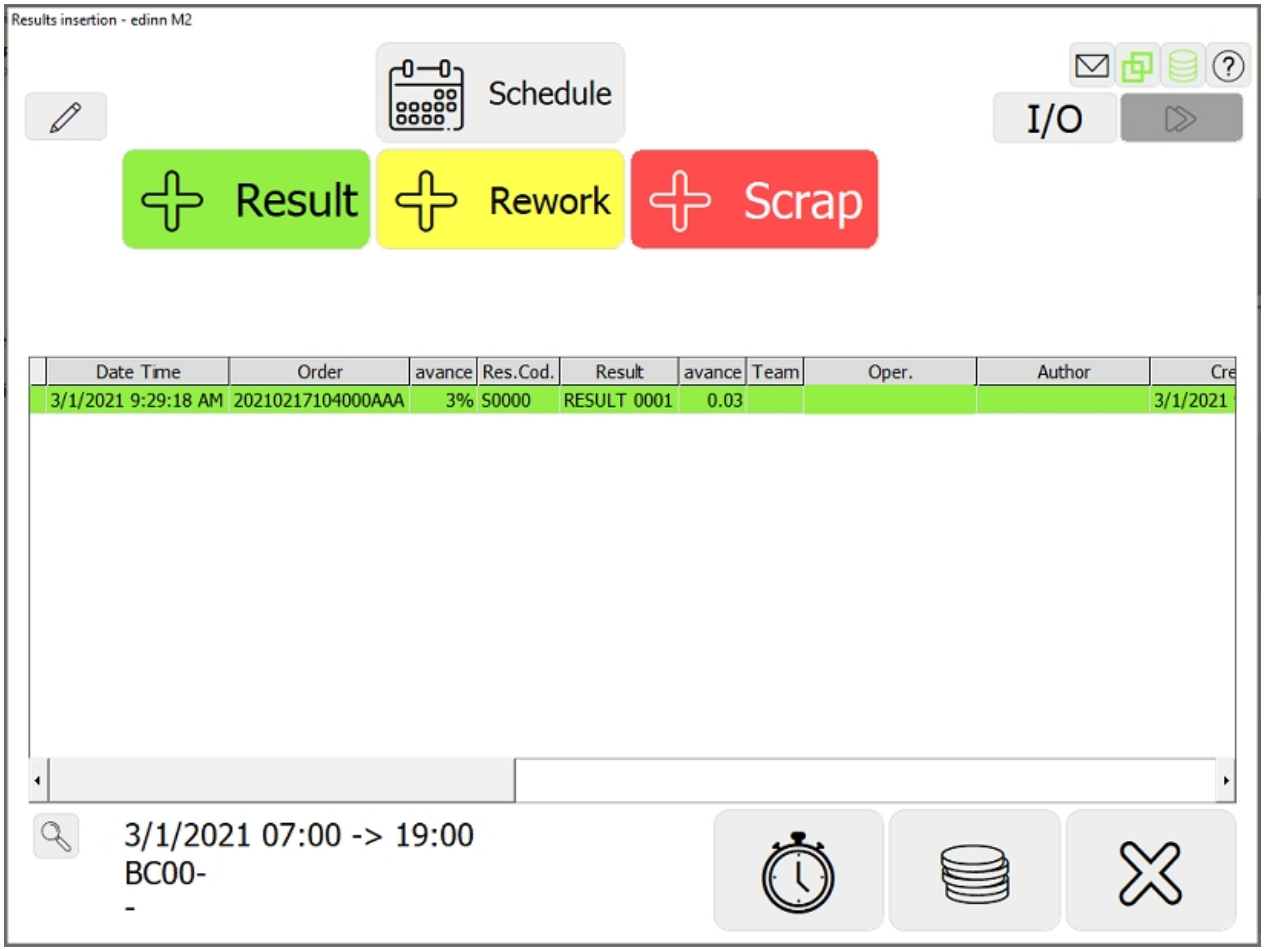
Step 1: Result Window

1. Click on the  button from the Main Window of the **Terminal**.
2. Select a **Process** in the dropdown.

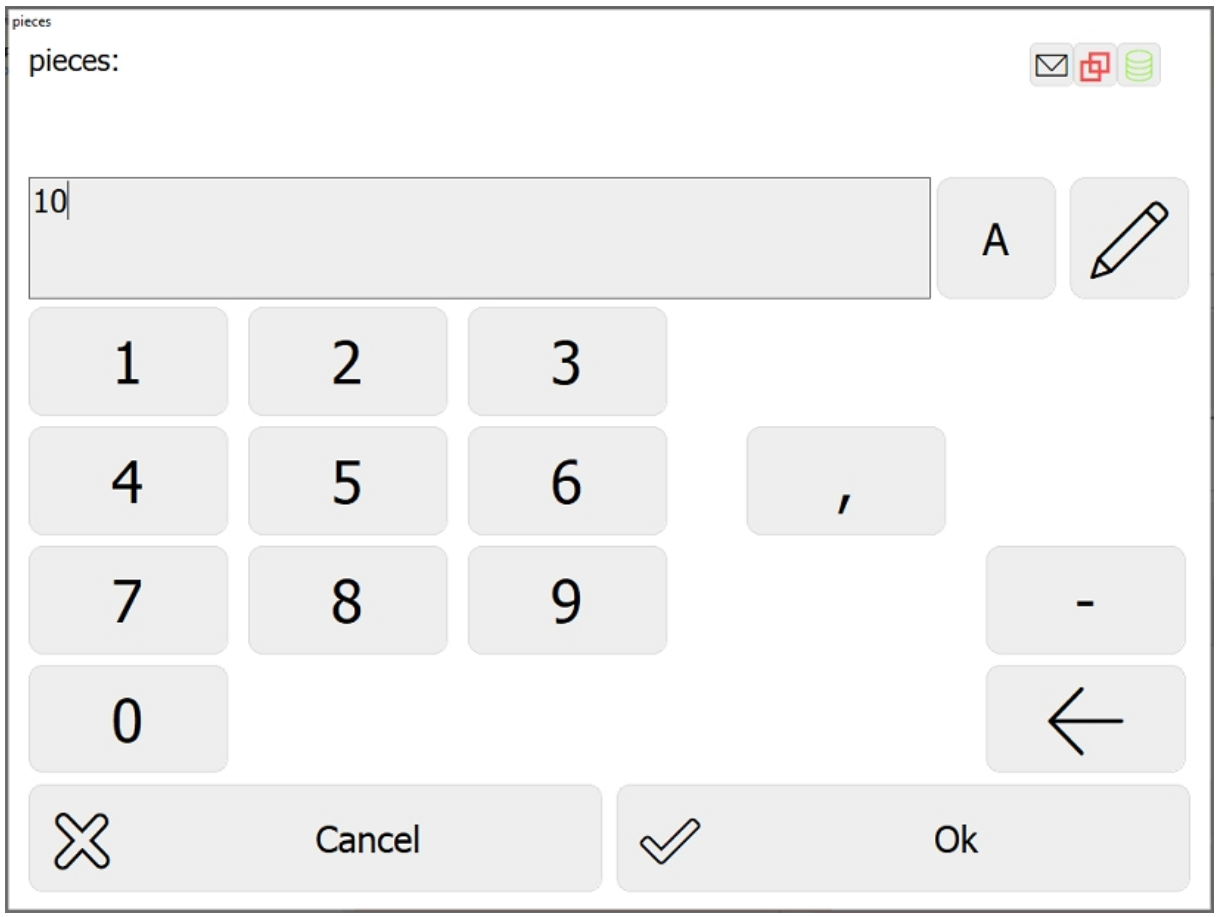


Step 2: Select a new result

1. Press the **[+ Result]** button:



2. Insert a quantity of production of the reference that is being produced and press ✓:



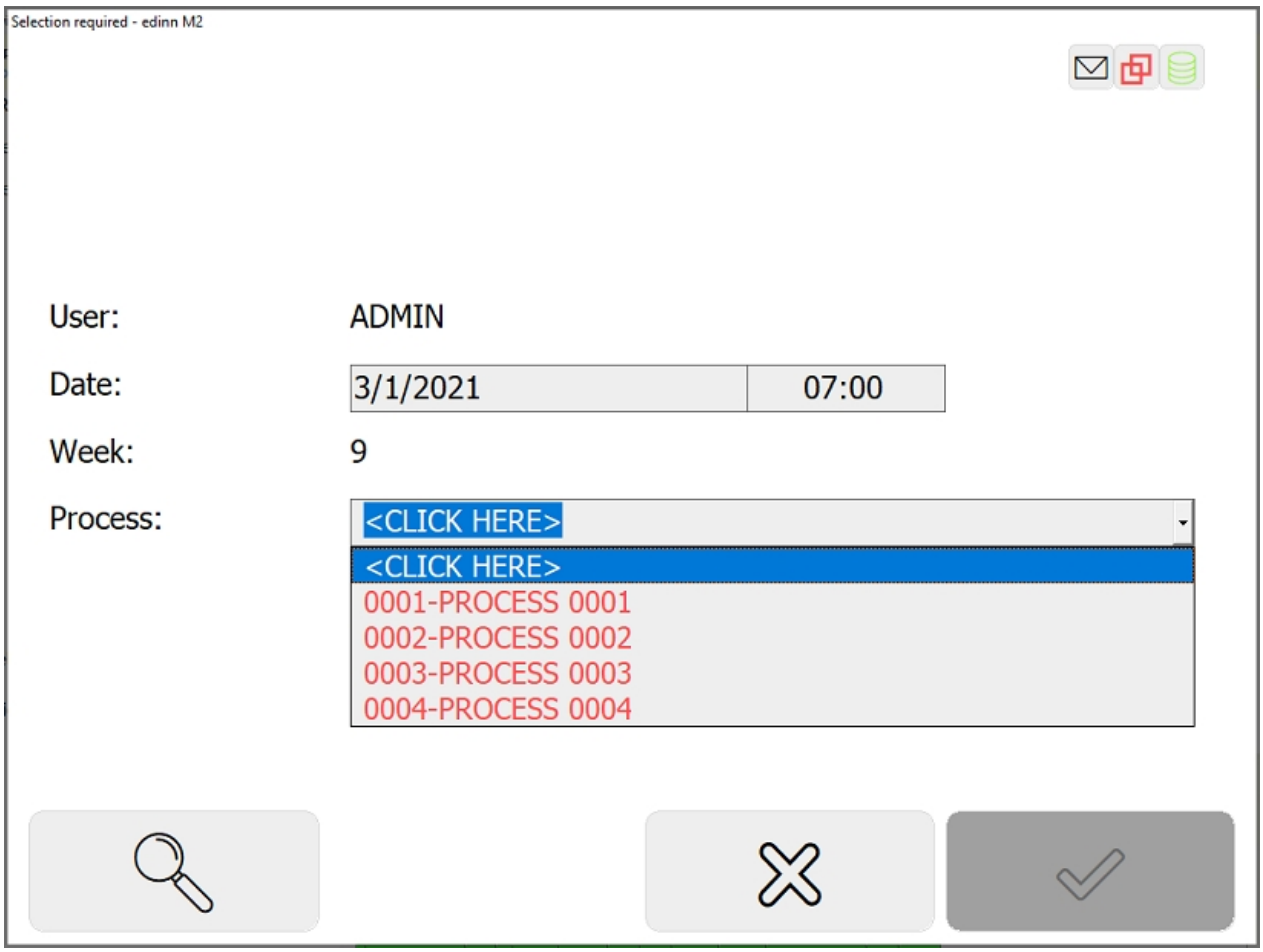
NOTE: to create a new Result, please see: Result configuration

How to change the result (without schedule)

This procedure details the steps for creating a new status. You can also view this video.

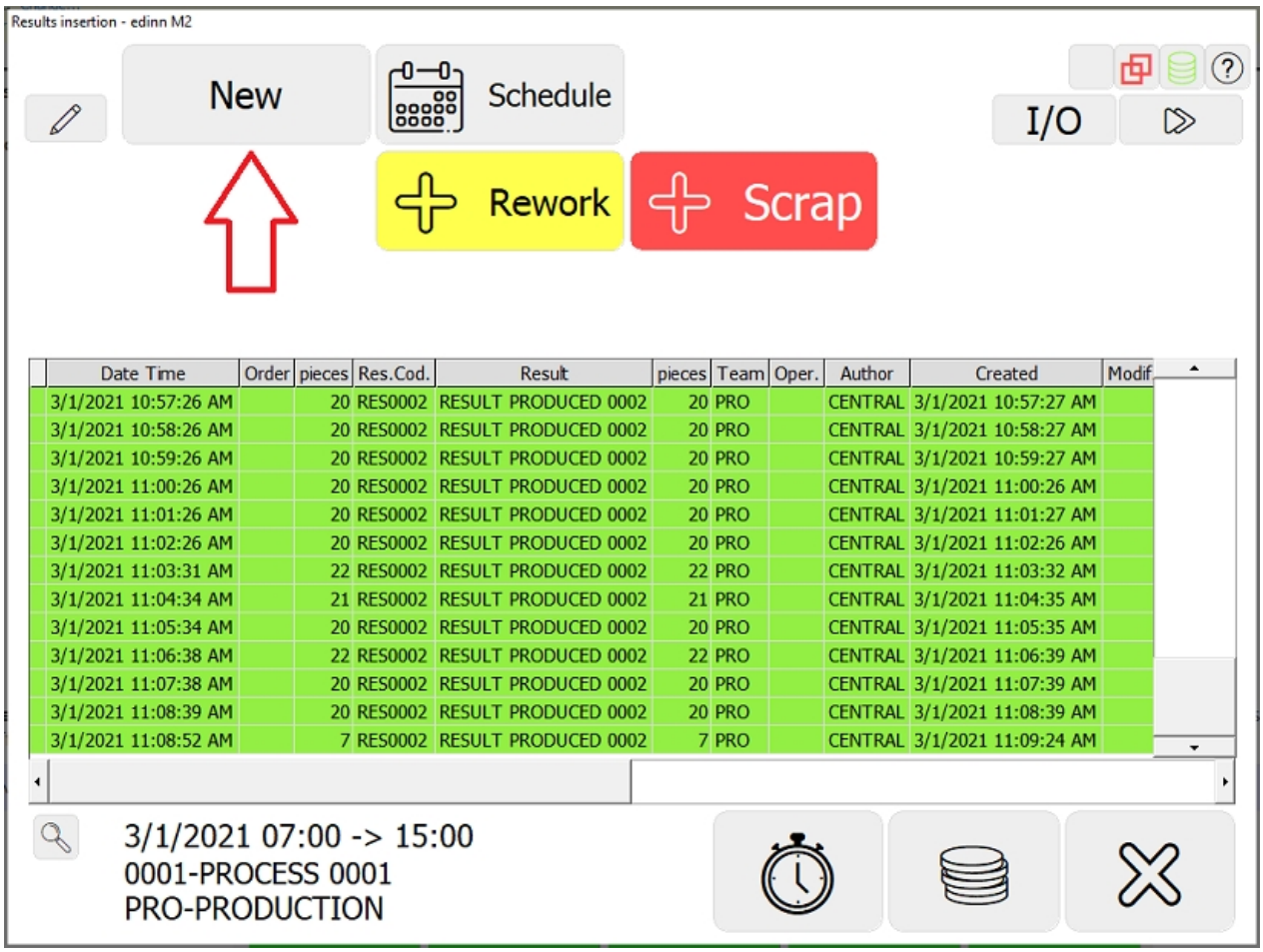
Step 1: Result Window

1. Click on the button from the Main Window of the **Terminal**.
2. Select a **Process** in the dropdown.



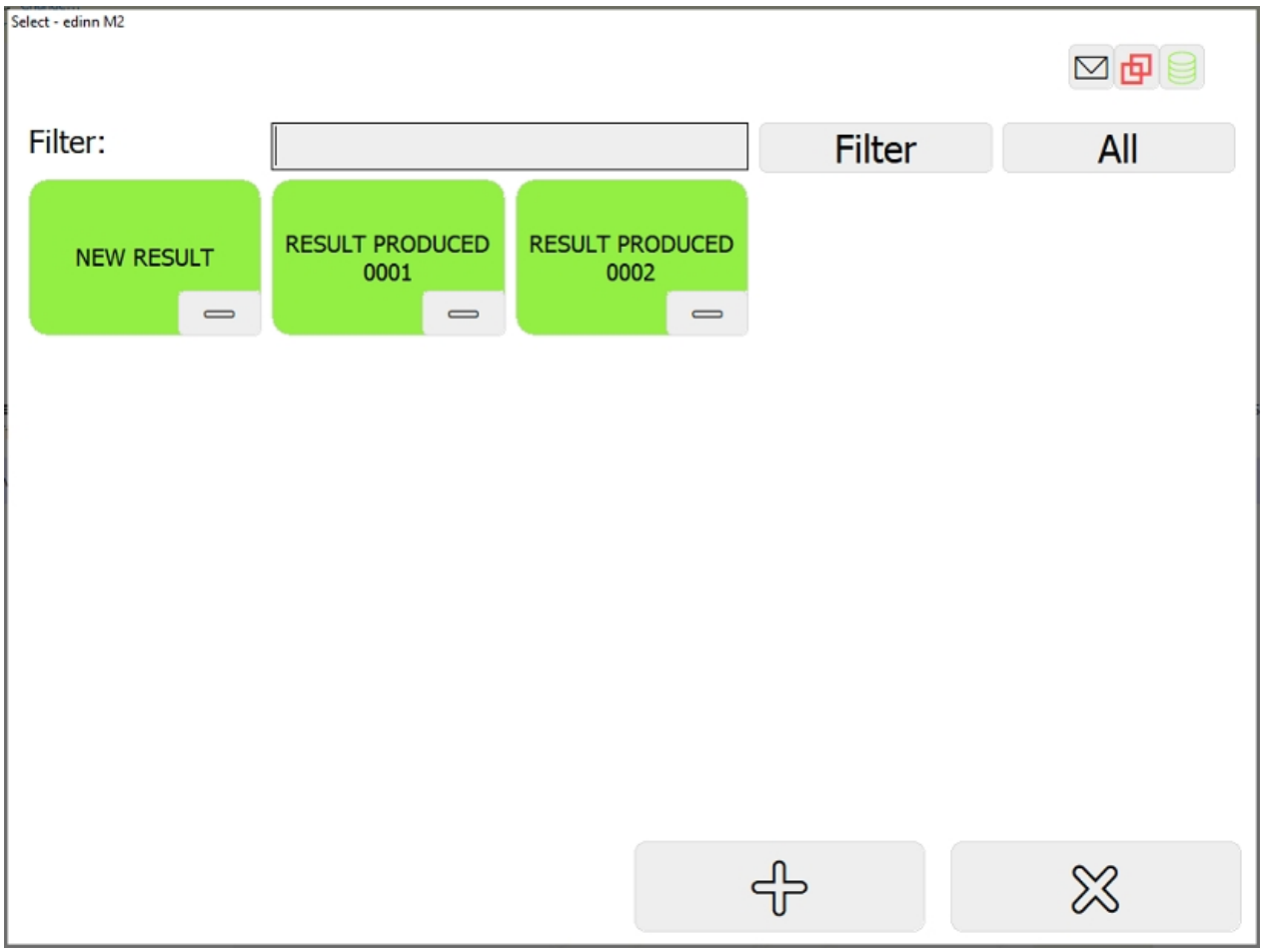
Step 2: Select a new result

1. Press the **[New]** button:



2. Select the **new reference** of result that will be produced from this moment on (the new reference will be inserted at the end of the grid) The **filter** at the top can be used to find the product by description or by identifier. When a product is selected, you will be asked to enter the order, add it and press **[ENTER]**.

NOTE: to create a new Result, please see: Result configuration.

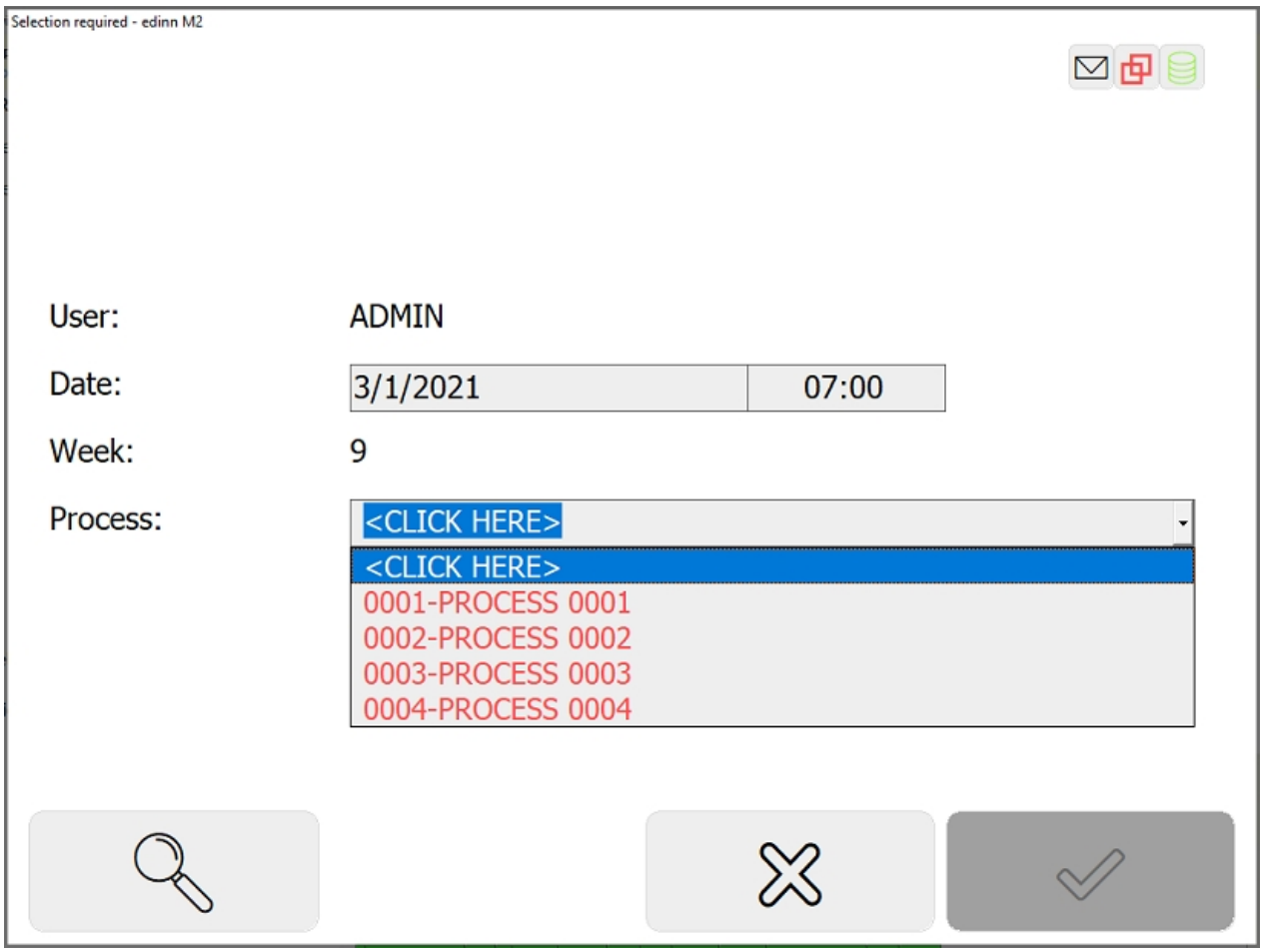


How to change the result (with schedule)

This procedure details the steps for creating a new status from the Schedule.

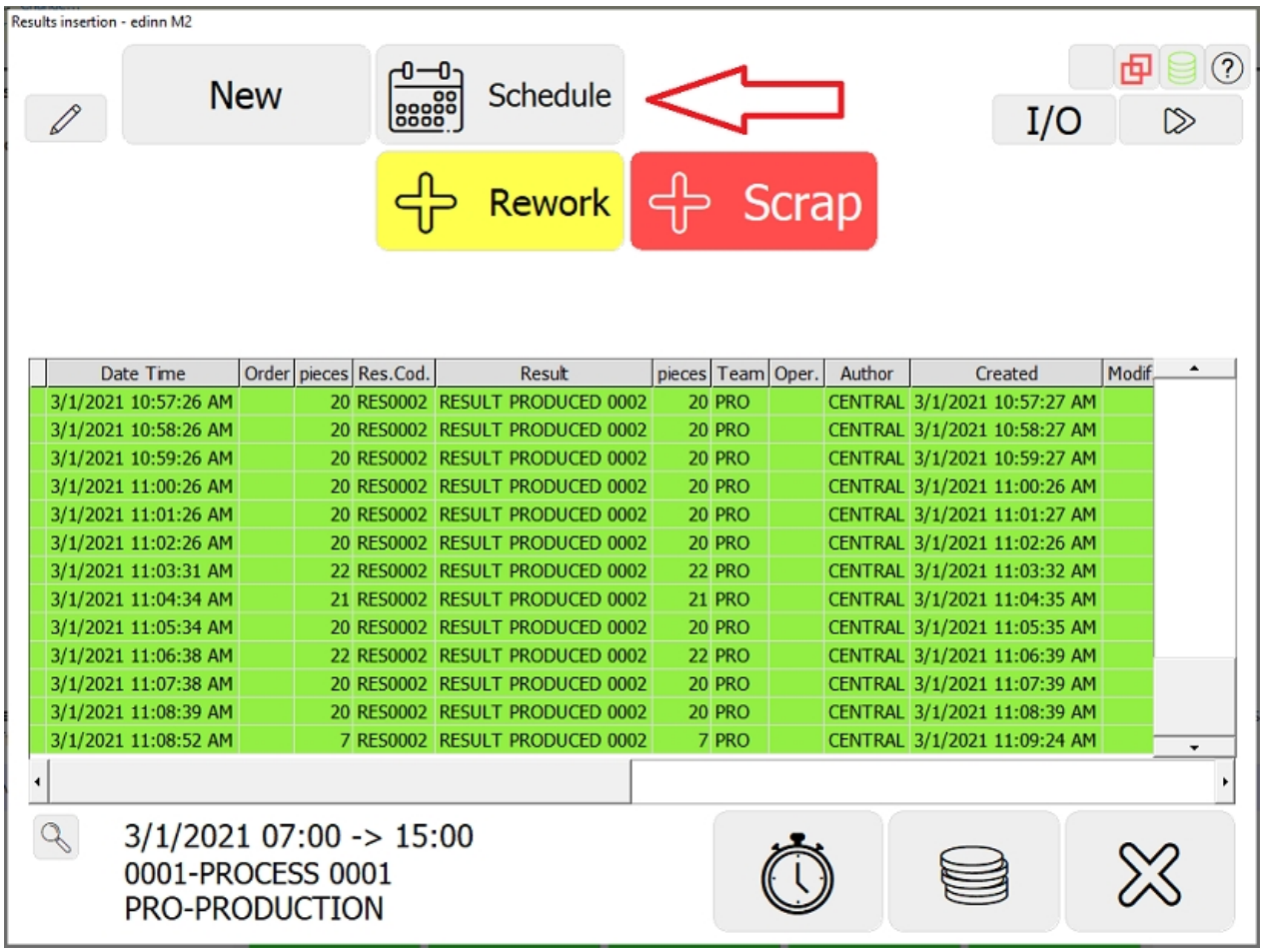
Step 1: Result Window

1. Click on the button from the Main Window of the **Terminal**.
2. Select a **Process** in the dropdown.

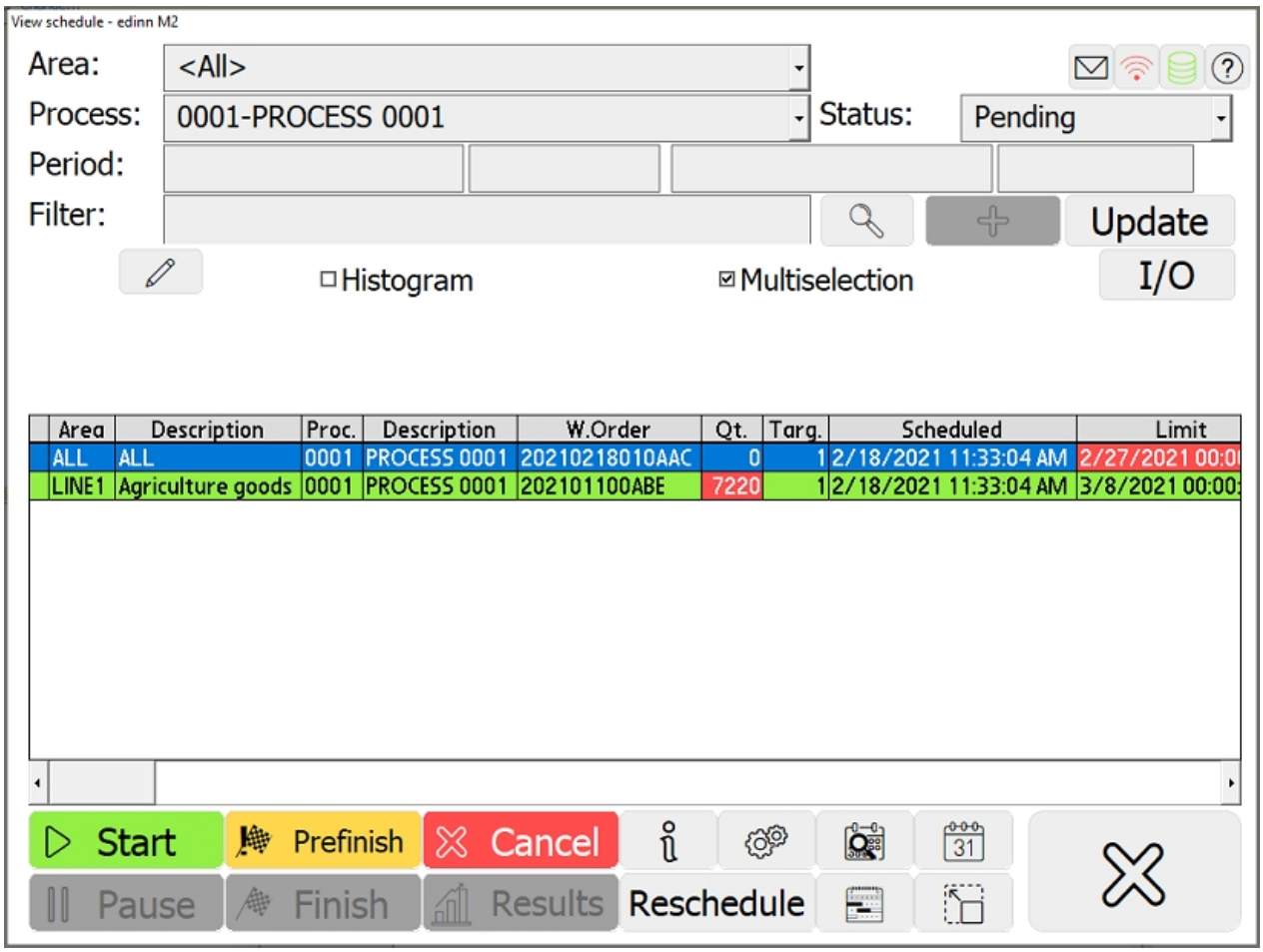


Step 2: Schedule window

1. Press the **[Schedule]** button



2. On the 'View Schedule' window, select the new **order** that you are about to start (the reference of the **order** will be inserted at the end of the grid) and press **[Start]**

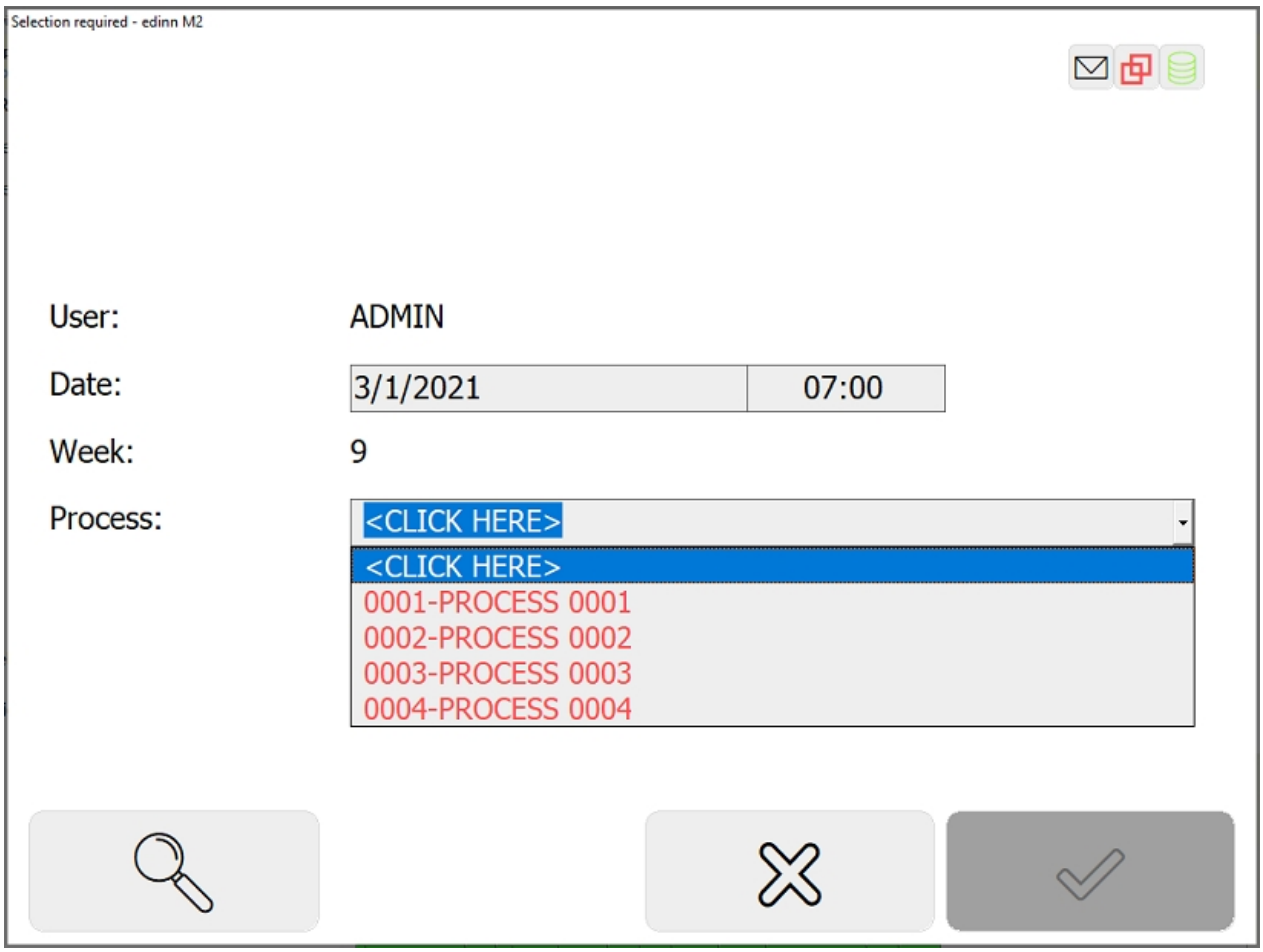


How to modify results

This procedure details the steps for modifying a Result.

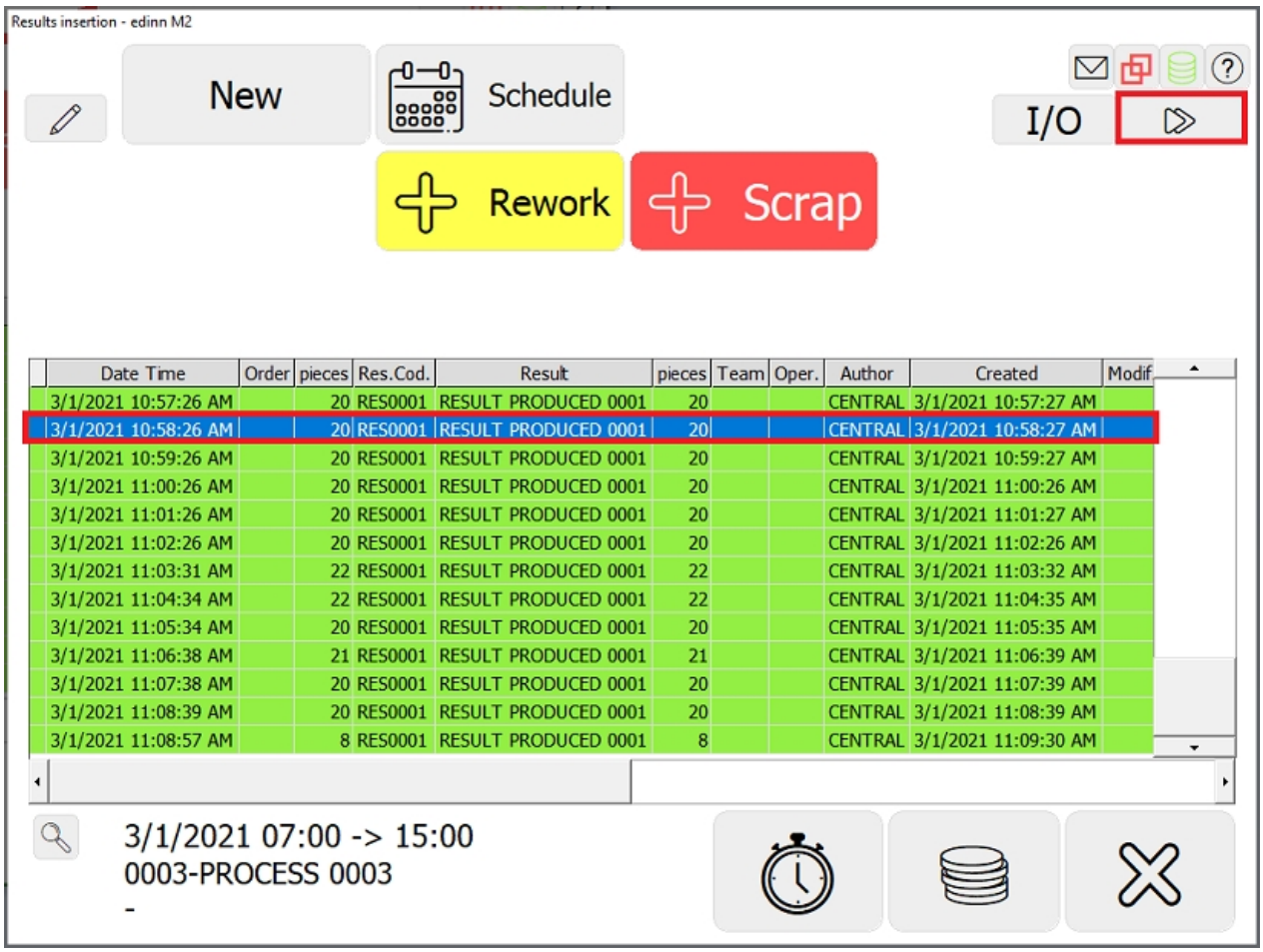
Step 1: Result Window

1. Click on the  button from the Main Window of the **Terminal**.
2. Select a **Process** in the dropdown.

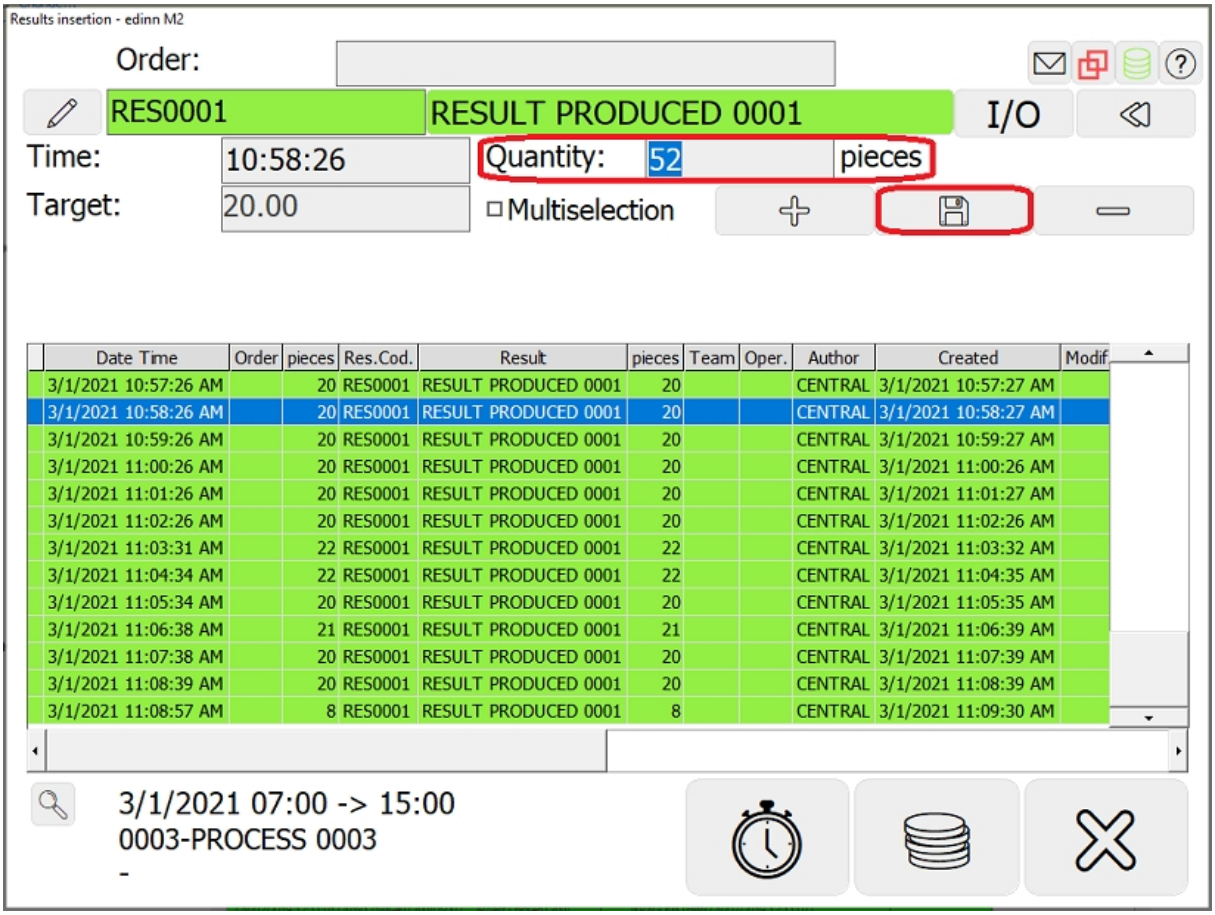


Step 2: Select and modify a result

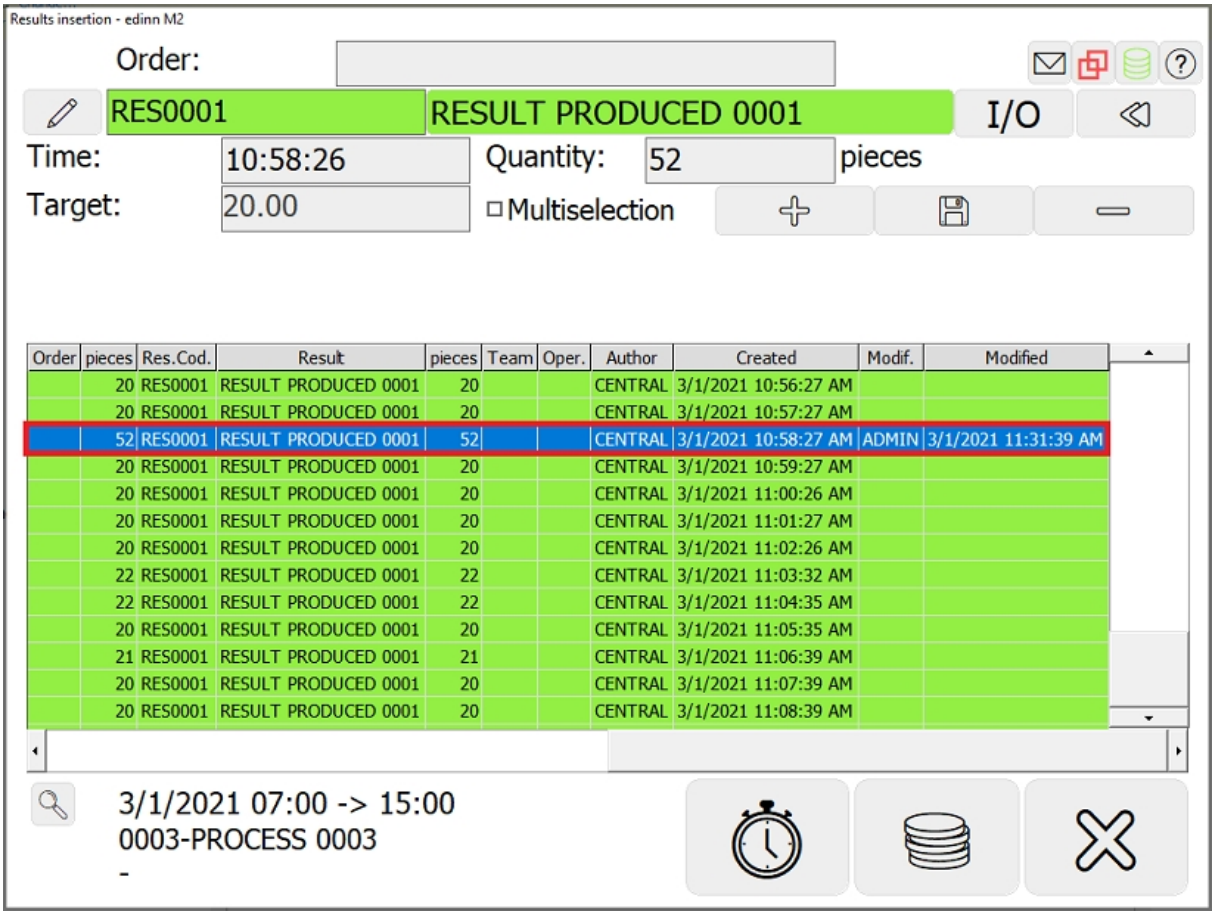
1. Press the row of the result you want to modify. Push in order to see the adjustable fields:



2. Modify the fields as you want and push the Save button ( ):



3. The table will be updated, including the changes and adding information about the date of the last modification and the user that did it:

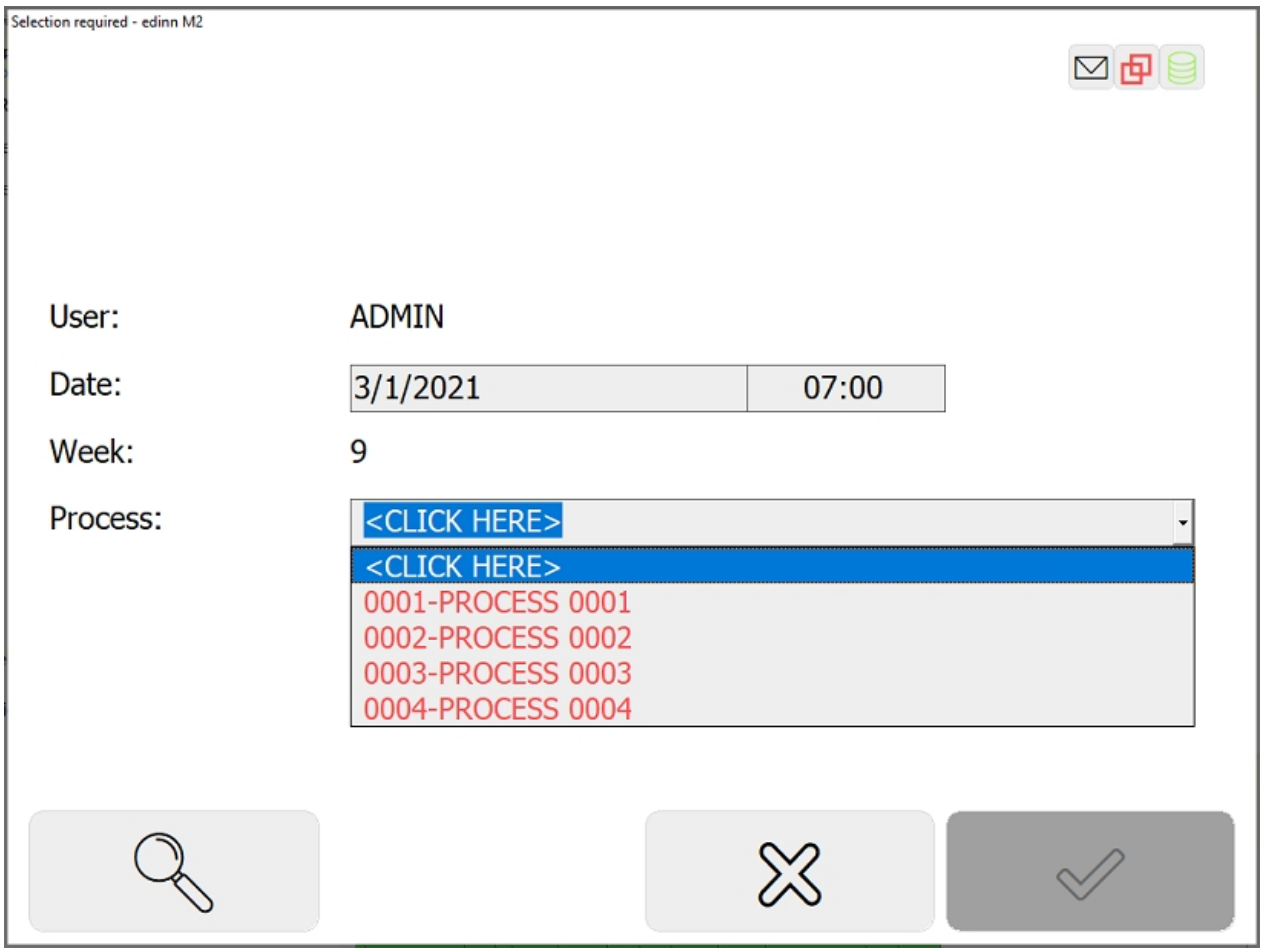


Status

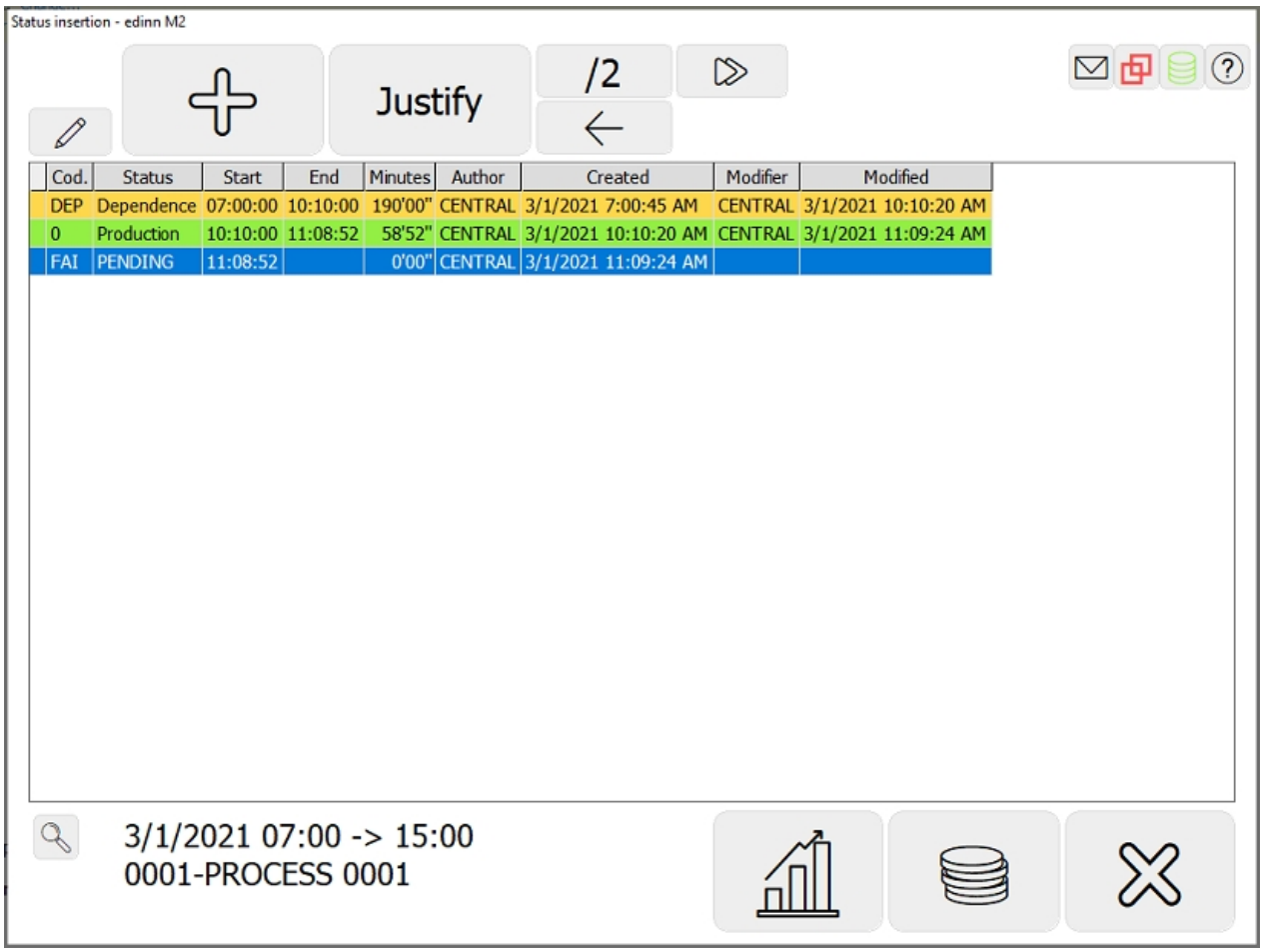
Main Window > Status

Utility: shows the different Statuses through which a process has passed.

If the process is not selected before clicking on the Status button, a selection window will be shown:



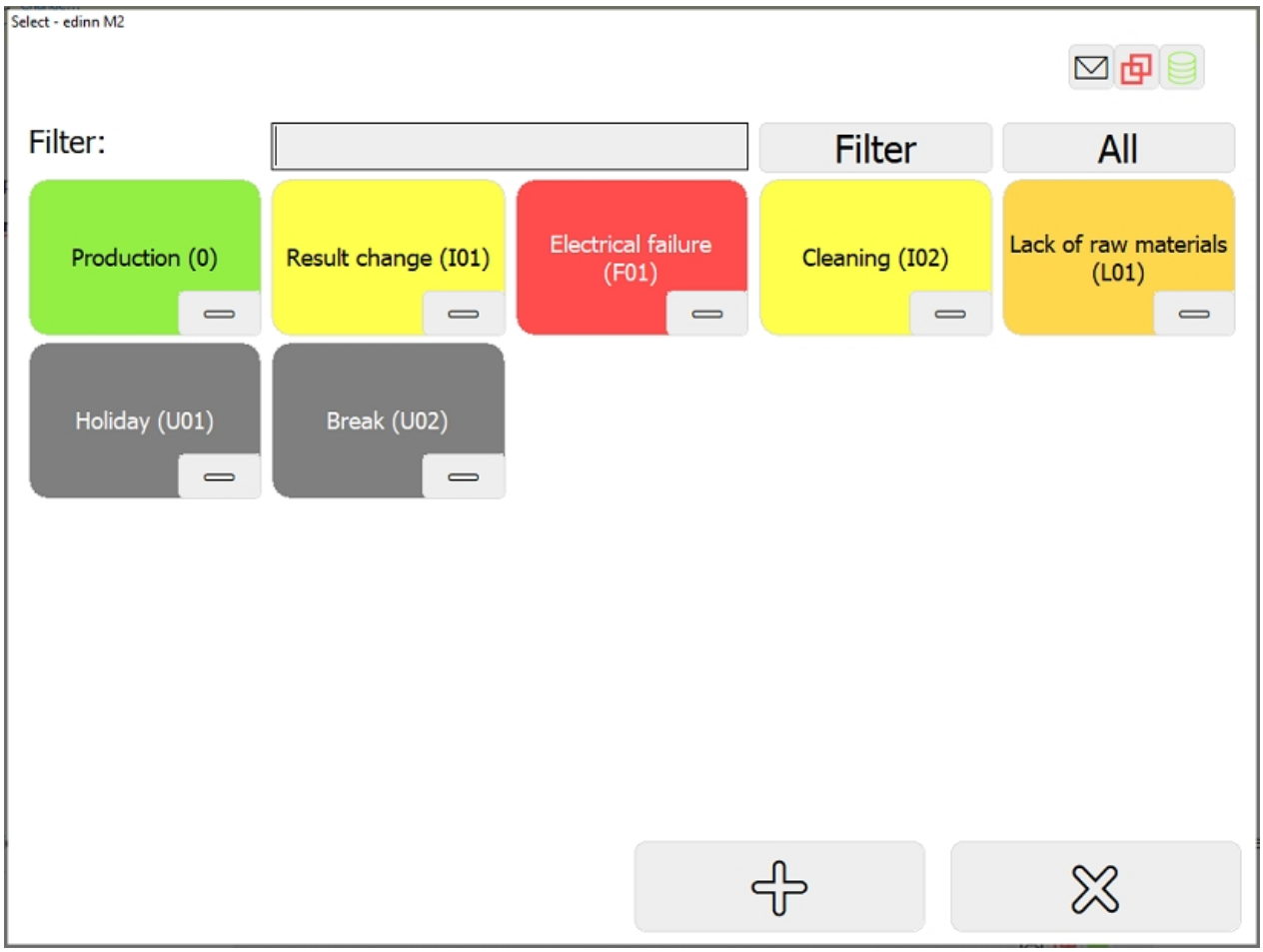
Once selected, the following window will be displayed:



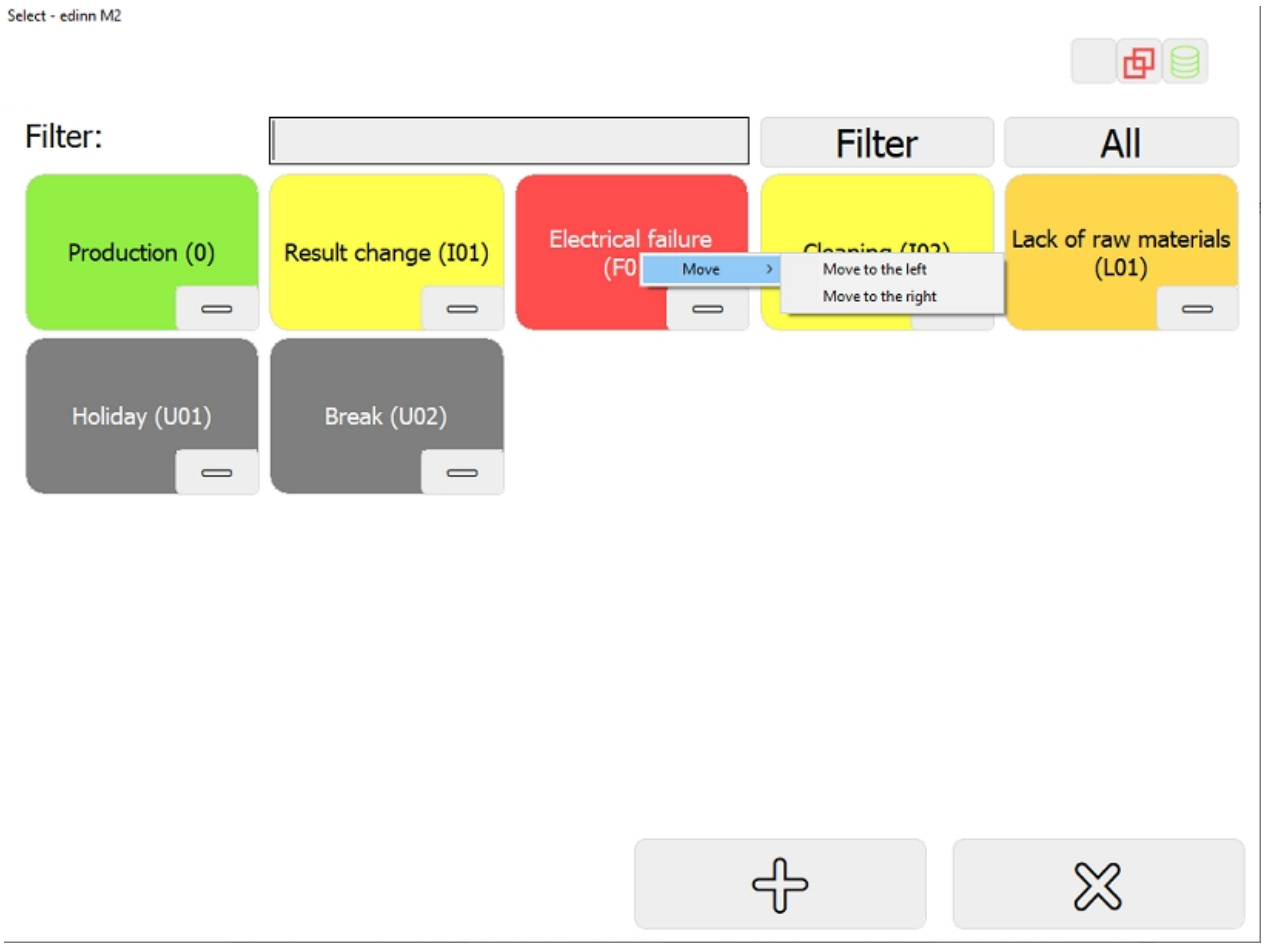
On this window we can:

- : generate a new status for the process.
- **Justify button**: to indicate an specific status for the selected status on the grid.

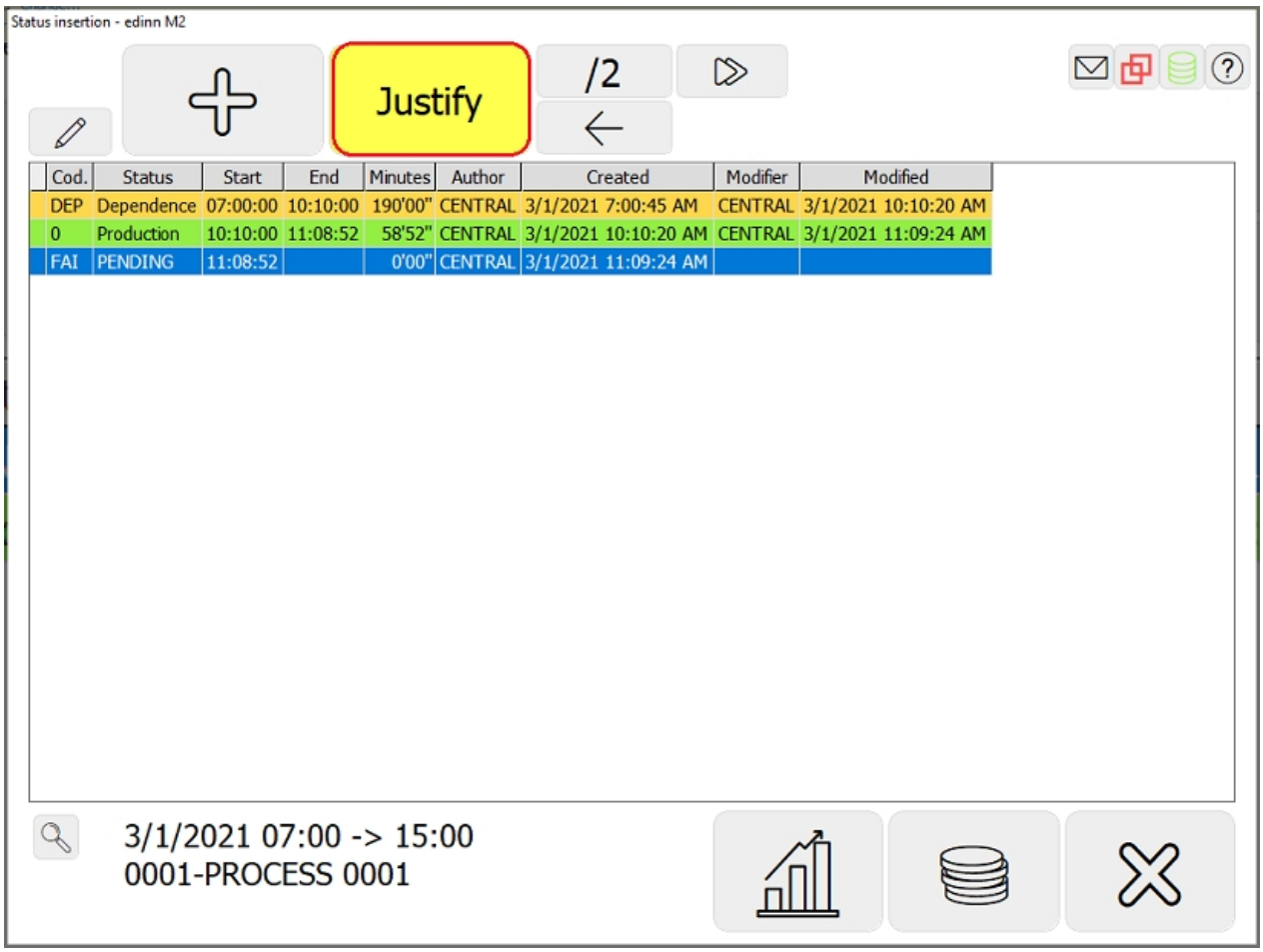
For both of the above buttons, the following window will be displayed with the status already configured:



From this window, if the user has administrator permissions, they can change the order in which the statuses appear. Simply click with the right mouse button and a menu will appear in which you can indicate whether you want to move the selected state to the right or to the left.



- /2: divides by 2 the selected status on the grid. This is useful for those cases where the system allows us to justify only 1 status, but we know that 2 statuses or more occured in that period. A finish time will be requested, for example: if you want to divide a status that goes from 11:00 to 13:00, you will have to write 12:00 to divide it in half.
- ←: allows to go to the next status that needs justification. If there are, the button will be shown in yellow. The following window will be displayed with the pending status to justify:



- : allows to go to the advanced editing functions. This function will show the status code and description and the start and end time, for **manual correction** of the data in the grid. You need Supervisor role to change status code or start and end time on the grid.

  Also you will be able to create future statuses. Future statuses can be generated when the user knows that from a certain moment on, the process will be in a certain status. For example: *a status of maitenance*. If the current status has a maximum time, for example, of 10 minutes, once surprassed, the next status will be the future status configured.

At the columns of the grid, at the first column, a ' * ' symbol indicates that there is a user comment for that record. You will have to double click the data record, or click the same ' * ' symbol above the table, in order to go to the comment.

Please note that there are user restrictions for the statuses' justification as follows.

- A user with the Supervisor role can change everything.
- The rest of the users can modify a status, as long as they are not:

- Trying to convert a non production status into a production one.
- Changing the times of start or end.
- Deleting the status.
- The rest of changes will be allowed only if the status is not locked by time and if the status (any of the following):
  - Is pending to be justified: status FAI, DEP, etc.
  - Is a generic micro stop.
  - Was previously justified by the user.

NOTE: you will be able to access the window for selecting the period and process again by clicking in the lower left box.

Procedures

In this section the following procedures for configure the statuses are detailed:

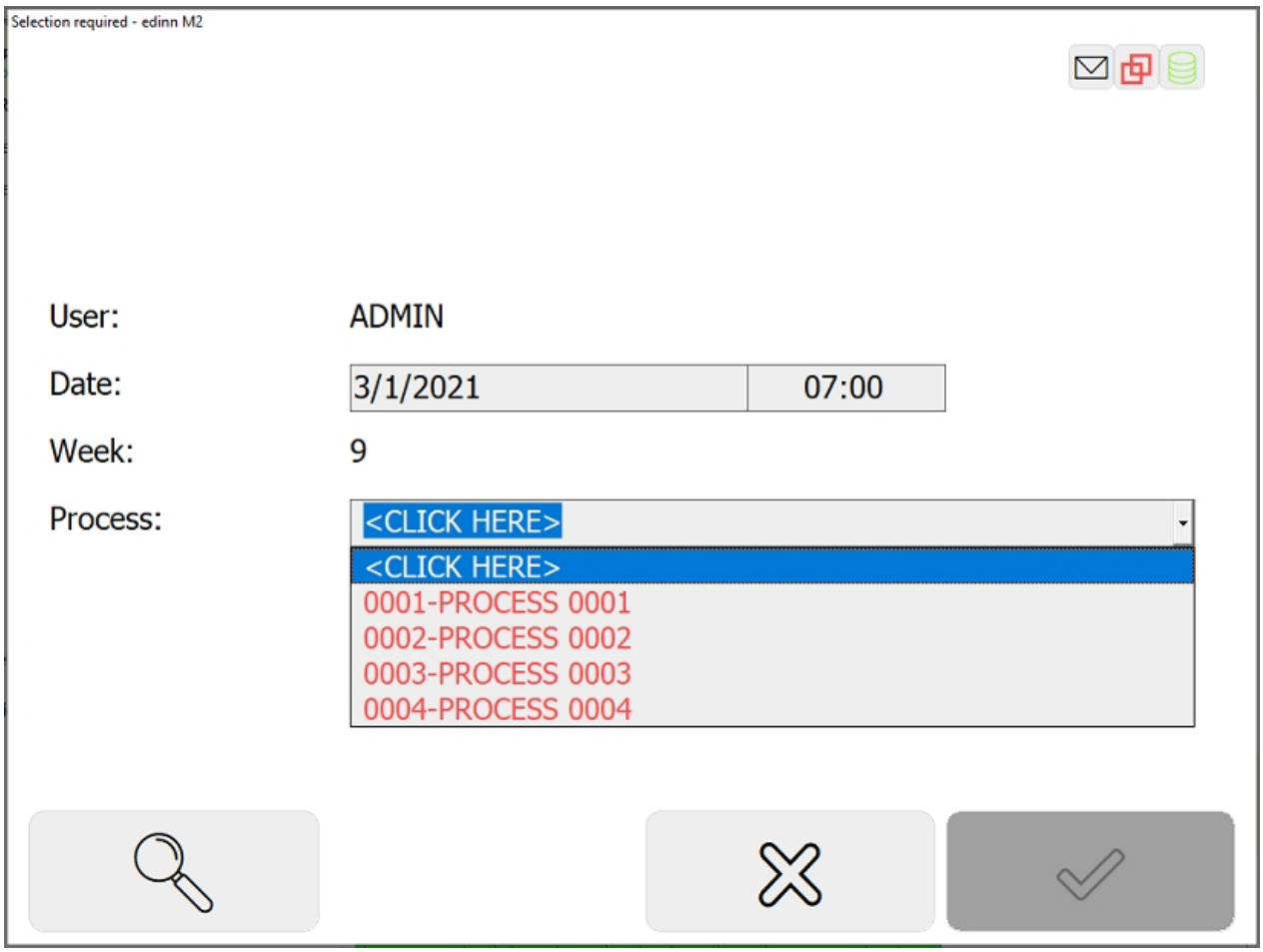
- How to create a new status.
- How to schedule the statuses:
  - How to schedule the statuses for all the processes.
  - How to schedule the status just for a specific process.
- How to configure exceptions for the statuses.

How to insert a new status

This procedure details the steps to insert a new status.

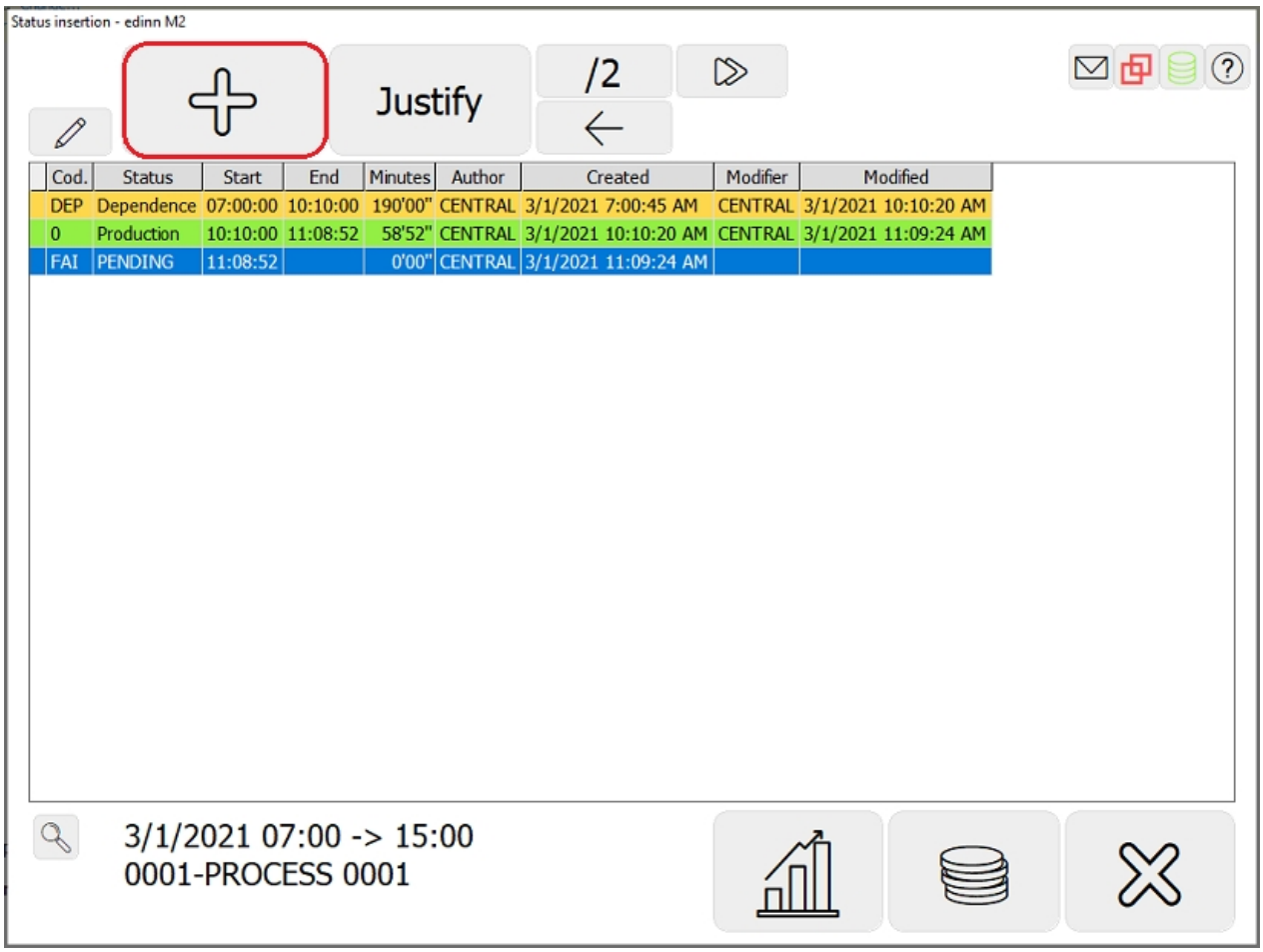
Step 1: Status Window

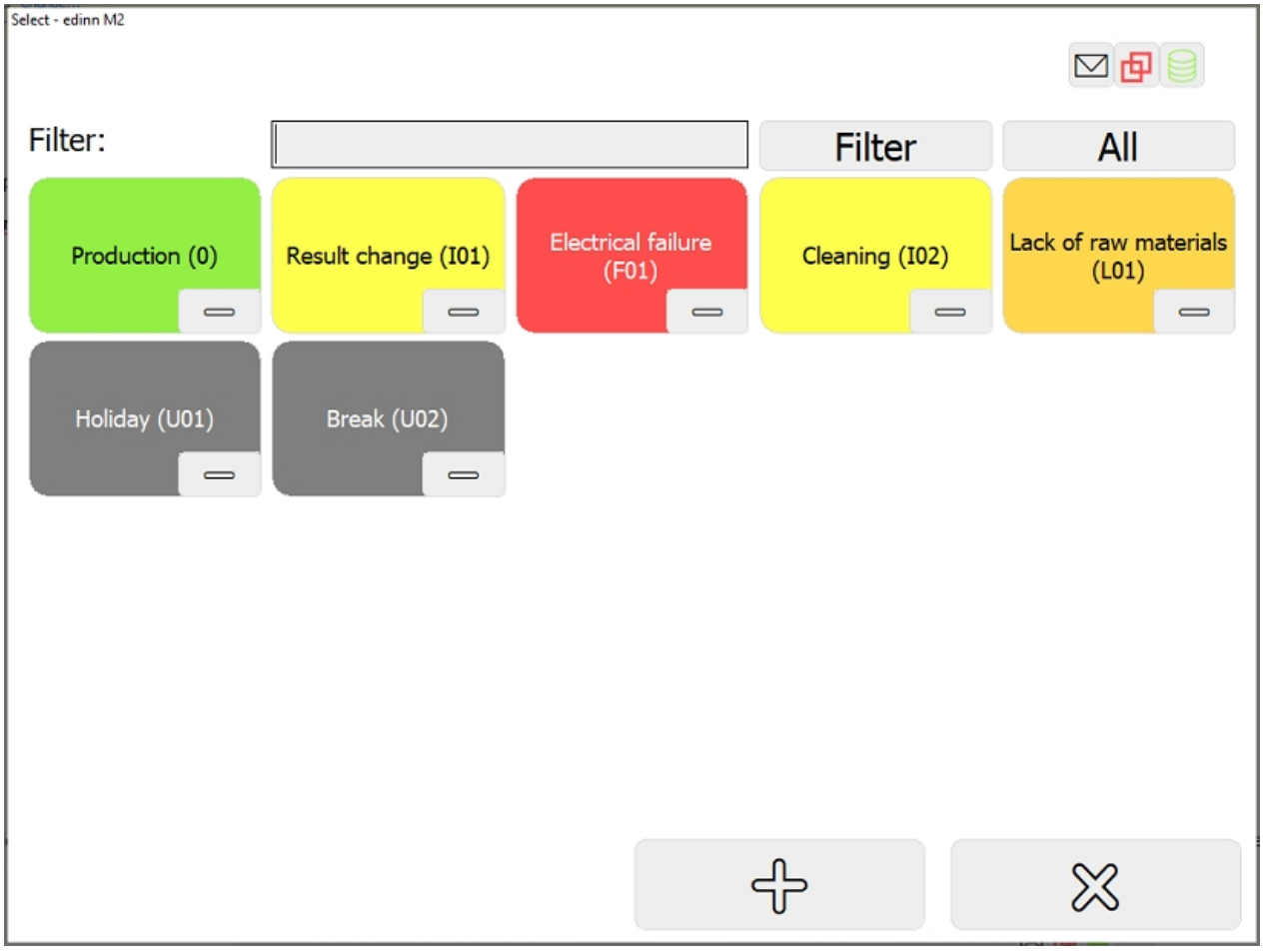
1. Click on the  button from the **Main Window** of the **Terminal**.
2. Select a **Process** in the dropdown.



Step 2: Select a new status

1. Click the ✚ button.
2. Select in the following window the status that you want to introduce.


Status insertion - edinn M2
Justify
/2
Cod.
Status
Start
End
Minutes
Author
Created
Modifier
Modified
DEP
Dependence
07:00:00
10:10:00
190'00"
CENTRAL
3/1/2021 7:00:45 AM
CENTRAL
3/1/2021 10:10:20 AM
0
Production
10:10:00
11:08:52
58'52"
CENTRAL
3/1/2021 10:10:20 AM
CENTRAL
3/1/2021 11:09:24 AM
FAI
PENDING
11:08:52
0'00"
CENTRAL
3/1/2021 11:09:24 AM
3/1/2021 07:00 -> 15:00
0001-PROCESS 0001


Select - edinn M2
Filter:
Filter
All
Production (0)
Result change (I01)
Electrical failure (F01)
Cleaning (I02)
Lack of raw materials (L01)
Holiday (U01)
Break (U02)

NOTE: To create new statuses or to change the actual ones, please refer to "How to create a new status".

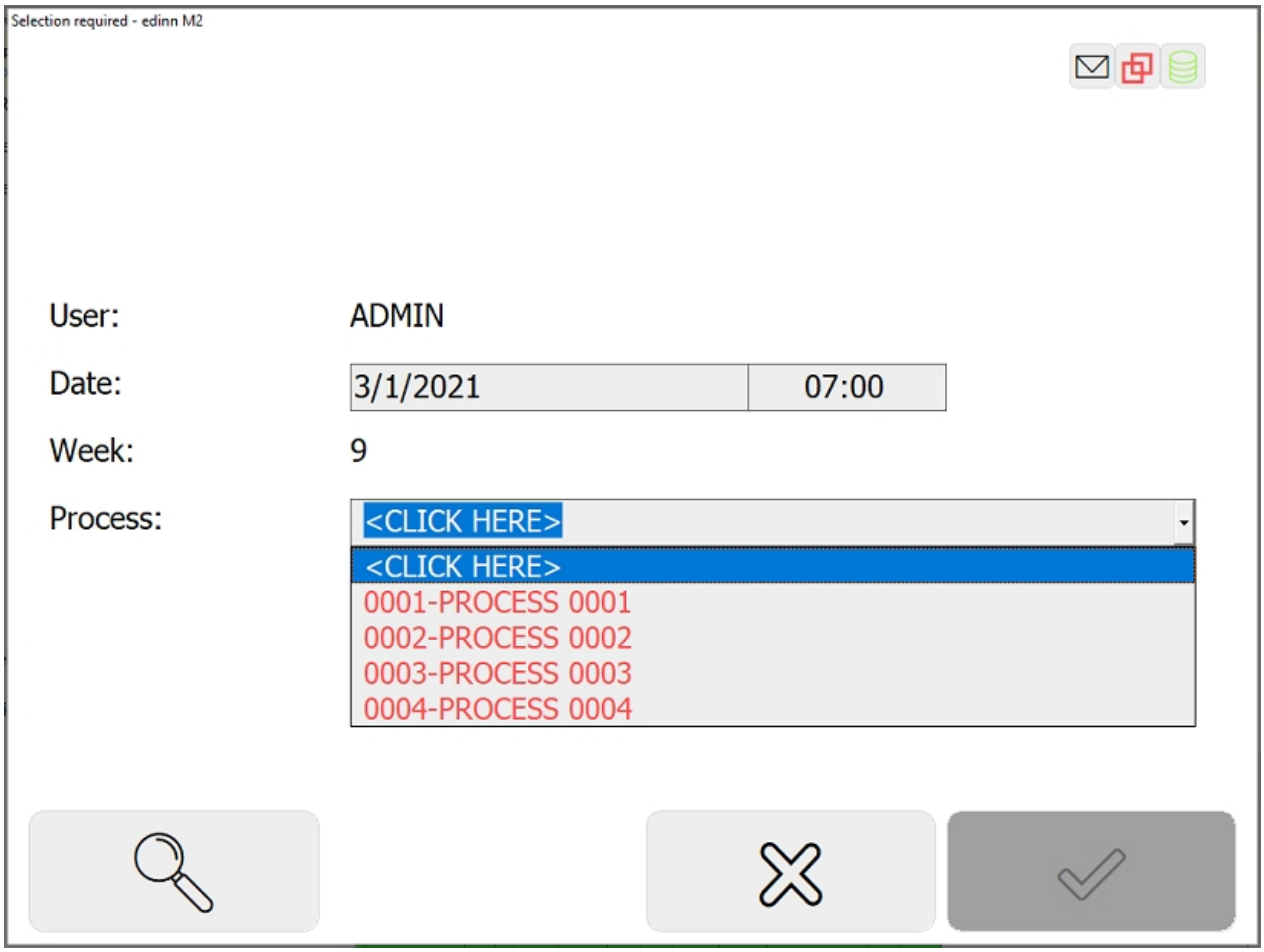
How to justify a stop status

This procedure details the steps to justify a stop status. You can also view this video.

Step 1: Status Window

When a stop in the machine is produced, the Status button  in the Main Window will blink.

1. Click on the  button from the **Main Window** of the **Terminal**.
2. Select a **Process** in the dropdown.

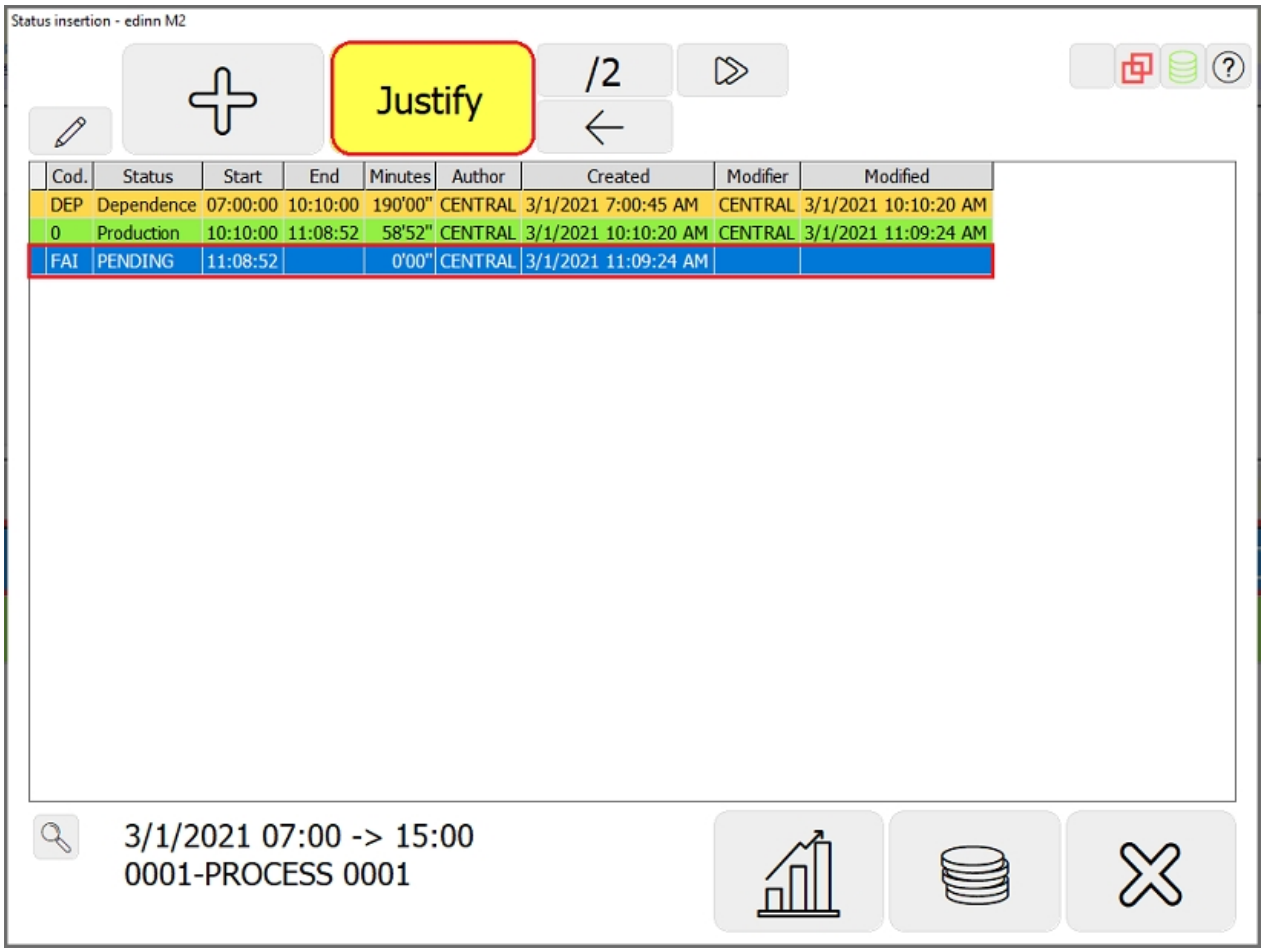


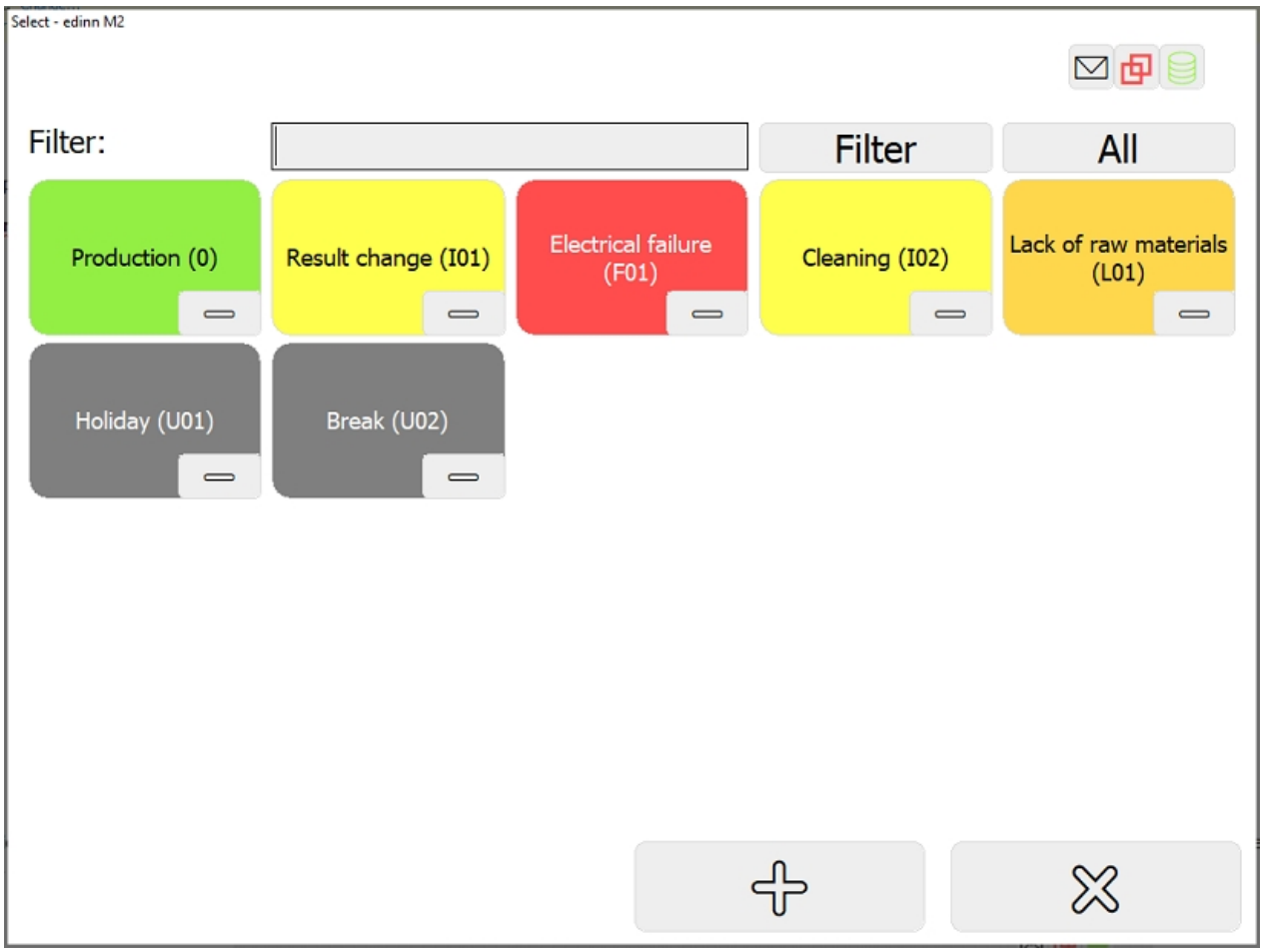
Step 2. Justify Status

The status that are not yet justified will be shown with a '**FAI**' code (abbreviation of failure). To justify them:

1. Select the 'FAI' status and then clic the **[Justify]** button.
2. Select the proper status, between the ones available:

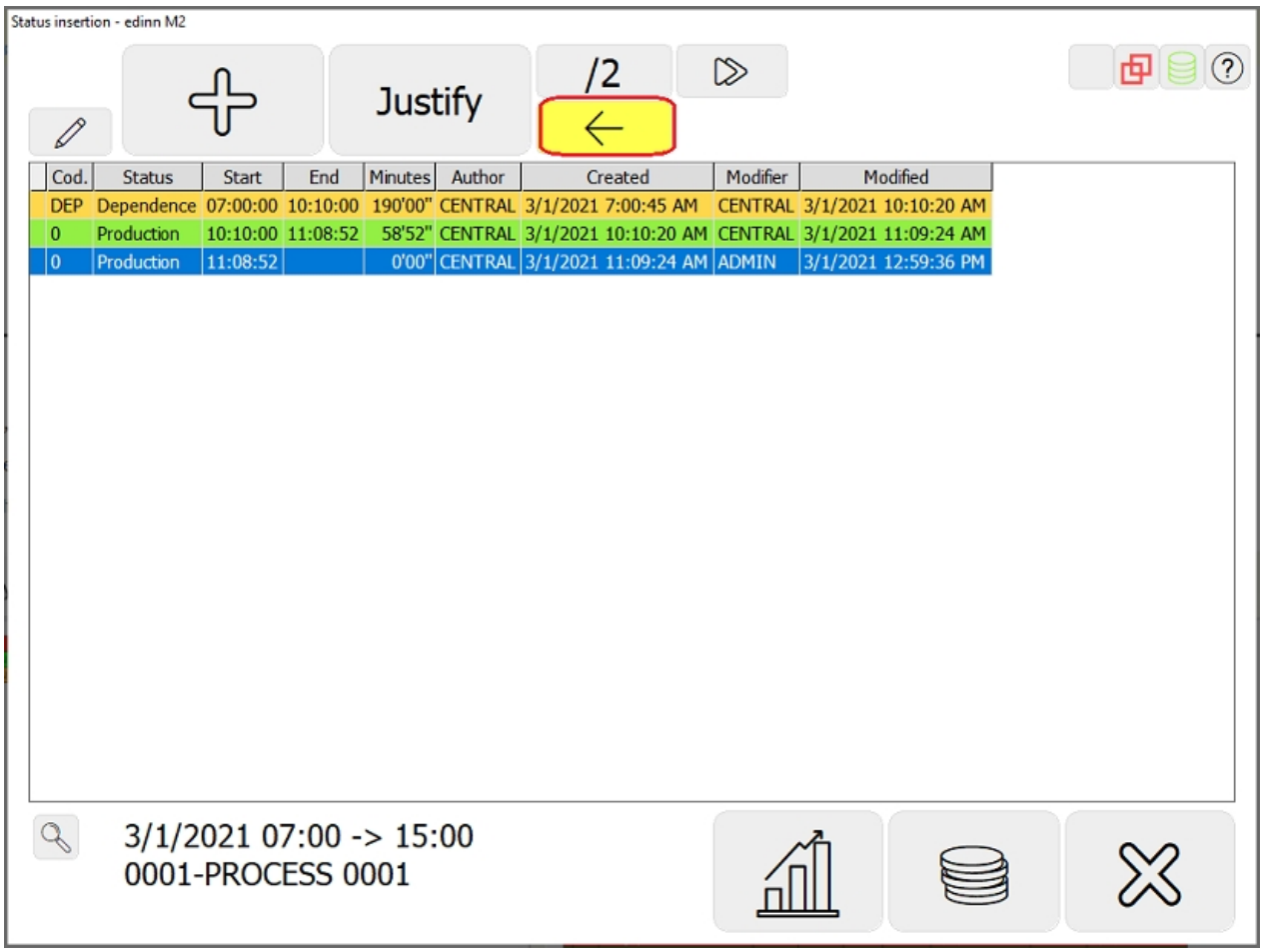
NOTE: to create a new status, please see: "How to create a new status". To modify a new status, please see: "How to modify a status"

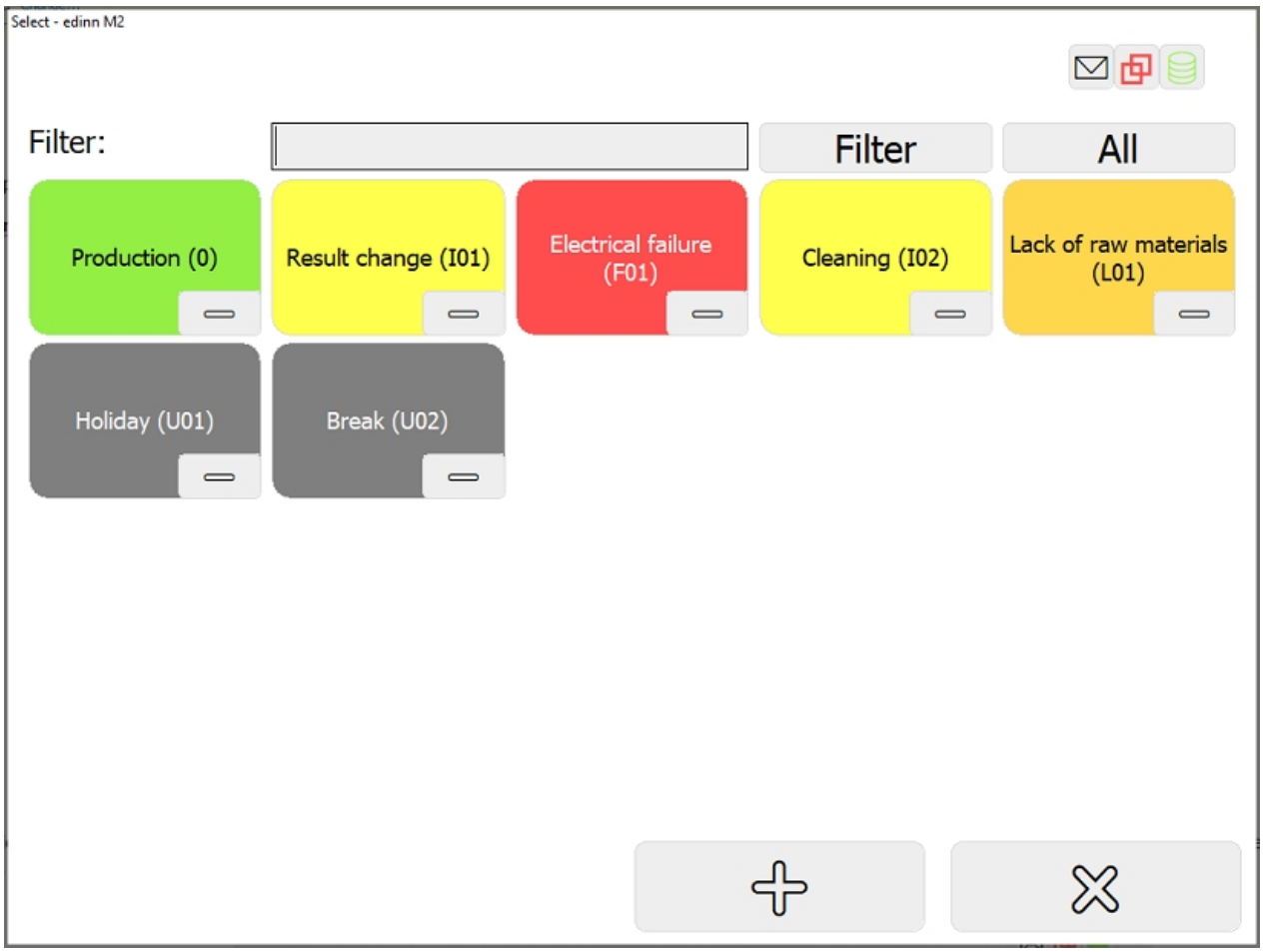




Step 2. Next 'FAI' status

1. By clicking the blinking button ←, you will be brougth to the next pending status which requires justification, even if it belongs to another period.
2. Select the 'FAI' status and then clic the **[Justify]** button.
3. Select the proper status, between the ones available:


Status insertion - edinn M2
Justify
/2
Cod. Status Start End Minutes Author Created Modifier Modified
DEP Dependence 07:00:00 10:10:00 190'00" CENTRAL 3/1/2021 7:00:45 AM CENTRAL 3/1/2021 10:10:20 AM
0 Production 10:10:00 11:08:52 58'52" CENTRAL 3/1/2021 10:10:20 AM CENTRAL 3/1/2021 11:09:24 AM
0 Production 11:08:52 0'00" CENTRAL 3/1/2021 11:09:24 AM ADMIN 3/1/2021 12:59:36 PM
3/1/2021 07:00 -> 15:00
0001-PROCESS 0001



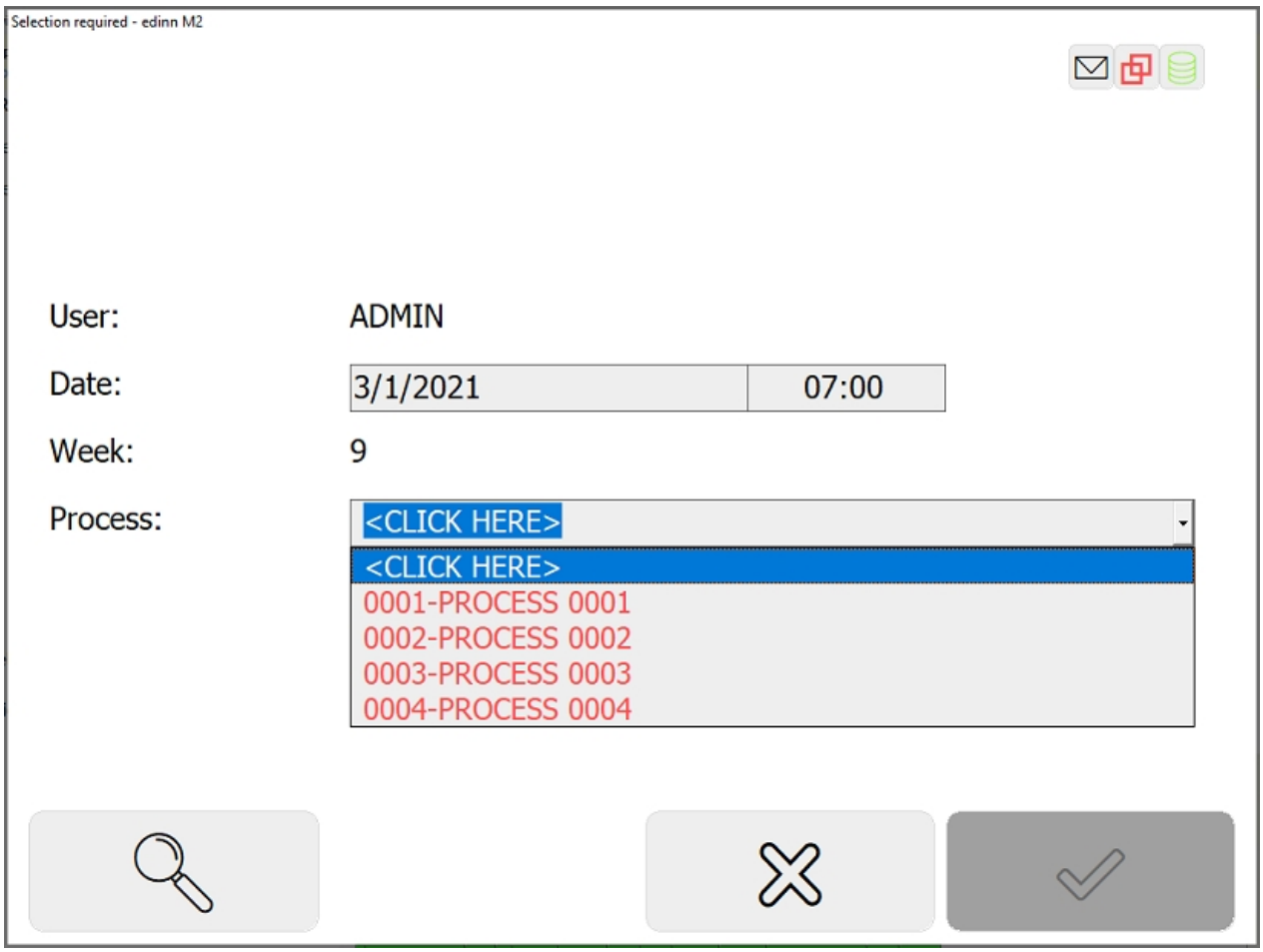
How to generate a future status

Future statuses can be generated when the user knows that from a certain moment on, the process will be in a certain status. For example: *a status of maitenance*.

If the current status has a maximum time, for example, of 10 minutes, once surprassed, the next status will be the future status configured.

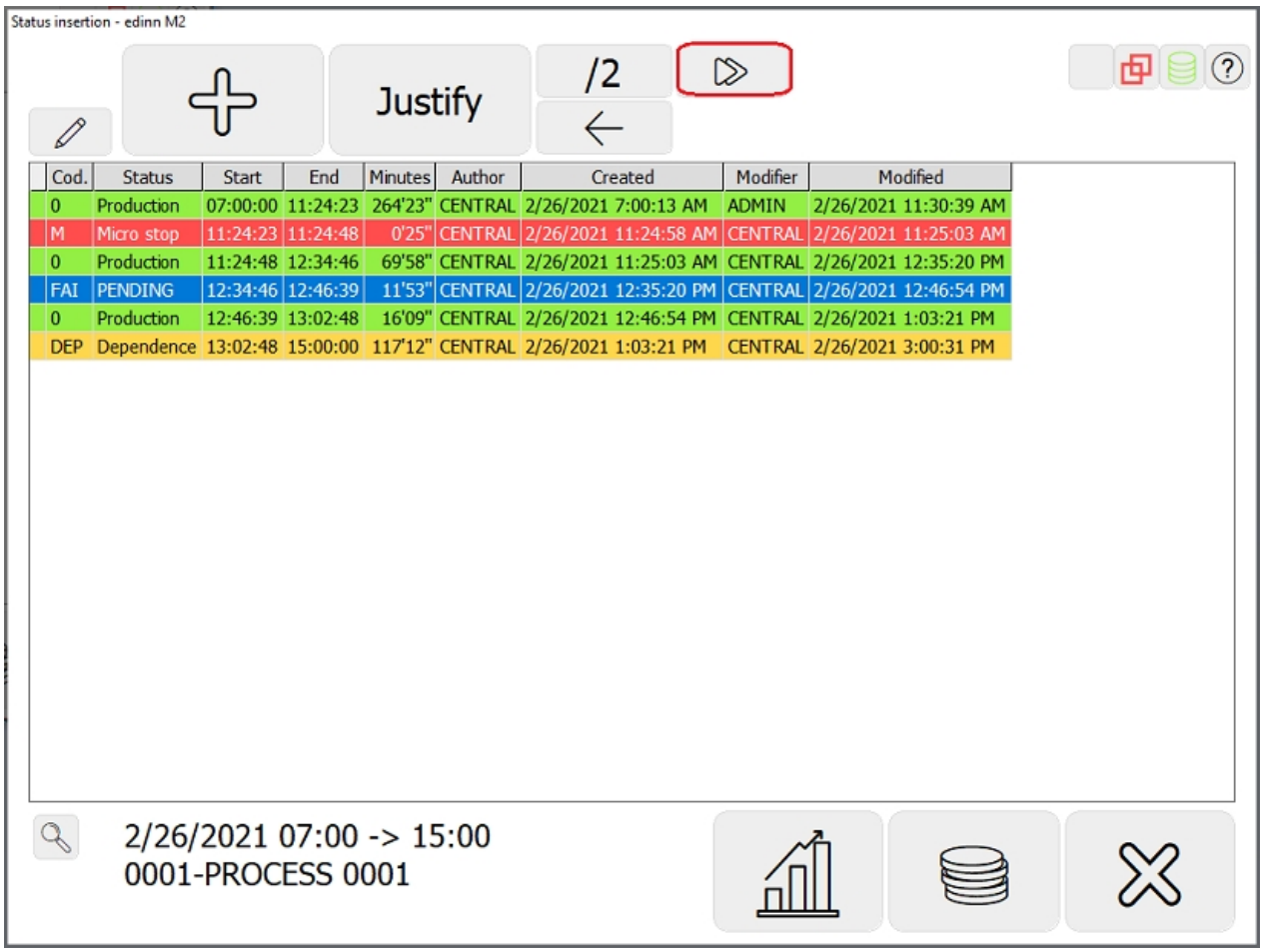
Step 1: Status Window

1. Click on the  button from the **Main Window** of the **Terminal**.
2. Select a **Process** in the dropdown.



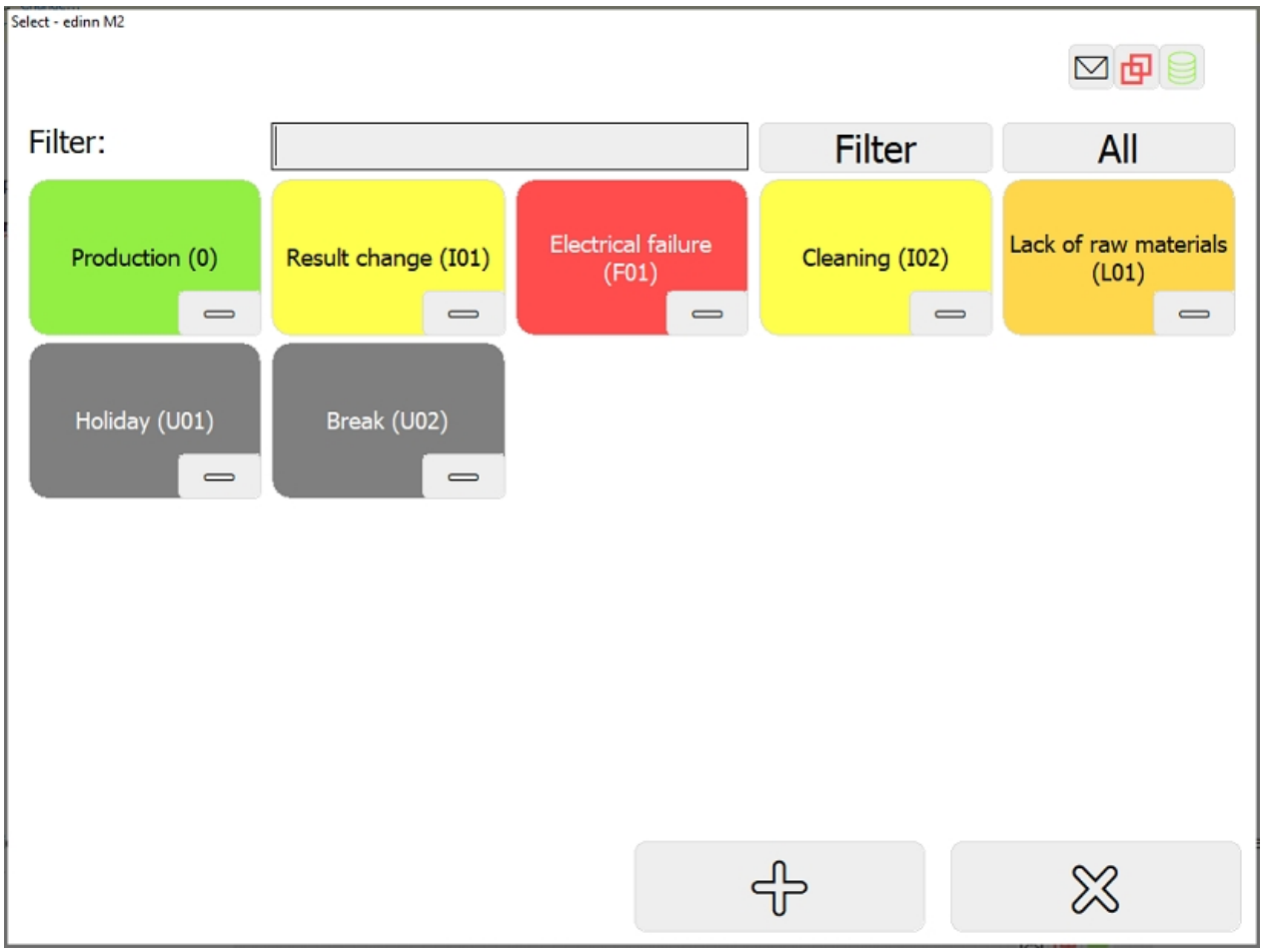
Step 2. Future status

1. Clic the  button to access the advance editing functions.



2. **Leave blank the start time field (the end time, until when the future status will apply, is mandatory)** and click on "<CLICK HERE>" to select the status:

3. Select, by clicking the status button, the one you desire:



4. **Remember to leave blank the start time field (the end time, until when the future status will apply, is mandatory)**. You can also schedule future statuses with the calendar.

5. Press the + icon.

The selected status is now programmed and will be automatically used to justify the next undetermined status (FAI). This automatic future status will be shown at the bottom of the window.

To cancel a programmed future status, just manually insert any other status.

> **NOTE:** To create new statuses or to change the actual ones, please refer to Status Configuration.

Consumption

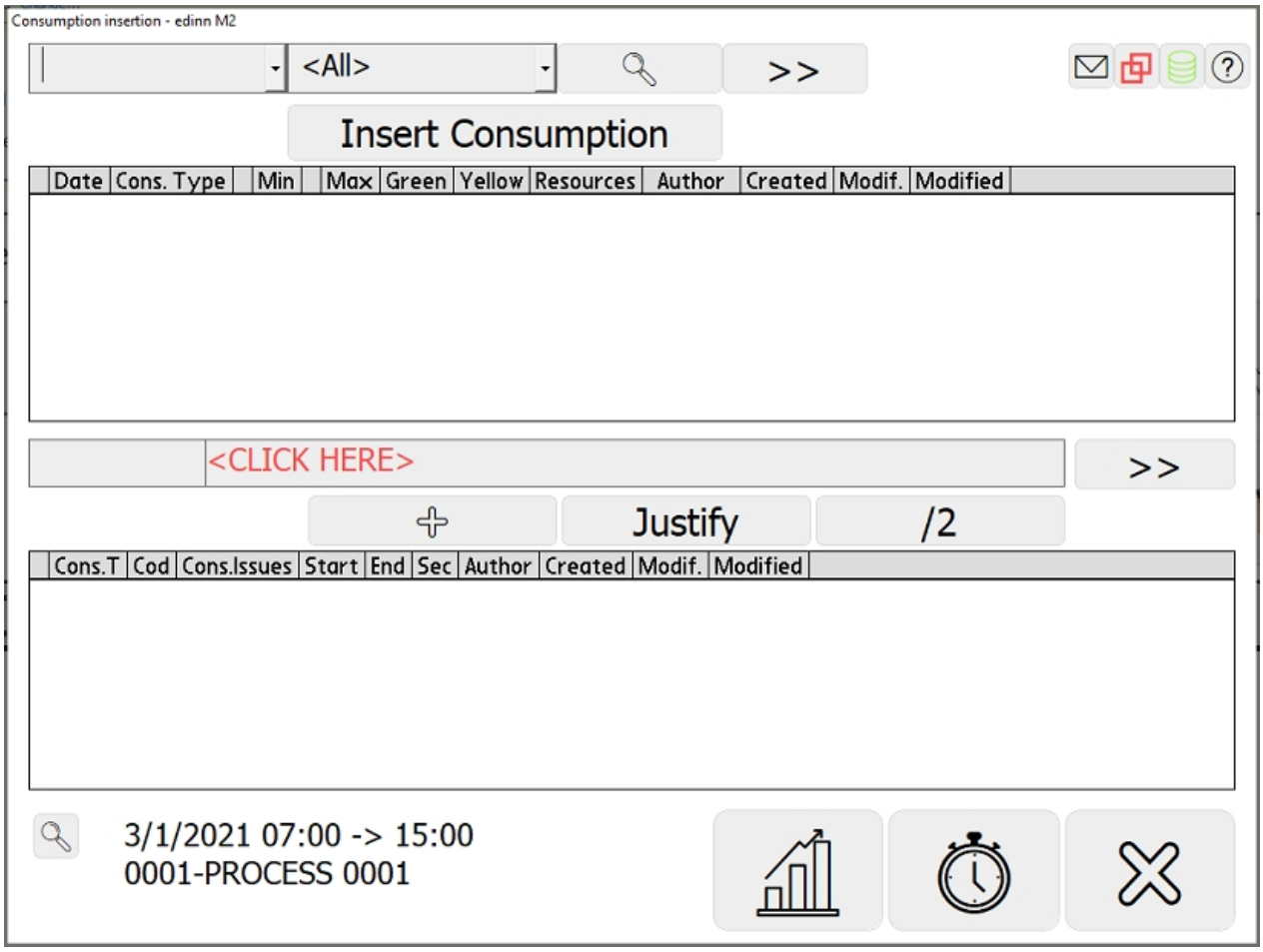
Main Window > Consumption

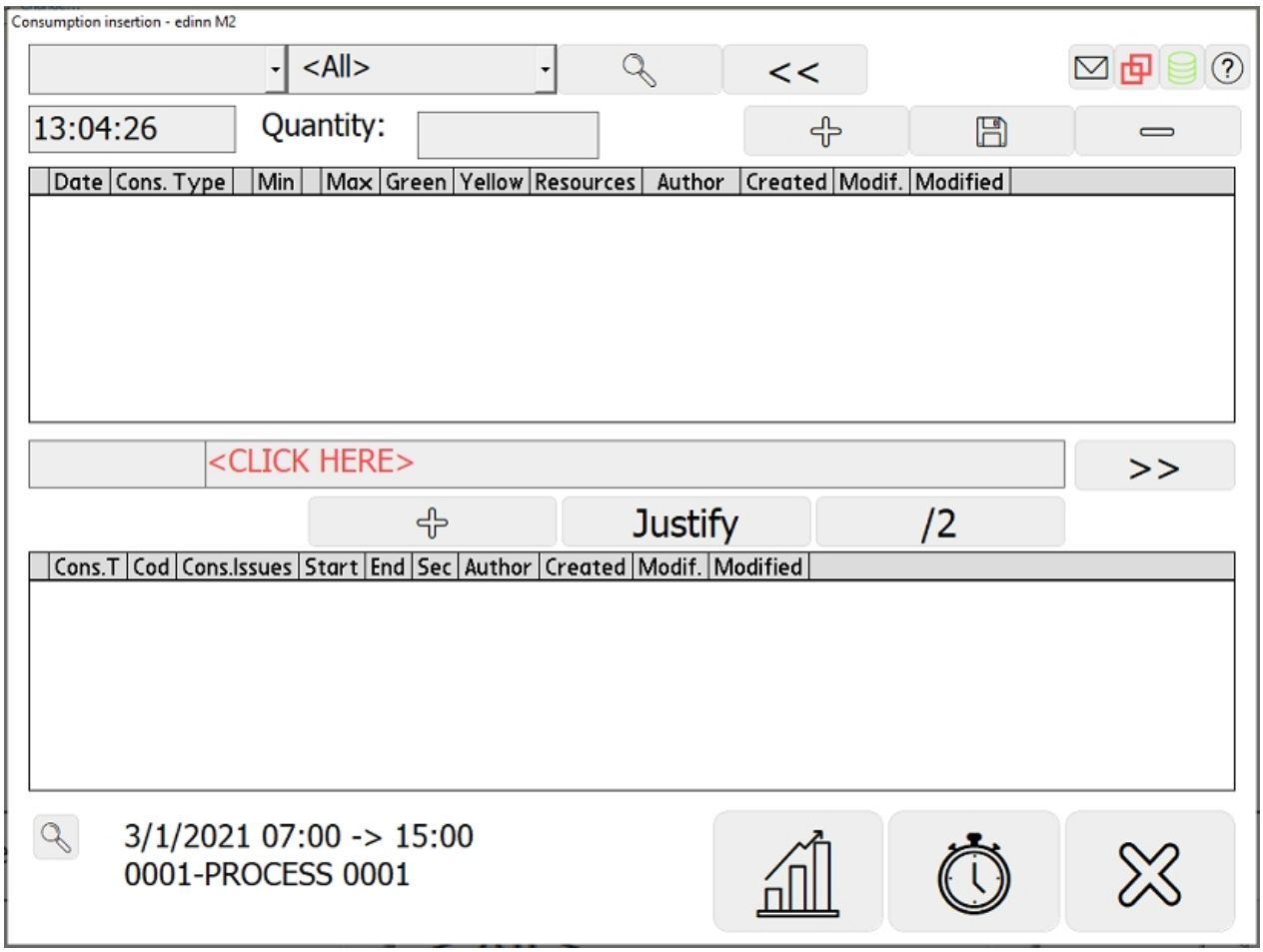
Utility: allows the user to view and manage the consumptions generated by the process.

If the process is not selected before clicking on the Status button, a selection window will be

shown:

Please refer to this document to know the difference between consumptions and inputs.





The top grid shows the consumptions that occurred. The bottom grid shows the different consumption statuses and the justifications selected by users. This is because in case of excessive consumption and if the system is specifically configured, a FAI record of excessive consumption will be generated at the bottom grid which will need justification from the user. The procedure to justify is identical to the procedure to justify statuses.

The detail of this window is as follows:

- **Selector of type of consumption**: selects the type of consumption you want to view and manage.
- **Selector of person**: consumptions can be associated to a person.
- **Search button**: updates the grids.
- : advanced editing functions.
- **Insert consumption button**: allows to manually insert a consumption.

The advanced editing functions will show a quantity of consumption and hour time field, to manually enter this data.

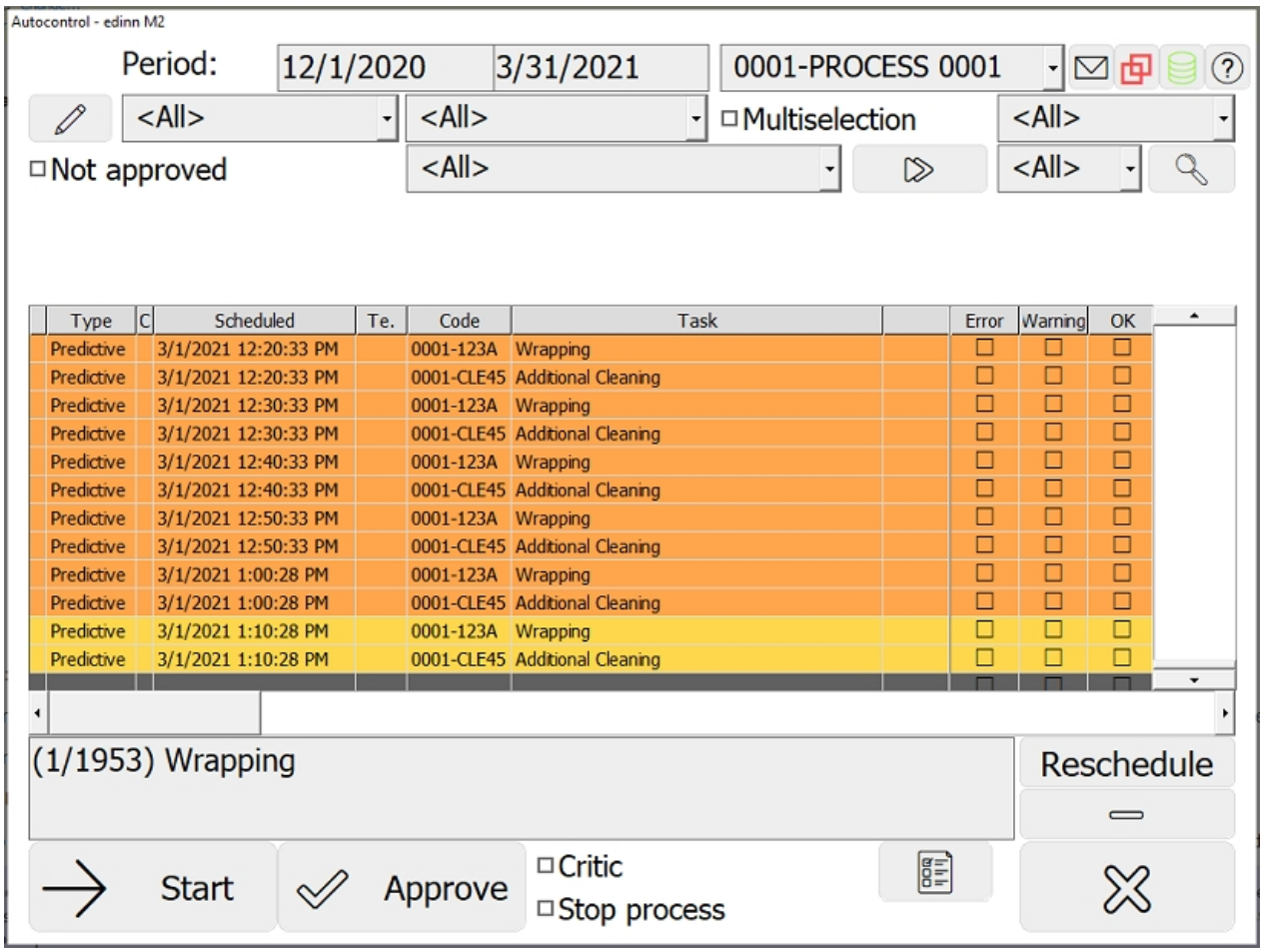
ATTENTION: Some functions, like for example deleting production registers of which you are not the author, or changing the date times, or type of product, can only be used if the administrator assigned

you these roles in the personnel configuration.

Autocontrol

Main Window > Autocontrol

Utility: allows you to **view and manage** the procedures and maintenance tasks of the processes.



AUTOCONTROL TASKS COLORS:

- **Green:** tasks that should be carried out at this moment, because they are in their Due Date, and only for those users who belong to the team associated with this task.
- **Yellow:** tasks that have still a margin before carrying out.
- **Red:** tasks that have to be carried out as quick as possible.
- **Orange:** tasks that were not done because they are out of tolerance, or tasks that are expired. Further actions (for example sending of notifications to

supervisors) could have been automatically taken by the system.

NOTE: The tasks which, in their configuration, a quantity different than zero has been indicated in any of its 'Every' fields, will be automatically converted to status 'Not done' when they are still pending and the time indicated in the tolerance (+) has passed after the due date. They will be later converted automatically to status 'Expired' when they are 'Not done' and another task of the same Id is created. The tasks which, in their configuration, all 'Every' fields are zero, will never be automatically converted to status 'Not done' but will be automatically converted to 'Expired' when a new one is created with the same Id.

AUTOCONTROL TASKS FILTERS: (from left to right and from top to bottom):

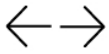
- **Period selector:** to select the period **Since -> To**.
- **Process selector:** to filter by process.
- **Team selector:** to filter by team.
- **Type of task selector:** to filter by type of task: *Predictive, Preventive, Corrective* and *Procedure*.
- **Periodicity selector:** to filter by the periodicity of the autocontrol tasks: *Daily, Weekly, Biweekly, Monthly...*
- **Not approved check:** to show only those tasks that are not approved by the representative of the line.
- **Resource (person) selector:** to filter by person.
- **Status selector:** to show autocontrol tasks with a specific status: *Pending, OK, Warning, Error, Not done and Expired.*
- (Press >>) **Personalized selector:** selector configured by the administrator that filters by a specific task field, for example: *time used, comments operator, comments engineer...*
- (Press >>) **Blank selector:** text to filter by a certain text. Indicating <NN> allows to search for those tasks that have a **N**ot **N**ull field somewhere. It is possible to combine <NN> with a specific field.

AUTOCONTROL TASKS ORDER:

The tasks appear **ordered** according to the following criteria:

1. **By due date.**
2. **By team.**
3. **By colour**: red, green and yellow.
4. First the **critical items** followed by the **non critical items**.

AUTOCONTROL TASKS FUNCTIONALITIES AND BUTTONS:

-  : to insert a comment on an autocontrol task.
-  : to navigate through the different comments.
- **Status selector:** to modify the status of an autocontrol task, for example, after performing it. **How?:**
    1. **Select** an **autocontrol task** on the grid.
    2. **Select a status** between the columns: *Error*, *Warning*, *Ok*.
    3. On the window that will appear, **select** the **end time** of the autocontrol task.

NOTE: If the user does not have the **AutoCtrl. Superv.** role, the status of those tasks that are not performed and that have passed the tolerance cannot be changed. The only information that can be changed is the information in the fields.

-  : to refresh the table according the filters.
- **Reschedule Button**: to reschedule the selected task to another due date and time.

NOTE: the Reschedule functionality is only available for users with the **Reschedule role**, which can be configured in Resources configuration.

- : to delete the selected task and all its fields.

NOTE: only available for users with the **Administrator role**, which can be configured in Resources configuration.

- **Start button:** to mark the selected task as started and to set the starting date and time.
- **Approve button:** to mark the autocontrol task as approved once it has been done.

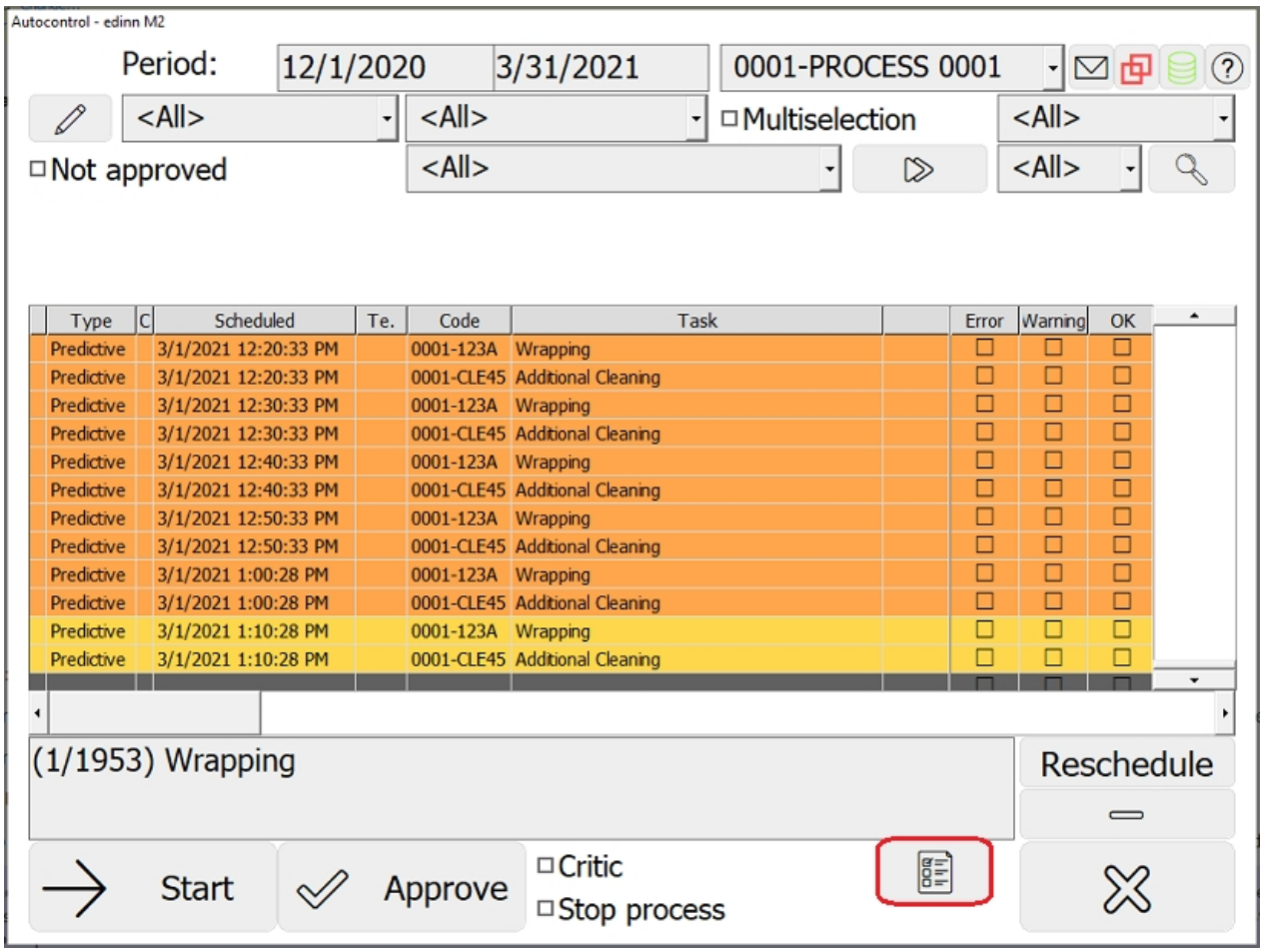
NOTE: this button is only available for those users who have the **Autoctrl. Super. role**, which can be configured in Persons configuration.

- **Critic:** to indicate that the autocontrol task is critical.
- **Stop process:** to indicate that the machine needs to be stopped before performing the task.
-  : to change the visualization to the step-by-step autocontrol.

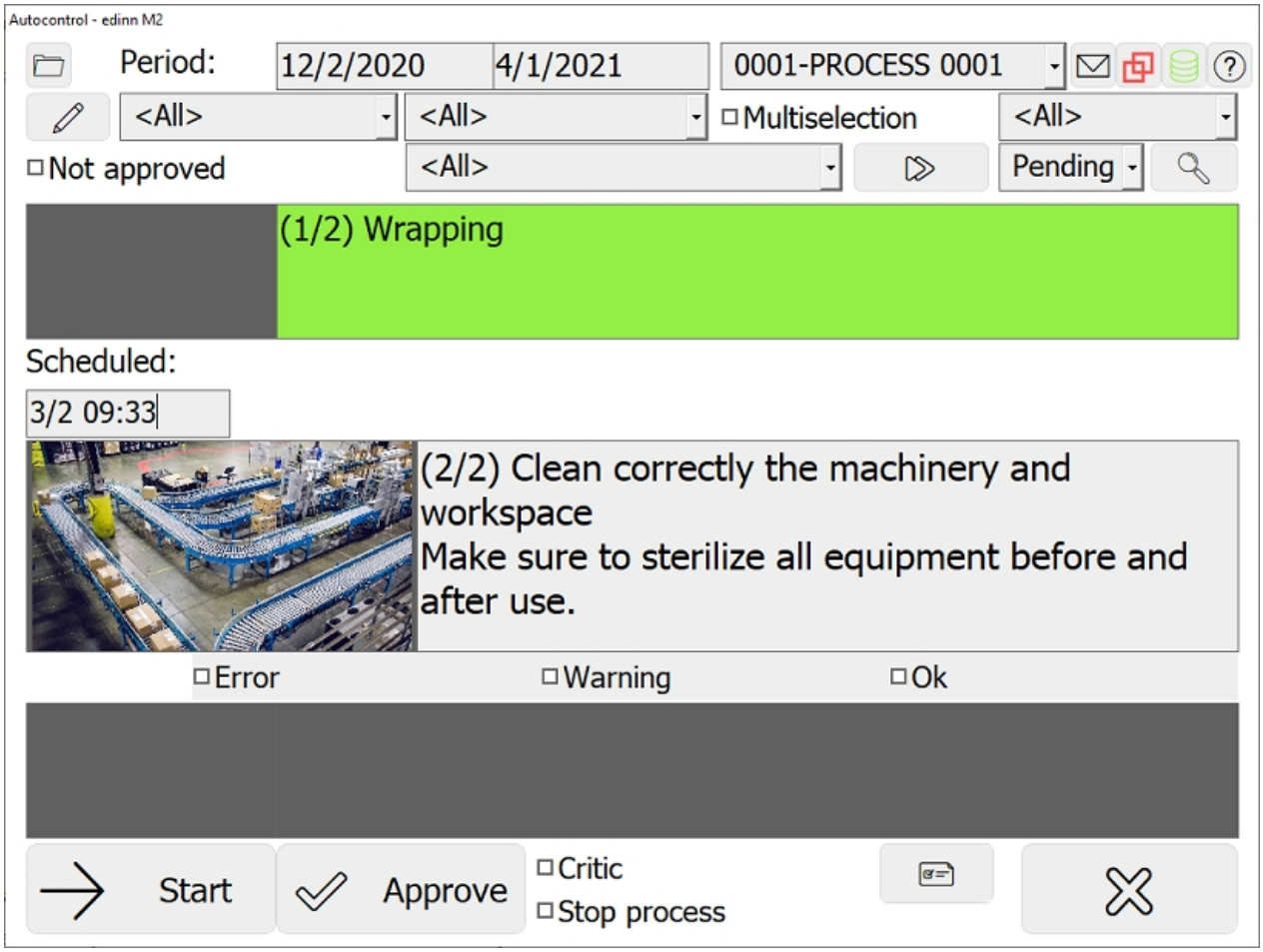
Step-by-step Autocontrol

Autocontrol task could be managed also using a **guided form**. This allows faster management and guidance step-by-step.

Enter the **Autocontrol window** and click the following icon to switch from the table view to the **form view**:

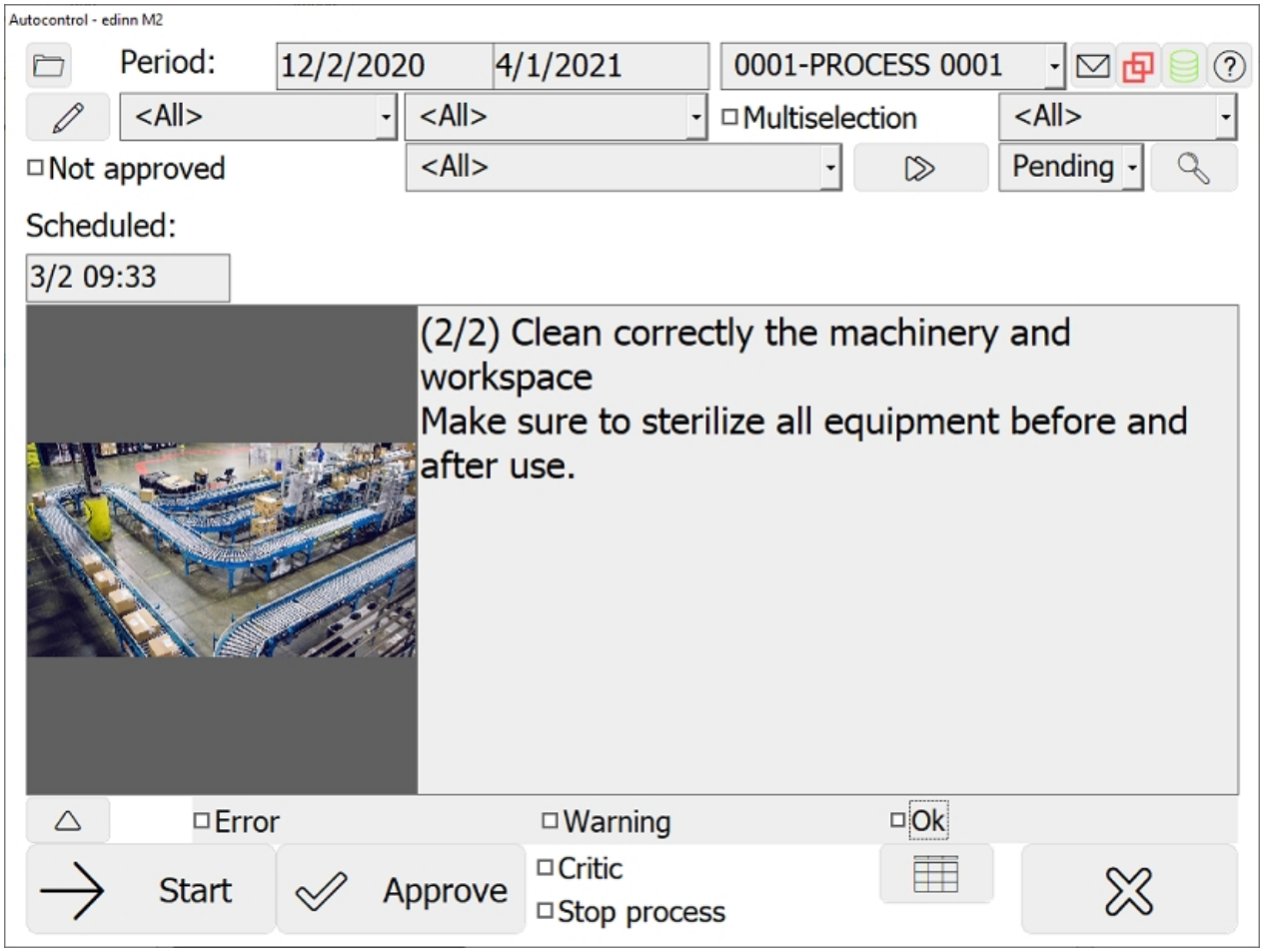


You will see the following window:



Here you will see the **previous**, the **actual** and the **future** autocontrol task (in that order), by selecting "**Error**", "**Warning**" or "**Ok**" you will define the status of the current autocontrol task.

To see only the current autocontrol task, click the following icon to switch from the table view to the **task view**:



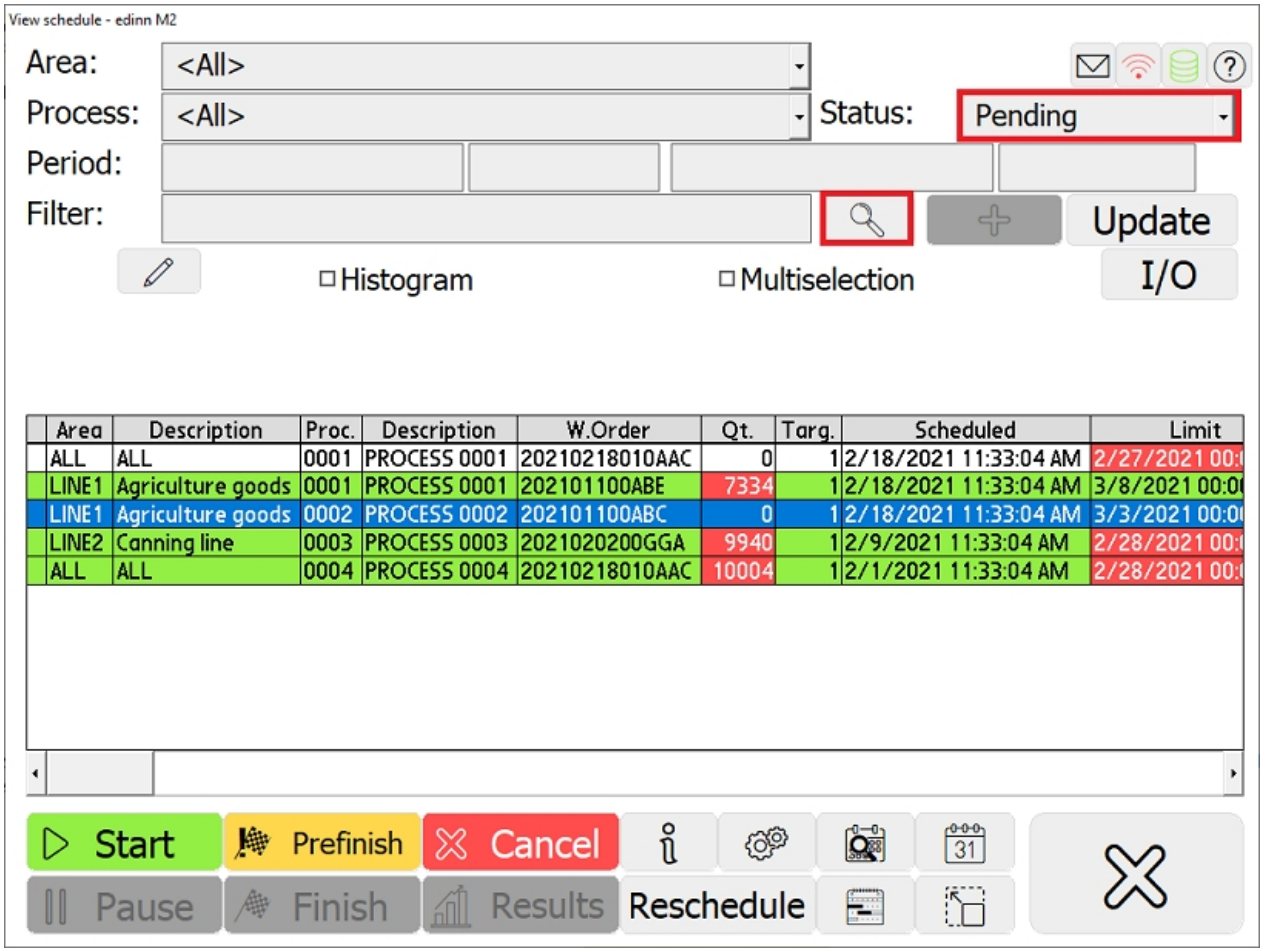
Here you will see the **actual** autocontrol task, by selecting "**Error**", "**Warning**" or "**Ok**" you will define the status of the current autocontrol task.

Procedures

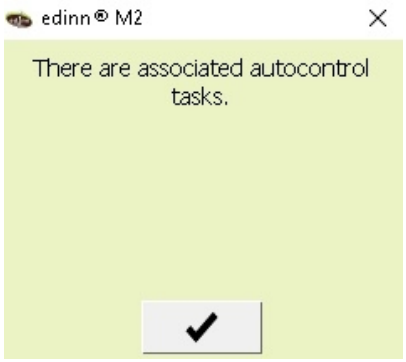
How to filter by working order

The autocontrol tasks can be configured to be **automatically launched** with the start of a **new order** in the **Event** field of the Autocontrol configuration.

This configuration will notify that are **pending autocontrol tasks** to be done each time a new order is started. If you enter the Schedule Window () in the Terminal, and search for the **pending orders**, you will see the orders that are not yet started:



For example, if we want to start the order **202101100ABC** we will have to **select it** and press the [Start] button, the Autocontrol window () will be opened with the Autocontrol tasks automatically generated for that specific order and you will also receive the following message:



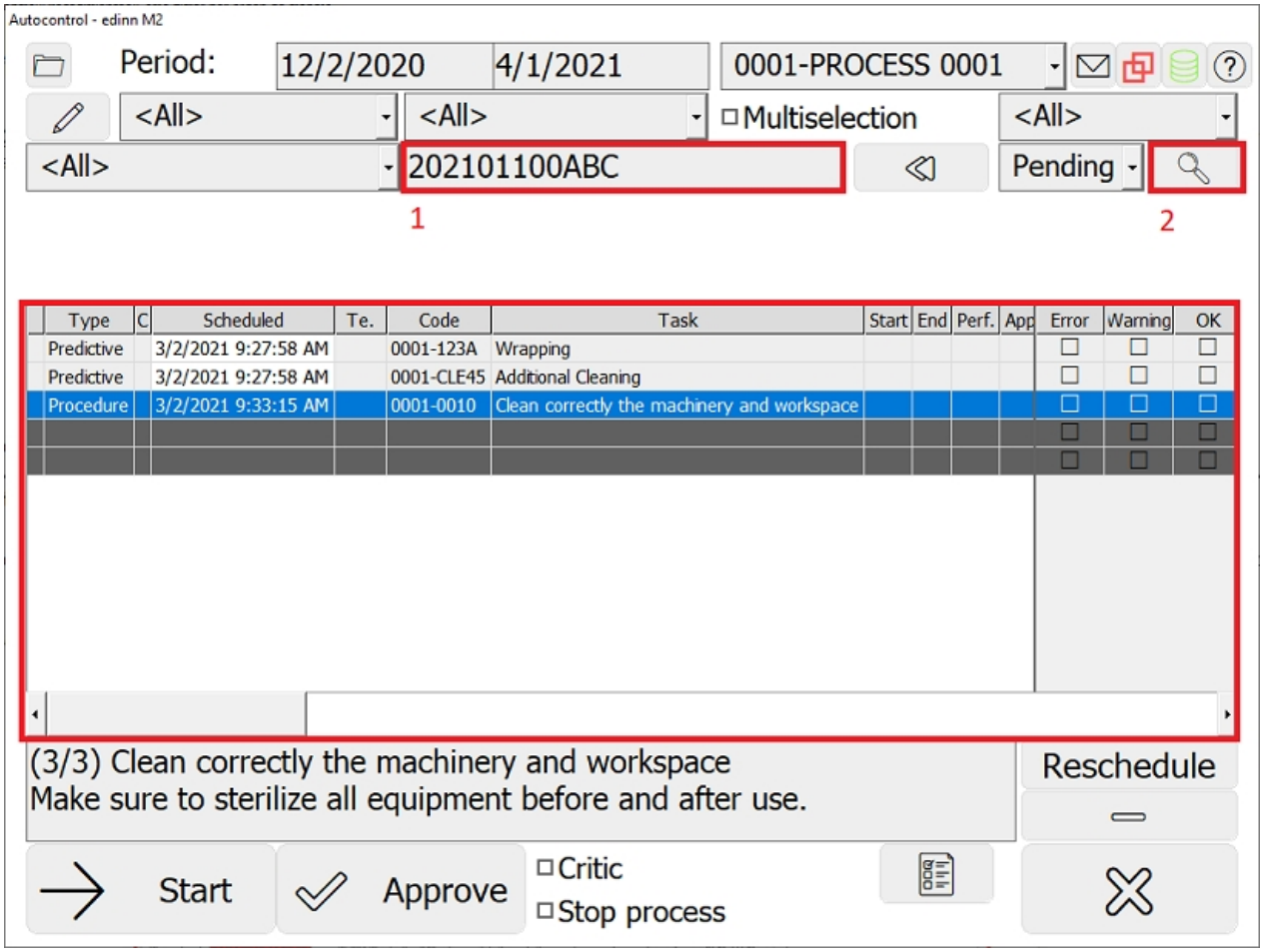
Furthermore, you can **filter** the grid of autocontrol tasks, done and to be done, for specific **orders**. This **filter** is available in the following windows of the **Autocontrol** functionality:

1. In the Autocontrol window of the Terminal

In this window the autocontrol tasks, done and to be done, are shown. To show the specific

tasks for a certain order press the button, fill the blank field with the **code of the order** and then press to search.

For example, the 2 autocontrol tasks for the order **202101100ABC** are:

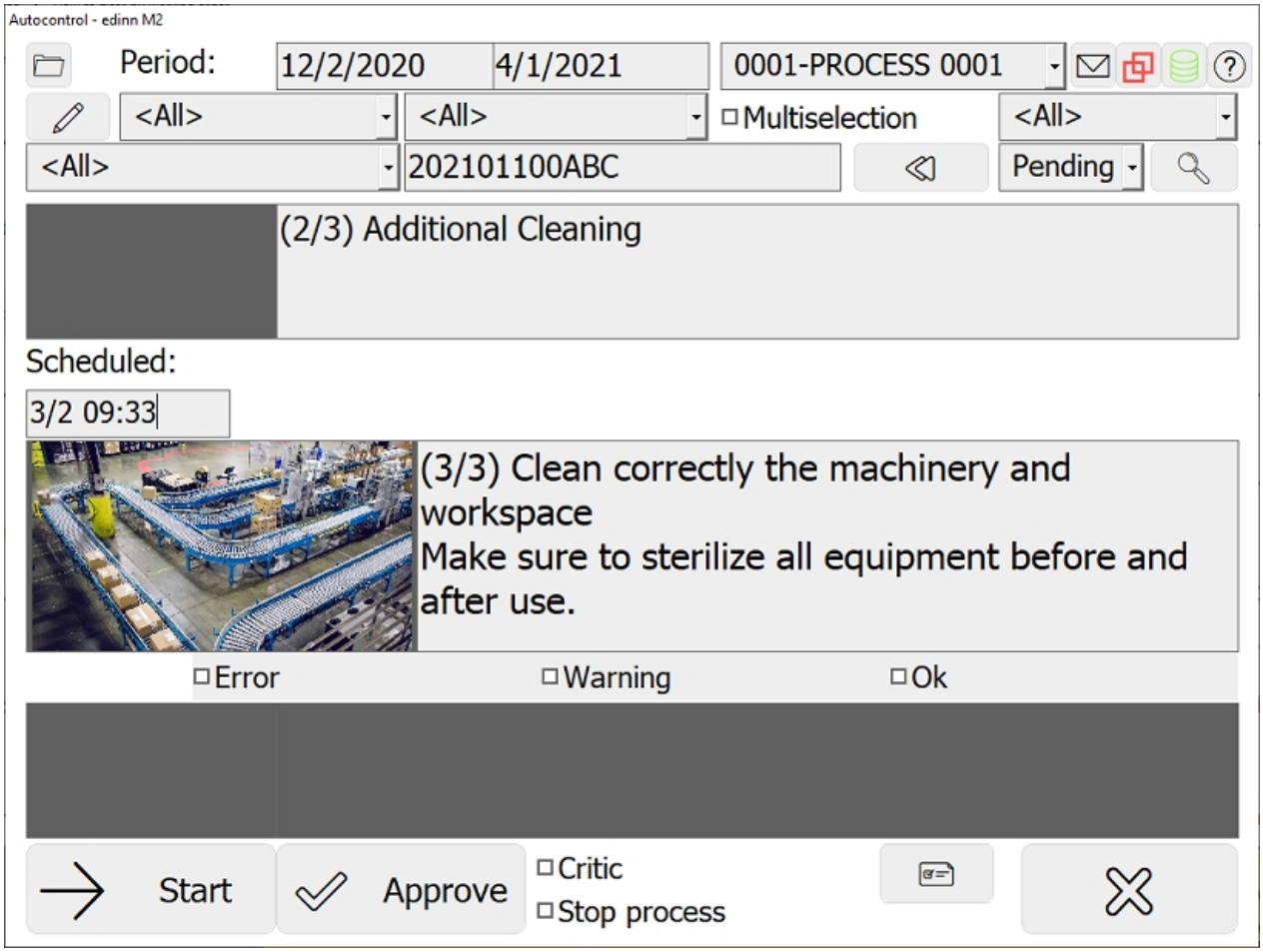


2. Step-by-step

In this window the autocontrol tasks, done and to be done, are shown, in a **step-by-step** visualization. To show the specific tasks for a certain order press the button, fill the blank field with the **code of the order** and then press to search.

> **NOTE:** to see how to access to the Step-by-Step Autocontrol guide, please see: Step-by-step Autocontrol

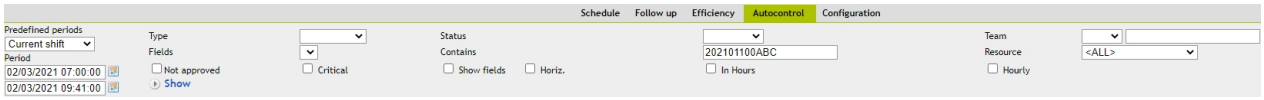
For example, to see the 2 autocontrol tasks for the order **202101100ABC**:



3. Autocontrol Reports

In the "**Efficiency**" and "**Autocontrol**" windows of the autocontrol functionality of the **Reports**, by filling the [**Contains**] field and pressing [**Show report**] you can filter by order.

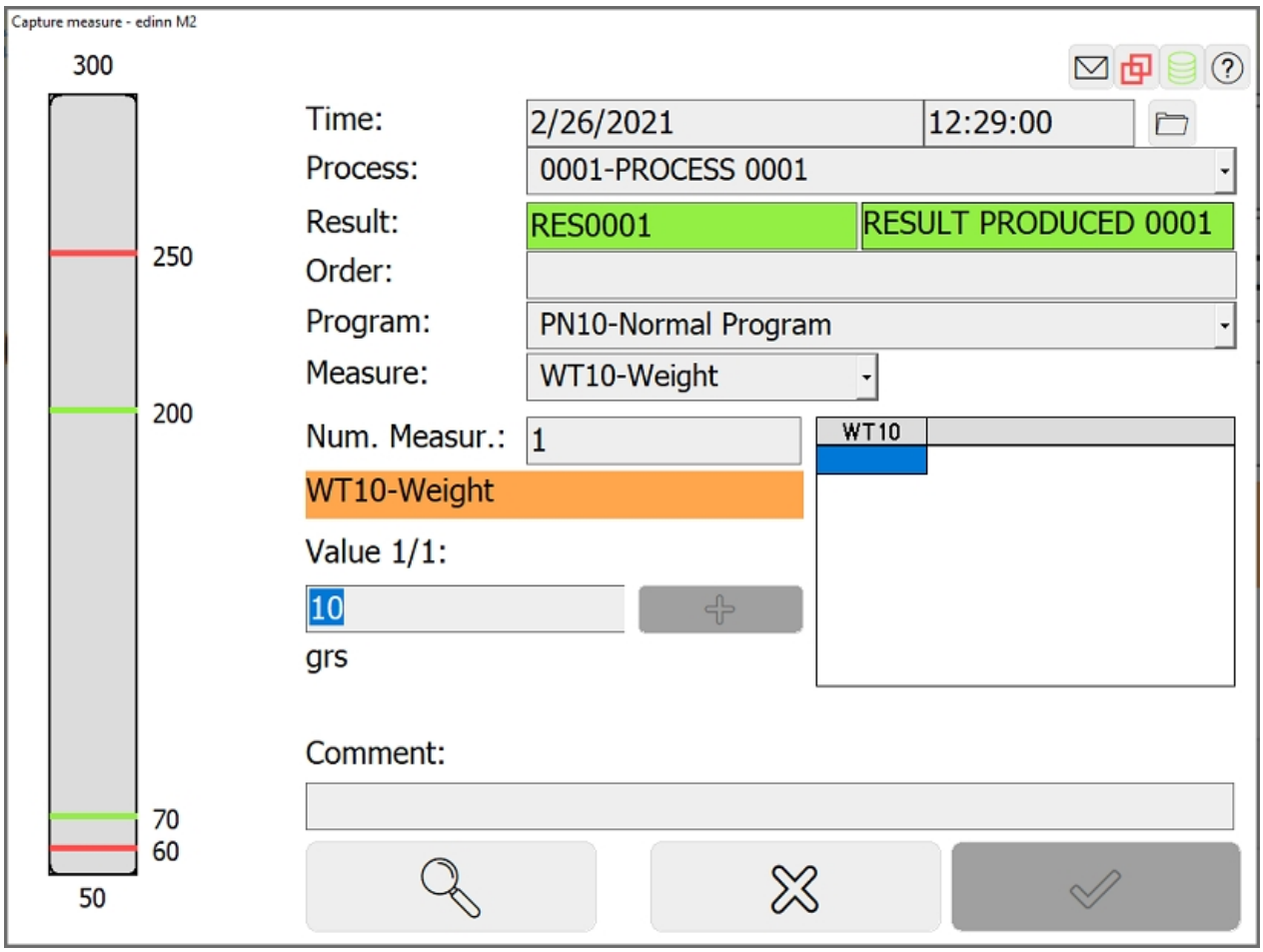
For example, to see the autocontrol tasks for the order **202101100ABC**:



Quality

Main Window > Quality

Utility: allows you to view and manage the data of the production process variables in order to perform the quality control by using the SPC (Statistical Process Control) methodology.



MEASUREMENT BAR:

On the left, it is shown a bar with the following parameters:

- **A scale:** (-1,1)
- **The average value:** the height of the bar.
- **Green lines:** the control limits.
- **Red lines:** the limits specified by the client.

NOTE: If the value is out of the scale, the whole bar will be shown in a grey color.

SPC SELECTORS (from left to right and from above to below):

- **Time selector:** select the period for which you want to impute SPC data.
- **Process selector:** select the process for which you are going to introduce the SPC data.
- **Result selector:** introduce the code or select the result (product) for which you are going to introduce the SPC data.
- **Order selector:** introduce the order for which you are going to impute the SPC

data.

- **Program selector:** select the SPC measurement programme.
- **Table:** in the table on the right, you will see the average values of each measure for the specified program. If there is only one measure, a vector will be shown. If not, the table will show a matrix.
- **Measure selector:** if the programme allows you to realize measurements for different measures, then we can select the measure we are going to take. If not, this selector will be blocked and the window will allow us to fill out a matrix to take the measurements, in stead of a vector.
- **Value field:** here we should introduce the **average value** of each of the measures (3 on the image). Click  to insert the value on the table.
-  **:** through this button we can find SPC data that are measured at other moments or for other processes.

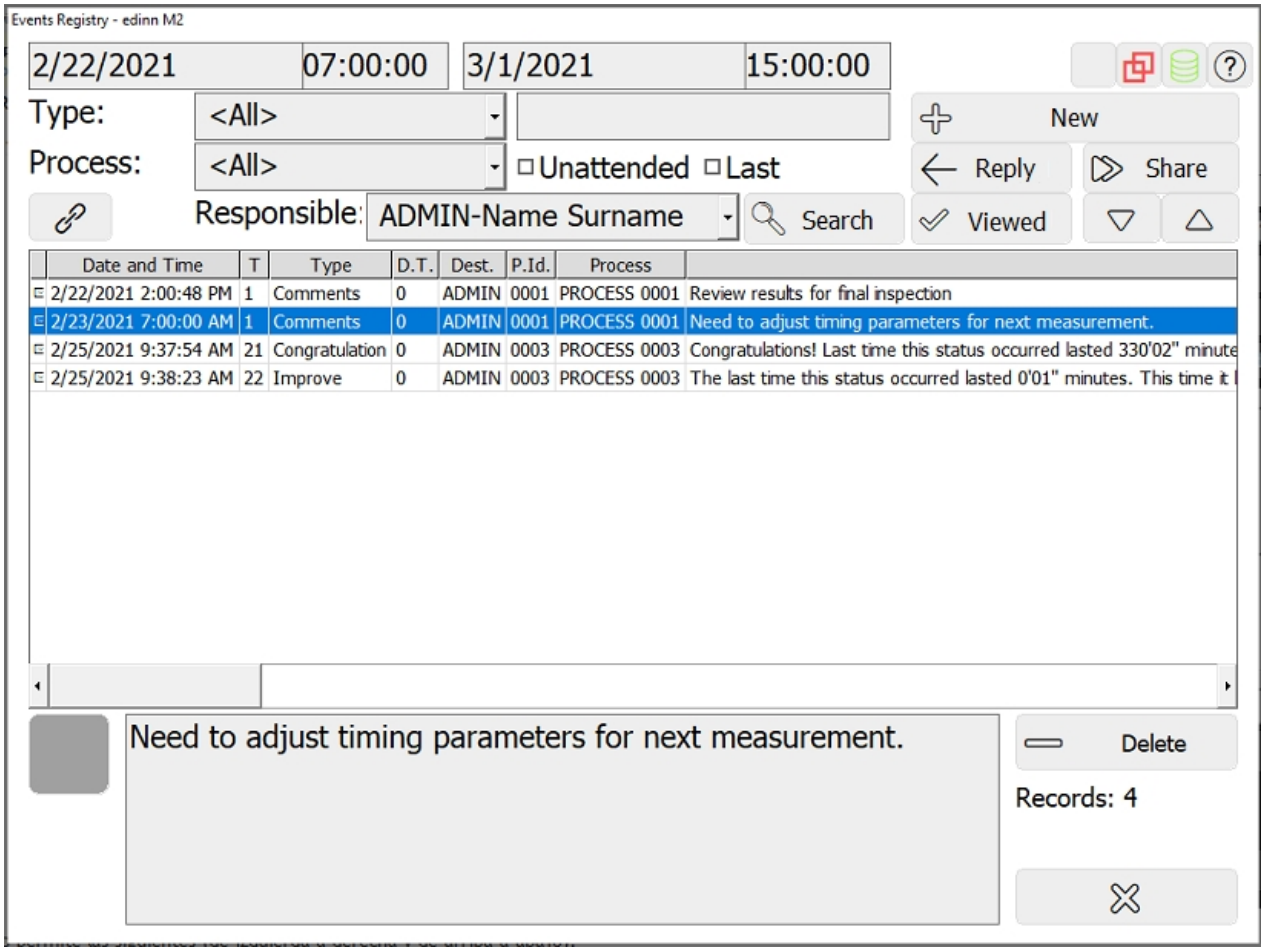
SPC OPERATIONS (according to the SPC standard):

- If **any value outside the control limits** is introduced, we will be asked to **justify** this and we can optionally leave a **comment**.
- If **7 or more consecutive points** are introduced in increasing or decreasing order, an **alert** will be shown and a **justification** will be asked.
- If **7 or more consecutive points** are introduced above or below the average between the upper and lower control limits, an **alert** will be shown and a **justification** will be asked as well.

Registry

Main Window > Registry

Utility: allows to view and manage the registry of the system. The registry of the system records a variety of events that occur in the system.



The registry window allows the following (from left to right and from above to below)

SELECTORES:

- **Period:** select the period you want to visualize by selecting a date on the calendar and introduce the start and end time.
- **Type**: select the type of information you want to see according to the available events.
- **Text:** find a text in the records and filter them.
- **Responsible**: to select a person to view his unattended records.

FUNCTIONALITIES AND BUTTONS:

- **Unattended check**: shows the records that have not been seen yet.
- **Last check**: shows only the last records.
- : to navigate to the related record.
- : to refresh the grid according to the filters.
- **New:** to create a new record.

- **Reply**: to reply to a record.
- **Share**: to share a record.
-  : allows navigating to unviewed messages.
- **Viewed button**: marks the record as “viewed”. This alert  stops blinking if we do not have any more records to view.
- **Delete**: to delete records (restricted to supervisors).

Reports

Main Window > Reports

Utility: edinn® M2 reports is the tool that provides all the information necessary to control and improve efficiency.

Once you click on the button you will be **redirected** to the reports window on the **web server**.

> **NOTE:** For more informaton, please see: Reports

Configuration

Main Window > Configuration

Utility: this window allows you to configure the system.

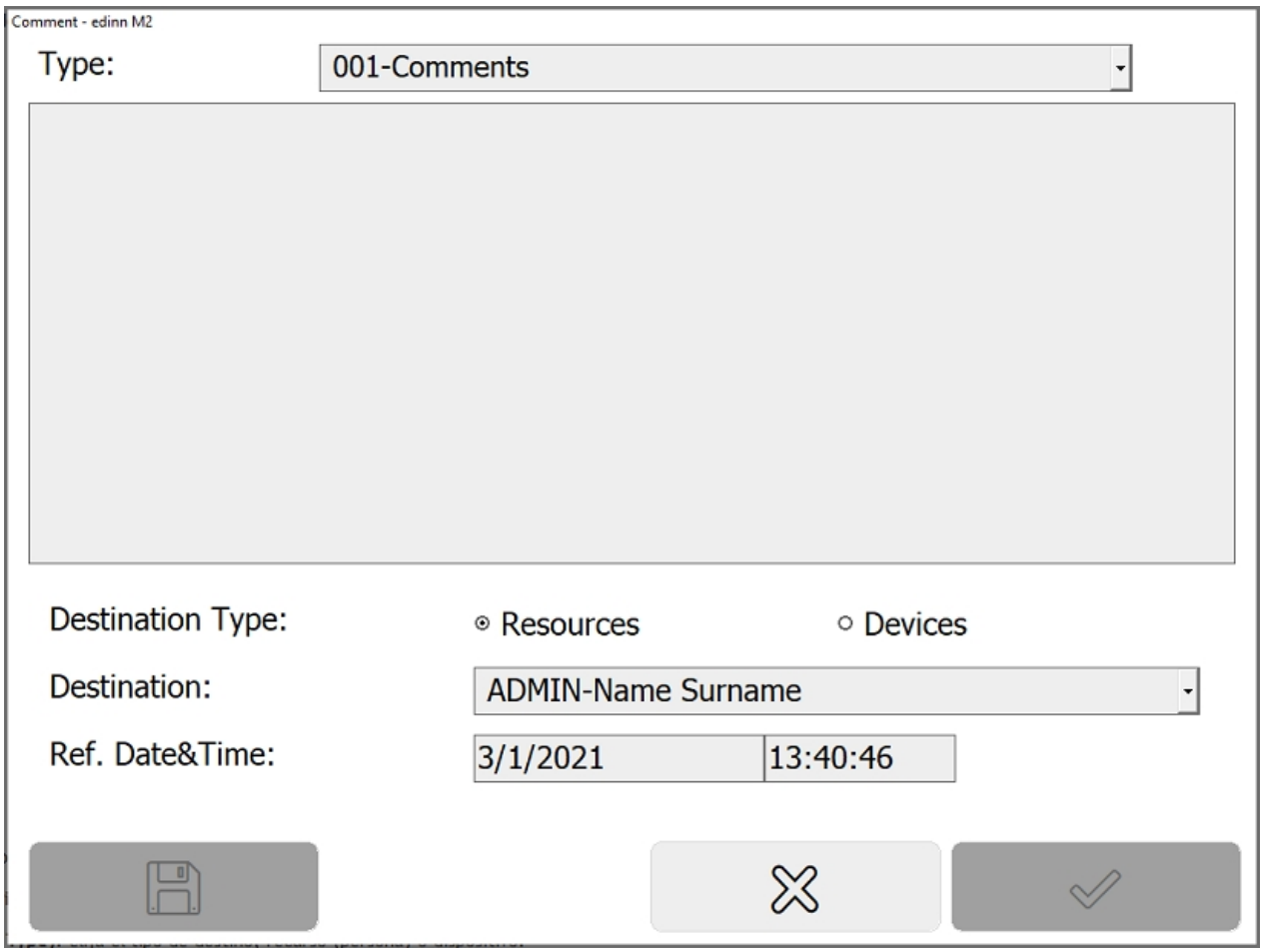
> **NOTE:** For more informaton, please see: *Configuration*

Other

Comment

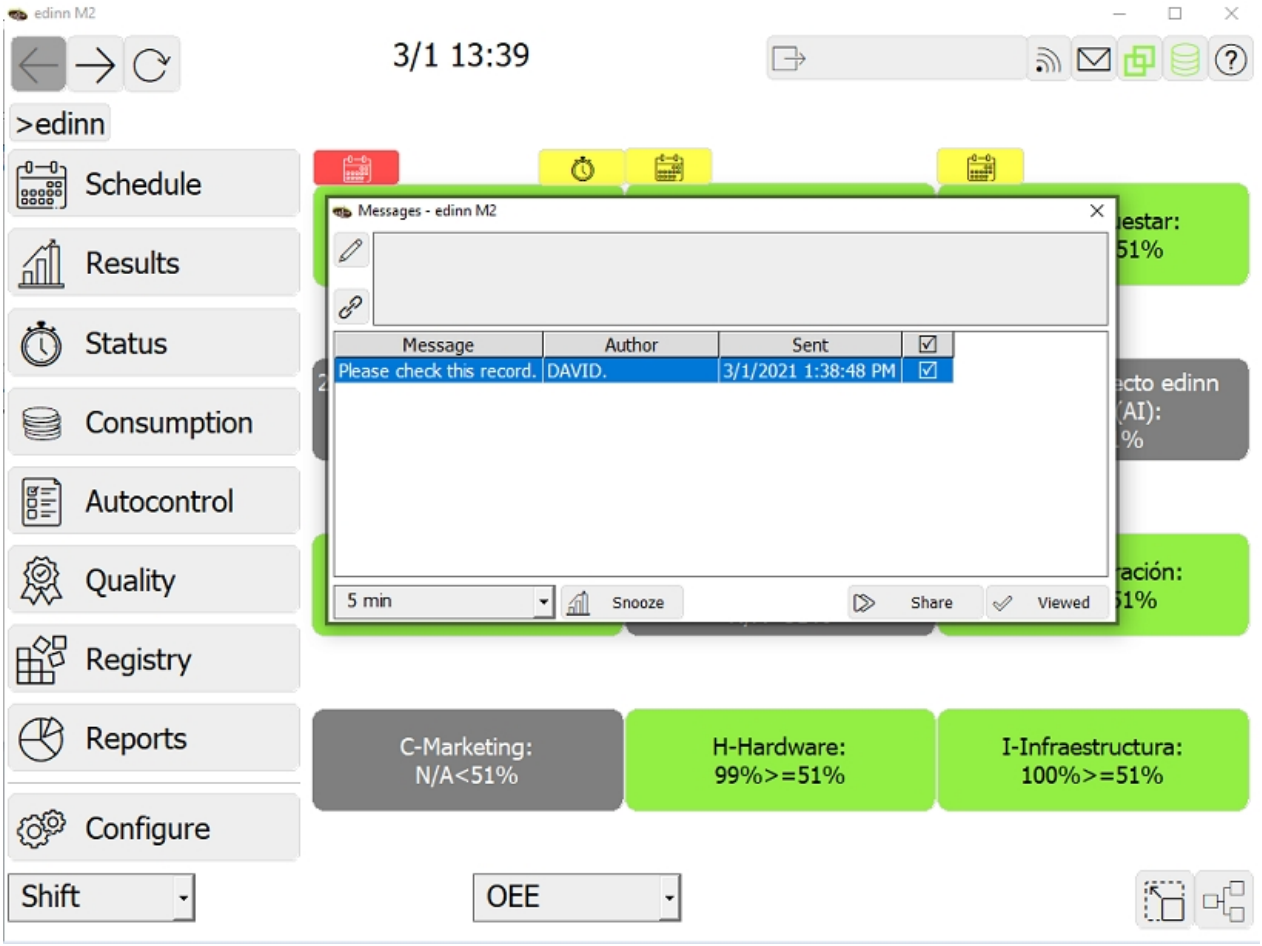
This window can be accessed by double clicking on any record of the data grids of the View schedule, Results, Statuses, Consumption and Autocontrol windows.

Utility: to make a comment of a record.



FUNCTIONALITIES AND BUTTONS:

- **Type**: choose the type of comment. For example: time of retention, time of delay, speed of impact...
- **Blank space:** type your comment here.
- **Destination Type**: resource (person) or device.
- **Destination**: destination.
- **Ref. Date & Time**: date and time referenced.
- : saves the message and, if there exist a destination person, sends a notification to the destination person. If correctly configured, the system will also email the destination person.
- : cancels the comment.
- : if the destination person is logged into the system, a pop-up will appear with pendant messages. If not, the message will appear when the destination person enters the system. The following window will appear:

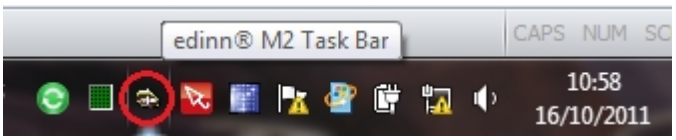


- **Snooze:** to delay the appearance of the pop-up for a certain time (5/10/15/30 mins, 1h)
- **Viewed:** to mark the message as seen or attended.

Edinn M2 Task Bar

When you execute the edinn M2 terminal, an edinn icon will appear at your windows task bar. That is the edinn M2 Task Bar.

> NOTE: the task bar will only appear with the win32 version installed and executed locally.



If you click on it, a pop up menu will show, with these options:

- **Show**: will show the edinn M2 Terminal application.
- **Supervise (selected by default)**: supervises that edinn M2 Terminal application is

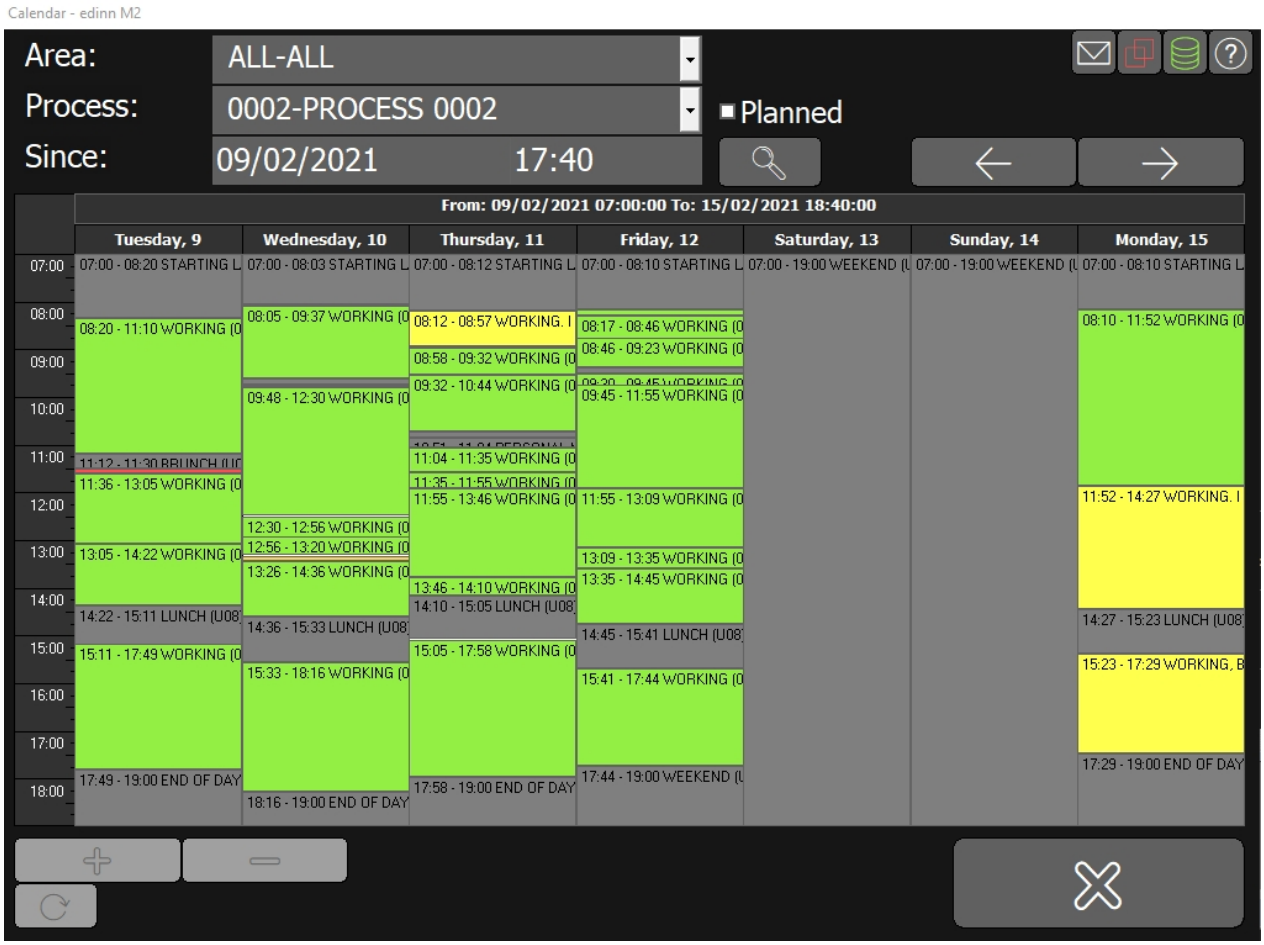
running correctly. If it detects that the application is stopped or not working correctly, it will offer to the user the option of restarting it. Please note that sometimes edinn M2 Terminal application is running to obtain lots of data and the "Restart" pop up will appear although the edinn M2 Terminal is working correctly. The user must use the restart only when he is sure that the application is not working correctly.

- **Ignore**: do not supervise the status of the edinn M2 Terminal application.
- **About**: shows the about of the edinn M2 Task Bar.
- **Close**: closes the edinn M2 Terminal application and the edinn M2 Task Bar.

Calendar

This window can be accessed from the Edit schedule window, Statuses configuration window and the Statuses window.

It shows the available-to-produce calendar which is automatically calculated by the system based on the statuses configuration. All the statuses which are not of type 'unscheduled' nor 'dependency' are considered periods where production is possible. Therefore, this is the calendar of availability used by the whole system, for example, by the automatic scheduler.



SELECTORS:

- Area.
- Process.
- **Planned**: indicates to show the scheduled (future) calendar or past real statuses.
- **Since**: date and time period of the start of the week.

BUTTONS:

-  : previous and next week.
-  refreshes the grid according to the filters.
-  requests to the server to update the calendar. Once activated, on the right of this button will show the progress of the update. Until the update is completed, the calendar could change. Normally, it is not necessary to use this function as the server will automatically update the calendar every time we change the configuration of the statuses, and also every day. The calendar is calculated for as many future days as indicated in the parameter 'Real time oldest record' of the server Console.
-  schedule a future status. If process is left blank, it schedules the status for all processes in the area.
-  delete a future scheduled status. If process is left blank, removes the status of all processes in the area.

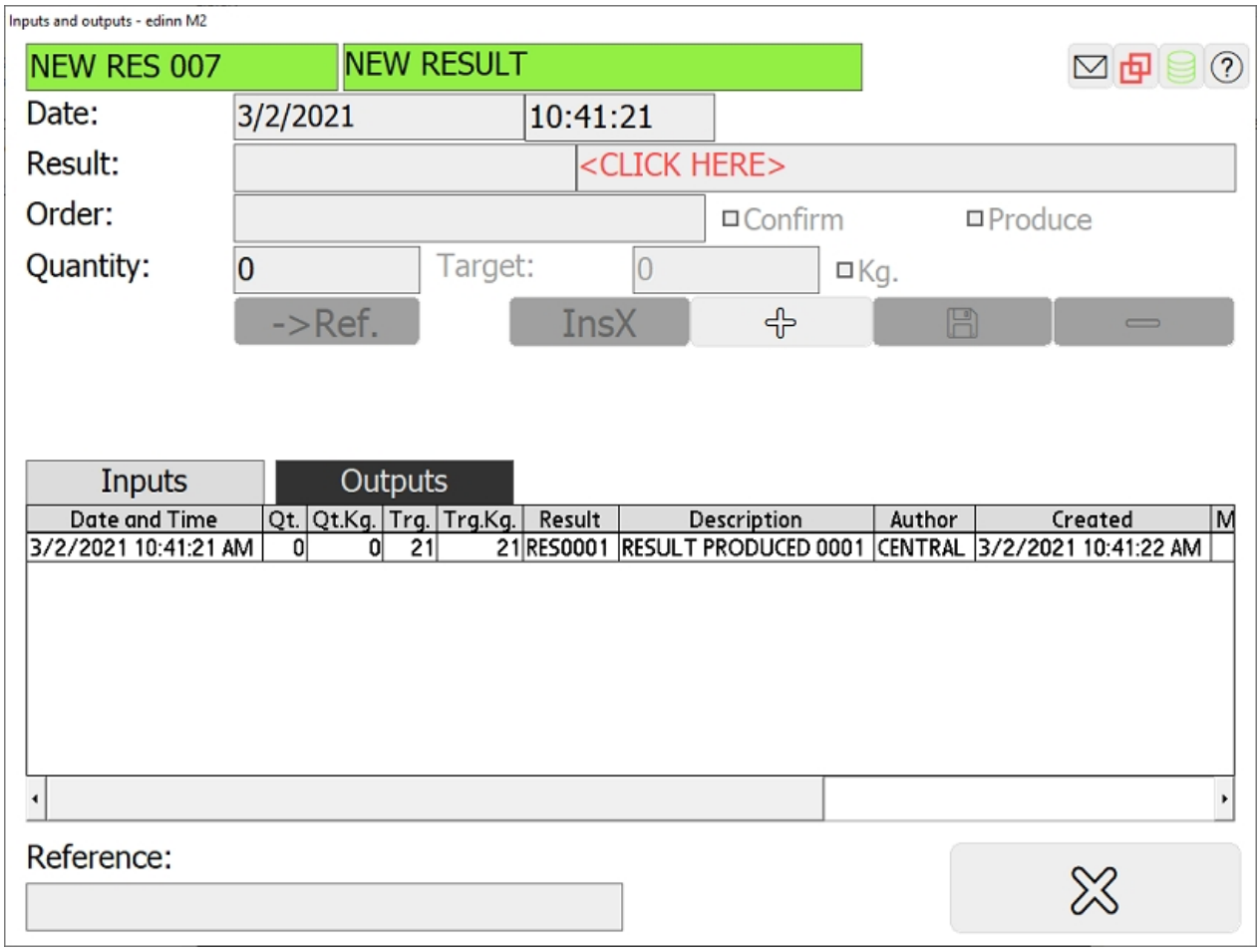
WARNING: statuses that are scheduled by the automatic scheduler cant be deleted, as the automatic scheduler will create it again when the calendar is updated.

I/O window

I/O is the abbreviation of Inputs/Outputs of the process. These are the elements consumed (inputs) and produced (outputs) by the production process. You can access here from Schedule and Results windows. Inputs and outputs are only needed when:

- You need to schedule considering the list of materials (*Bill of Materials*), structure, recipe or similar.

- You need to establish a relationship between two results, for example: when I produce the result A, I consume X units of the result B.
- You need to create scrap or rework which, by definition, consume units of the good result.
- You need to indicate inputs and outputs into the results.
- You need to manage stock.



The detail of the window is as follows:

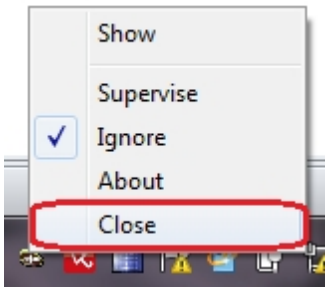
- **Result**: for edinn® M2, all inputs and outputs are also results.
- **Order**: the order that will be used to take the input or to generate the output.
- **Grouped check:** to group all the records that are equal.
- **Quantity**: the quantity that has been taken (input) or generated (output).
- **Target**: the quantity that has to be taken (input) or to be generated (output).
- **Confirm:** asks for confirmation about the selected item.
- **Produced:** indicates that the result was produced internally and not acquired.

- **Stock:** transfers you to the stock window, where you can see the stock you have of the selected item.
- **-> Ref:** allows you to change the reference or fields (e.g. batch, finish, etc.) of the selected result as input or output.
- **InsX:** press this button to enter more than one input or output (N times) to the grid.
- **According to results:** it will show the consumptions and productions actually produced. If it is not marked, it shows those provided in the order.

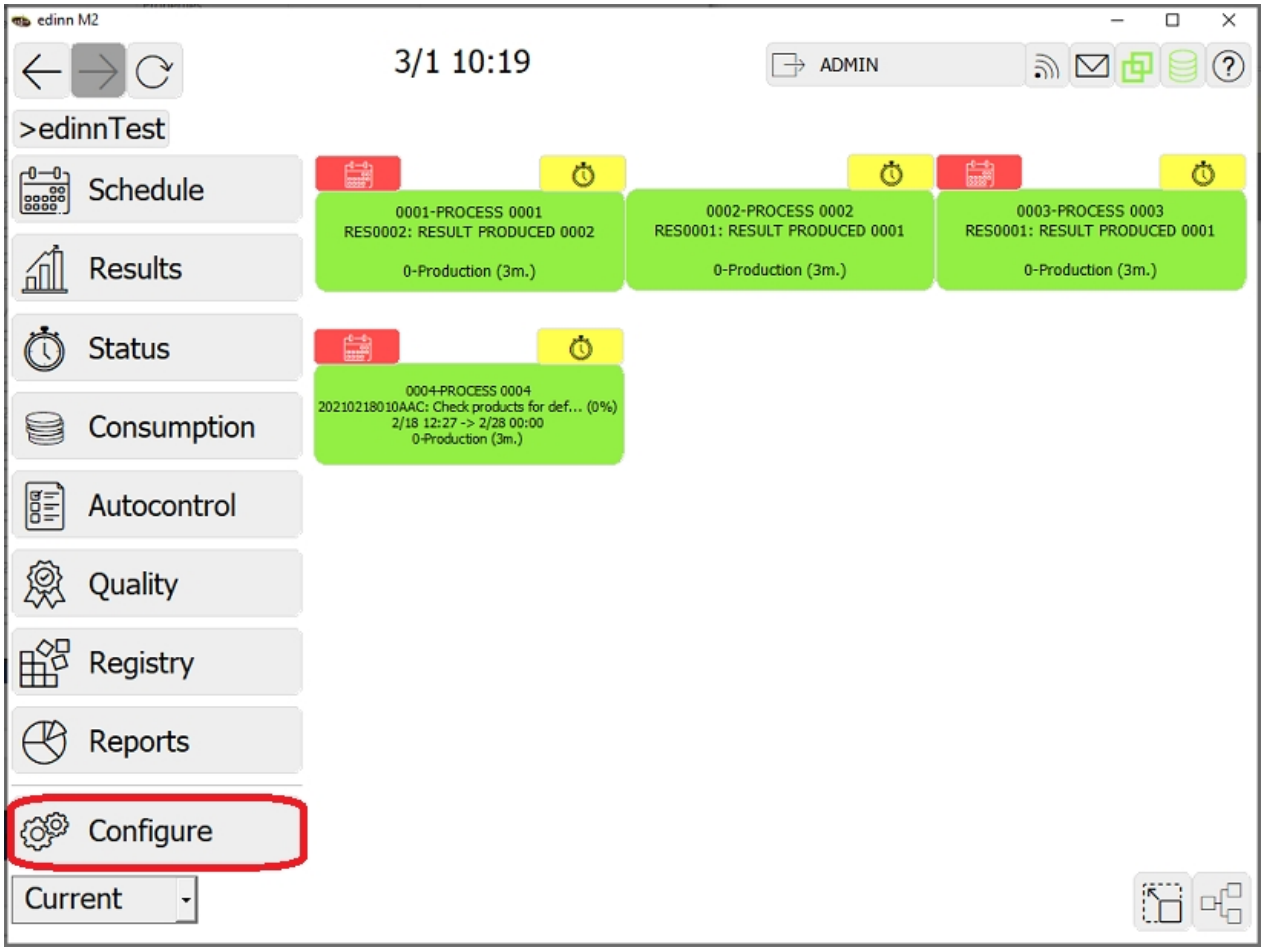
Exiting

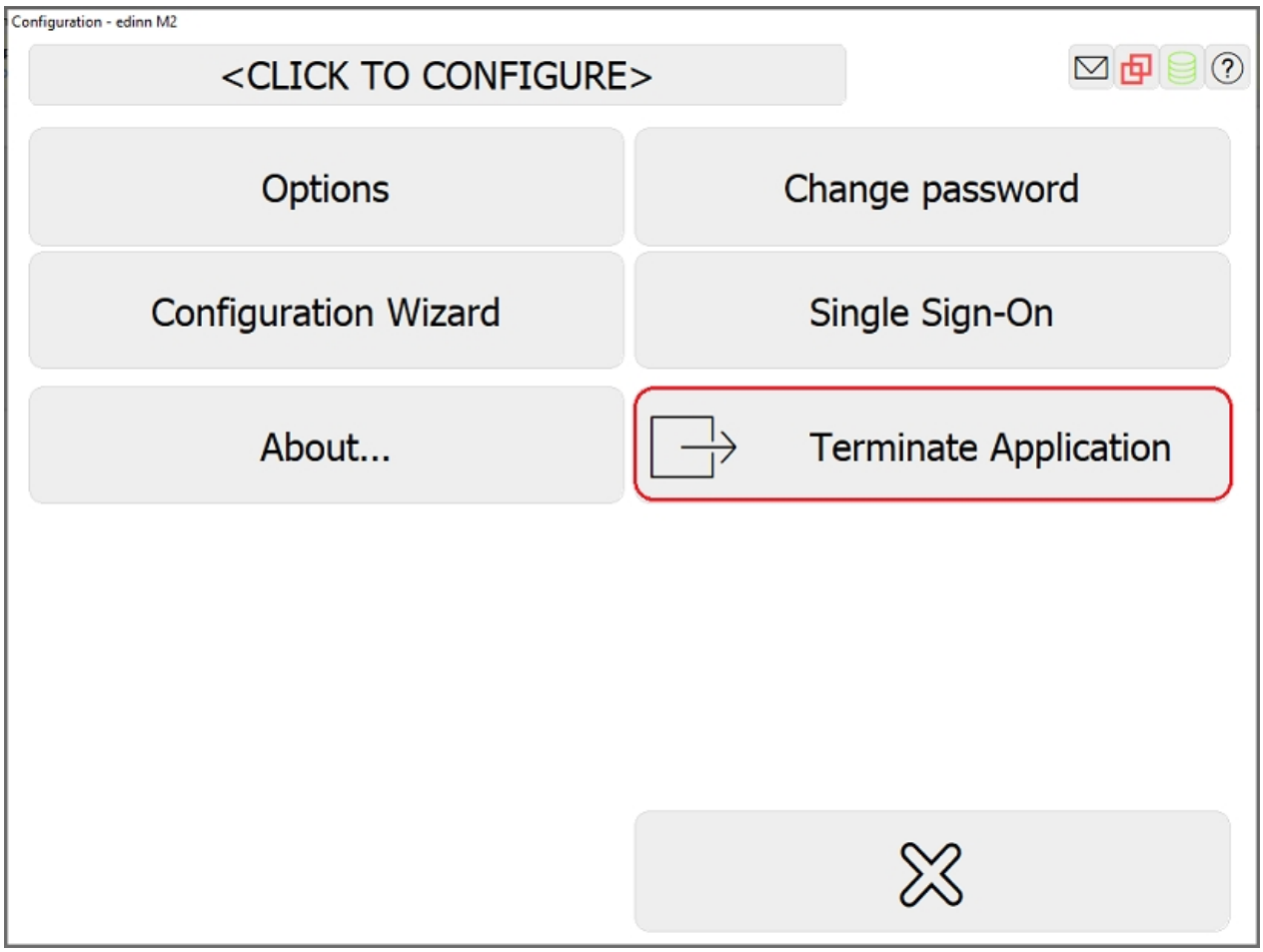
There are 2 ways to exit the application:

1. Click on the edinn icon on the edinn® M2 Task Bar, generally at the right bottom of your windows desktop, and choose "Close".



2. Go to the configuration window and click on the "Terminate Application" button. If the "Terminate Application" is blocked, only the administrator will be able to terminate the application.


edinn M2
3/1 10:19
ADMIN
>edinnTest
Schedule
Results
Status
Consumption
Autocontrol
Quality
Registry
Reports
Configure
Current
0001-PROCESS 0001
RES0002: RESULT PRODUCED 0002
0-Production (3m.)
0002-PROCESS 0002
RES0001: RESULT PRODUCED 0001
0-Production (3m.)
0003-PROCESS 0003
RES0001: RESULT PRODUCED 0001
0-Production (3m.)
0004-PROCESS 0004
20210218010AAC: Check products for def... (0%)
2/18 12:27 -> 2/28 00:00
0-Production (3m.)



Reports

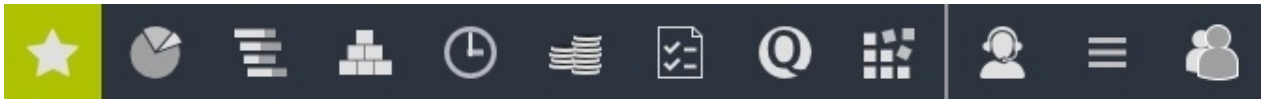
Welcome to the edinn® M2 Reports!

This tool provides all the necessary information obtained and processed by edinn.

For a quick overview of the reporting tool, please see the edinn Reports Basic Tutorial:

- In Microsoft PowerPoint format
- In PDF format

The reports are shown within these different sections:



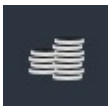
Favorites: this section allows to select and show the most frequently used reports in one screen.

Analysis: this section is used to analyze processes.

Schedule: this section shows the scheduled orders, their notifications and statuses.

Results: this section shows the evolution of the results done by the processes.

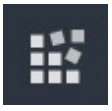
Status: this section shows the statuses through which a process passes.



Consumption: this section shows how the different changes affects the evolution of the consumptions done by the processes.

Autocontrol: this section perform a simulation and shows the autocontrol tasks that will be launched in the future, to predict the work load of tomorrow, next week, next month, a special period, etc.

Quality (SPC): this section shows the results of the quality measures of the processes.



Registry: this section reports all the events that have occurred in the system, such as a registry modification, a system crash...

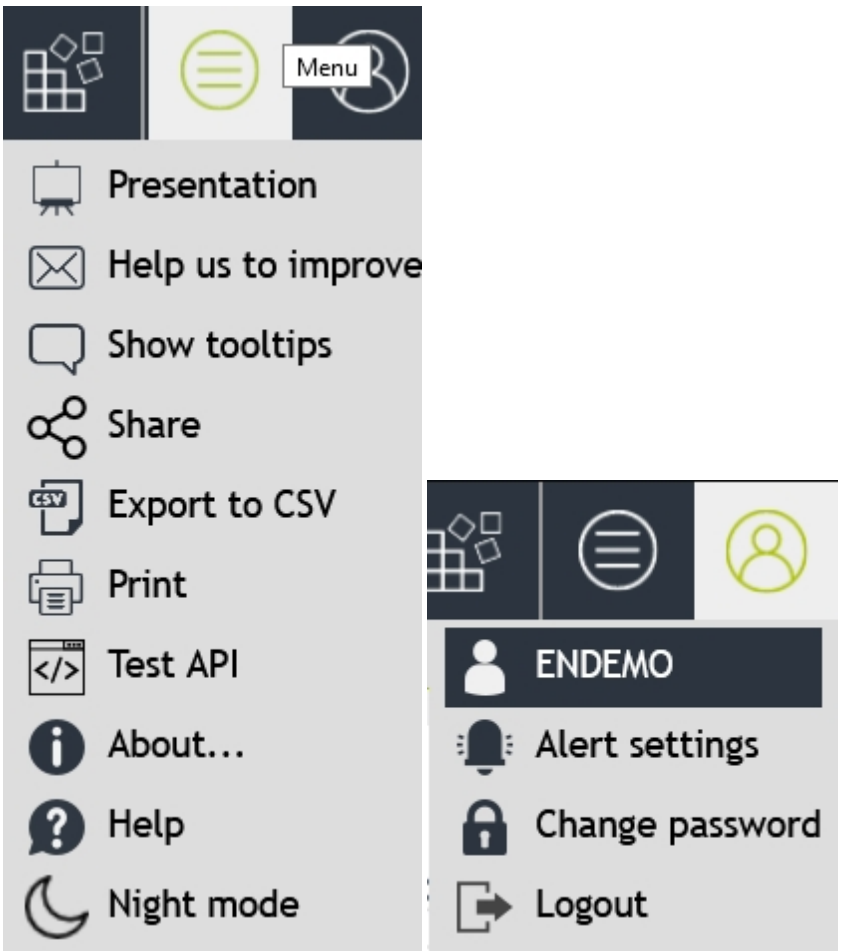
Common aspects

The reports have the following **functionalities**:

- **Graphic division by colors:** dividing the information by colors improves and makes easier the user experience. For example, if you want to find the losses, you will only have to look for the information in red.
- Some reports can be **expanded** by clicking on the areas.
- Visualizing different reports by introducing only once the **filtering parameters**.

- Different **types of visualization**: tables with data or graphics (pies, bars, 3D, etc.)
- Showing the **description** of columns of the tables (if you have activated the *tooltips* function), by passing the mouse over.
- **Comparing** different reports at the same time by opening different tabs or internet explorers.
- Possibility to **hide or show columns** in the reports. This functionality will only apply to the user that performed it.
- Possibility to **order** tables based on columns.

All the reports have some **common options** and **selectors**:



- **Presentation**: appears only when viewing dashboards. It activates the presentation mode, which consists in showing the current dashboard, at full screen, passing later automatically over all the dashboards every certain seconds. After activating this option, please try to put your web browser in full screen mode (each web browser has a different menu option or key combination for this purpose). In this option you will also be able to copy the link which will allow you to reproduce this presentation in other devices.
- **Add to favorites (only available in certain reports):** adds present report to the

Favorites reports. If you have the Common Favorites role you can create the favorite for all users, otherwise the favorite report will be only accessible to you. Additionally, if the server has the "Cache of favorites" option activated, favorite reports will be automatically generated and stored in order to be shown instantaneously to users.

- **Remote apps**: appears only when remote applications have been enabled. Provides access to the page to download the direct accesses to the remote applications, usually the console and the operation terminal, published by a server. Executing remote apps can be necessary when:
    1. There is no other way of accessing the Console.
    2. The terminal works very slowly.
    3. The terminal does not work in the device (tablet, mobile phone, etc.) but Microsoft Windows Server remote applications do.
- **Help us to improve**: sends an email directly to the heardquarters of edinn. This will help us to improve the system for you.
- **Show / Hide tooltips**: shows or hides the pop-up help / support messages at fields and columns of reports.
- **Send link**: prepares an email with a link to the report. The receiver of the email can access the same report by clicking on the link and introducing its own username and password to access reports. The receiver should have permission to access reports.
- **Export to CSV**: allows to export data (not graphics) to a .CSV format which can be read by MS Excel or by an OpenOffice calculation sheet.
- **Print**: allows to print the report.

> **Note**: When **printing**, keep in mind that the **colors** of the table cells, the elements of the graphs and the option boxes, will only be seen if you activate the "Background graphs" option in the printing options, found in the "more settings" or "advanced options" section depending on your browser.

- **Test API:** allows you to test edinn API calls without programming. Examples of use can be found in the API documentation.
- **Backups:** Allows to access the list of backups of your database if correctly configured. This option is only available for resources with super administrator role. Backups files will be compressed with the password indicated in the corresponding section of the console.
- **About**: shows the about section.
- **Help:** shows the help section.
- **Night mode**: switches to dark mode which is more suitable for low light

environments.

- **Alert settings:** allows you to change whe do you want to recive alerts by email.
- **Change password:** allows you to chage your password.
- **Logout:** close our session.
- **Show report:** show the selected report.

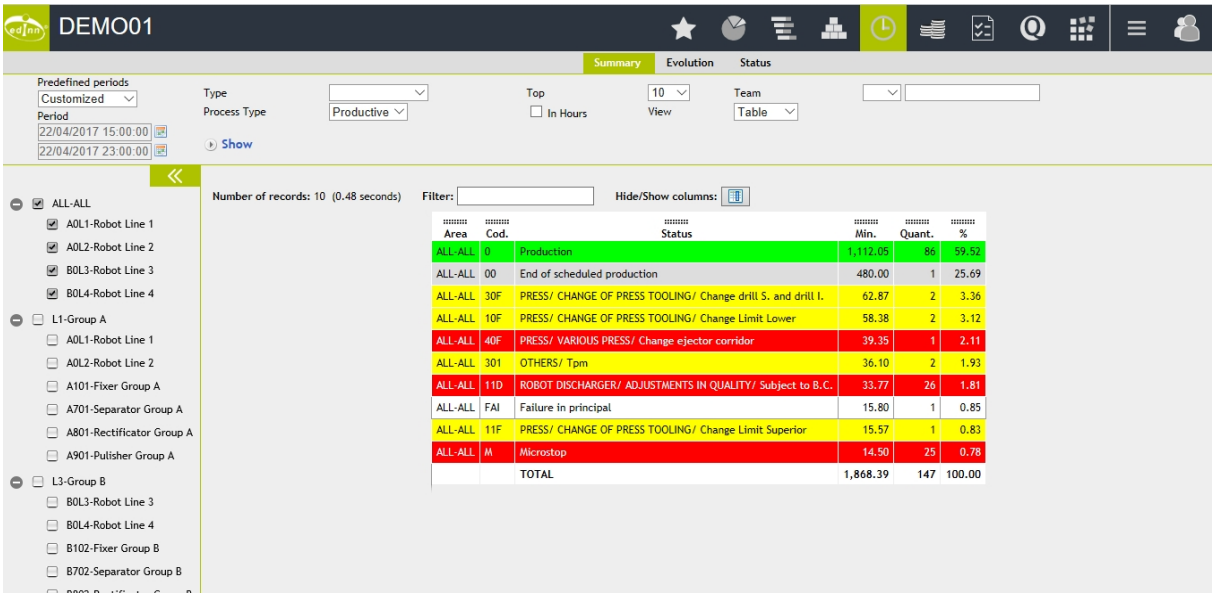
- Fields in each report:
  - **Predefined periods:** the report includes a series of predetermined periods, for faster access to information.
  - **Period:** to enter the dates of the specific period of time to analyze.
  - **Area:** selects the area or areas of processes to analyze.
  - **Process:** selects the process or processes to analyze.
  - **High precision**: higher precision, much more time to calculate. Most of the times the gain of precision is not necessary.
  - **In hours:** it displays information in hours rather than minutes.
  - **View**: table or 2D graphic.

- Areas and processes tree:

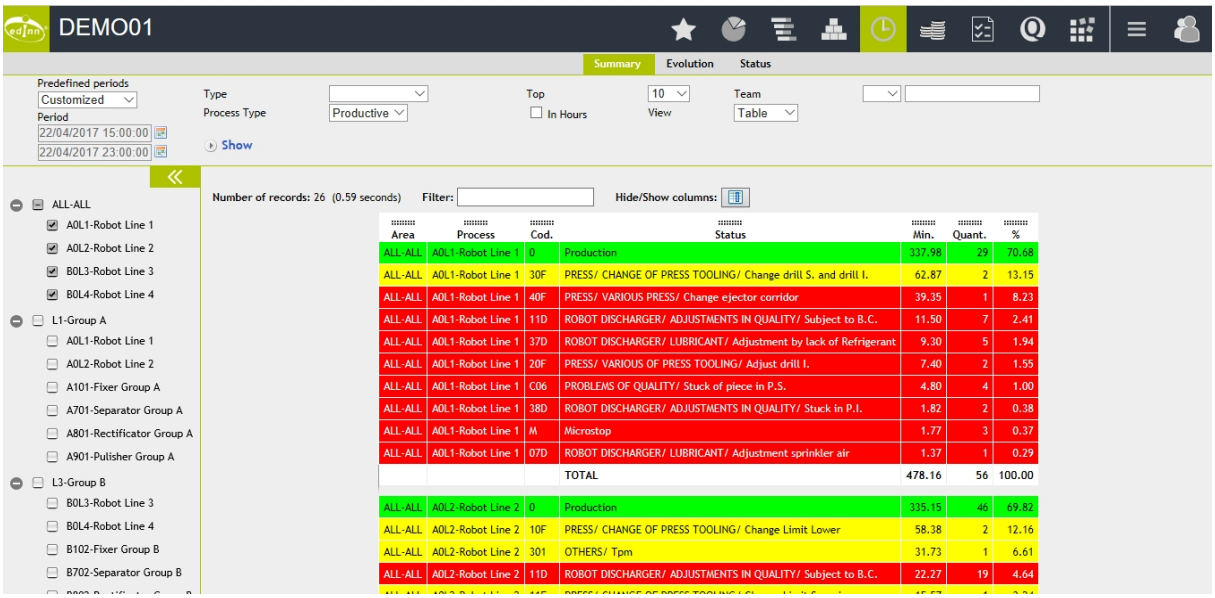
  The operation of the area and process selector is as follows: If an area is selected, any report that can be executed for areas will appear with the data grouped by area.

> **Note:** Please keep in mind that, in edinn, area ratios are not calculated as the sum of the components (quantities, times, etc.) of every subprocess, as additional mathematical calculations must be performed. The objective is to avoid that, for example, a process that works a little time and thus is not representative for the area but produces a lot more units that other processes of the same area could, when adding its quantities to the area, result in a high ratio when the ratio is actually low.

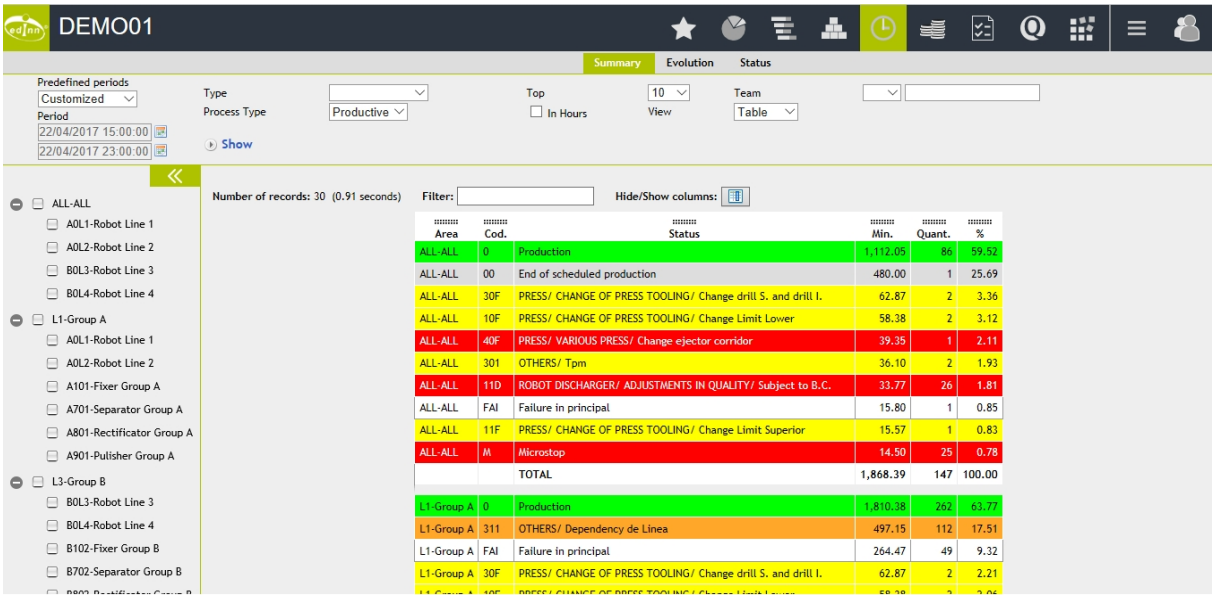
As shown in the image, the box for selecting the area and its child processes are marked.



If you click again on the box of the area, it is marked with the "minus" symbol (-), but the boxes of the processes will remain marked and the report will be shown differentiating the data by process.



If you click again on the box, this and the child processes will be completely unchecked, there will be no area or process selected and the report will show the information of all the areas with the data grouped by area or, in the reports where required the selection of some area or process, will show the corresponding error message.



- Most of the reports perform weighted calculations to provide ratios more adjusted to reality. For example, if results where generated in a period where there was no production time, the results may be shown reduced or in a contiguous period which had production time. This can cause, specially in short periods between shifts, a mismatch between the totals of weighted and not weighted reports.

To view gross quantities, with no weighting, use the following reports:

Results	Status	Consumption	Autocontrol	Quality	Registry
• Summary • Results	• Summary • Evolution • Status	• Panel • Justification • Consumption	• Follow up • Autocontrol • Configuration	• Data	• Events • Not recorded

Favorites

This is the section of the reports regarding recomnendations, favorites and dashboards.

Recommendations

Diagnosis

This report shows if the organization is using correctly the edinn platform.

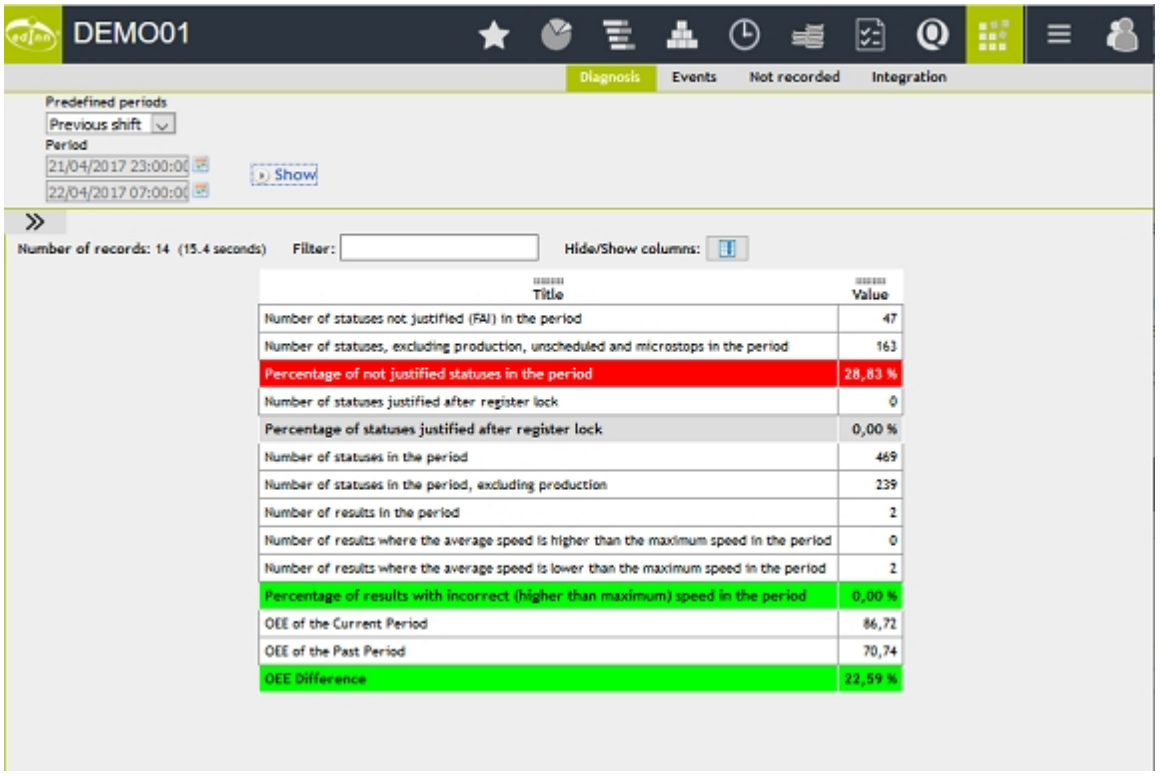
These are the first 3 factors to determine if an organization is using the edinn platform correctly:

1. **Users justify failures.**
2. **Users justify failures in the time they have for this.**
3. **Real cycle times are not faster than those configured in edinn.**

This report analyzes these 3 factors, and shows:

1. **Percentage of not justified statuses in the period**: indicates the percentage of the stops detected by edinn which have not been justified by electrical signals nor users. Red color means that should be improved.
2. **Percentage of statuses justified after the registers were locked**: indicates the percentage of the stops that were justified by a human supervisor after they were blocked to the user because of too much delay (in minutes, field Lock in the processes configuration) in the justification.
3. **Percentage of results with incorrect (higher than theoretical maximum) speed in the period**: indicates the percentage of results (also known as products, parts or references) which average cycle time was faster than the maximum theoretical configured in edinn. Red color means that it should be improved.
4. Additionally, this report shows the **evolution of the OEE** for the selected processes and areas and the chosed period.

An example of this report is shown below:



To improve these 3 factors, we recommend these 2 steps:

1. **Achieve that all results have a correct theoretical maximum cycle time.** Verify that your results are configured in edinn with a correct maximum theoretical cycle time (theoretical speed), and correct them if they are not correct, comparing with the actual speed. You can correct the theoretical maximum cycle time (Cycle Time and Cycle Units) here.

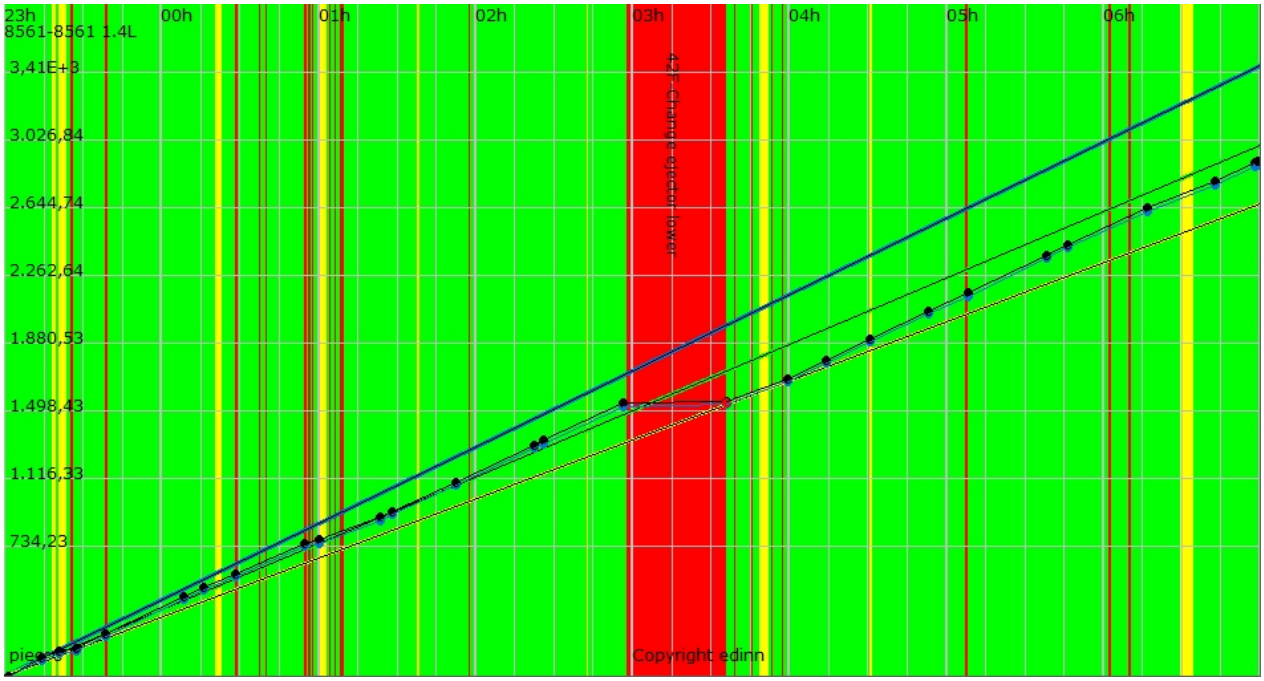
2. **Achieve that users justify all PENDING stops.** For this, your edinn software and operation terminals should be fully operative, and your users should follow these steps.

To compare the actual speed with the theoretical speed, you can use the Path graphic, which is shown in:

A) The terminal, in the main window.

B) The web reports, in the Path report.

In the path graphic, the black dots are real production and the blue dots are theoretical production at the maximum speed. Therefore, the black dots should never be above the blue dots. If black dots are shown above the blue dots, that means that the cycle units / time are not configured correctly, and actual speed of process is higher than that configured in edinn. Please see the example below, where black dots are slightly above blue dots, but the difference is so little that this is consider acceptable.



To obtain further help, you can contact support.

Relevant (beta)

This report shows which elements (shifts, results, teams, processes or resources [persons]) are relevant to increase or decrease the selected ratio. It is useful to detect which elements significantly influence to achieve the objectives of OEE, Availability, Speed and Quality. Once the report has been obtained, it is possible to clic on each element to obtain more information in the Follow up report.

Note: This report is in beta phase, which means that edinn is still receiving comments from users and analysing them to improve this report.

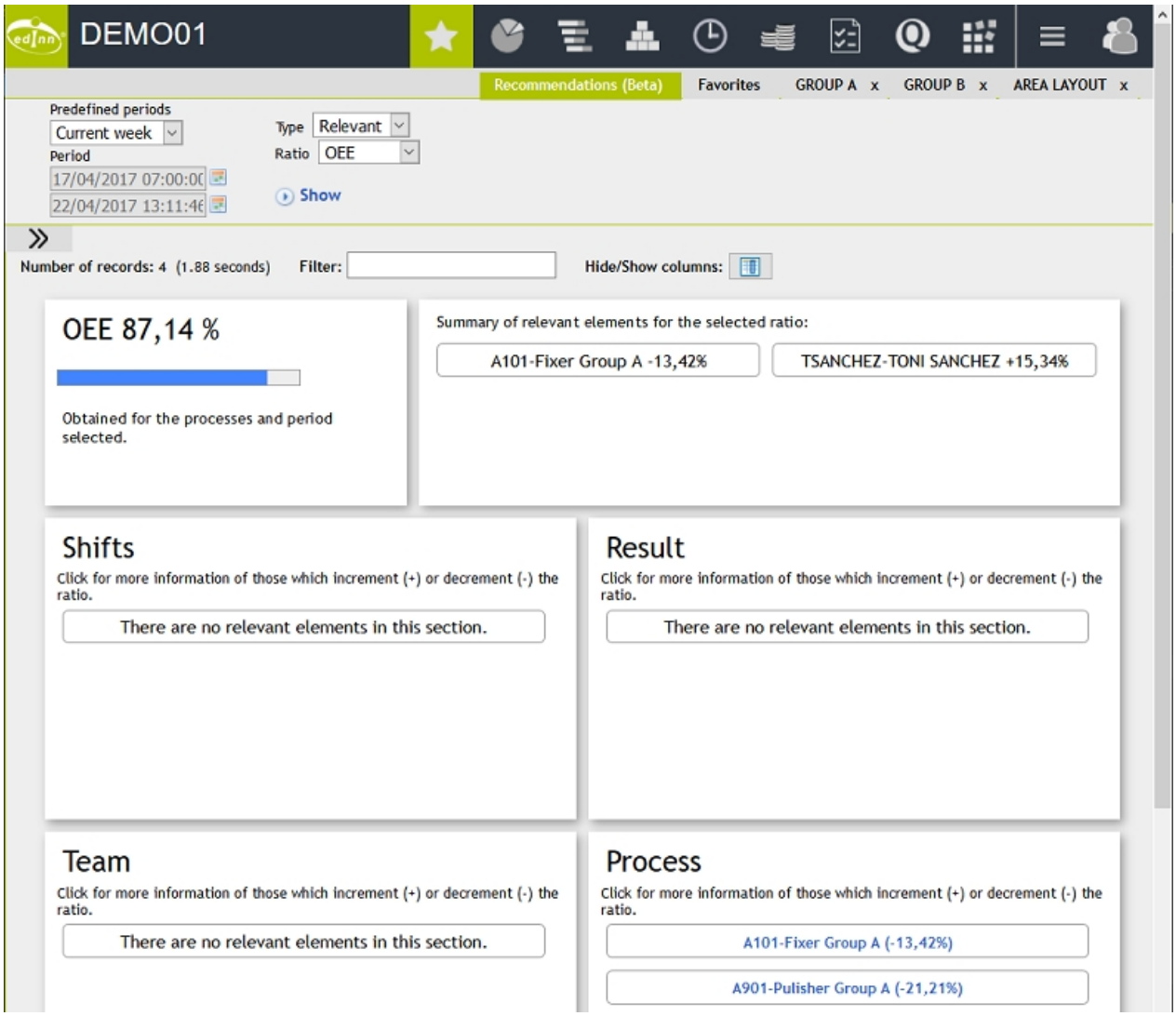
Note: This report requires the Big Data & Artificial Intelligence module and Big Data automatic calculations to be activated.

Take into account that:

- If the report does not show a certain element, it does not mean that the element did not contribute to increase or decrease the selected ratio, but that there is no statistical evidence that its influence is significant, and this could be because of not having enough data or enough variability inside them.
- Resources (persons) are not considered individually when they are working together. For example: if a resource (person) was working together with another resource (person) in the same process and period, they will be considered together as a new element named "OPERATOR1,OPERATOR2".

Additional details are as follow:

- **See:** "Common aspects"
- **Ratio:** select the ratio you wish to analyze.

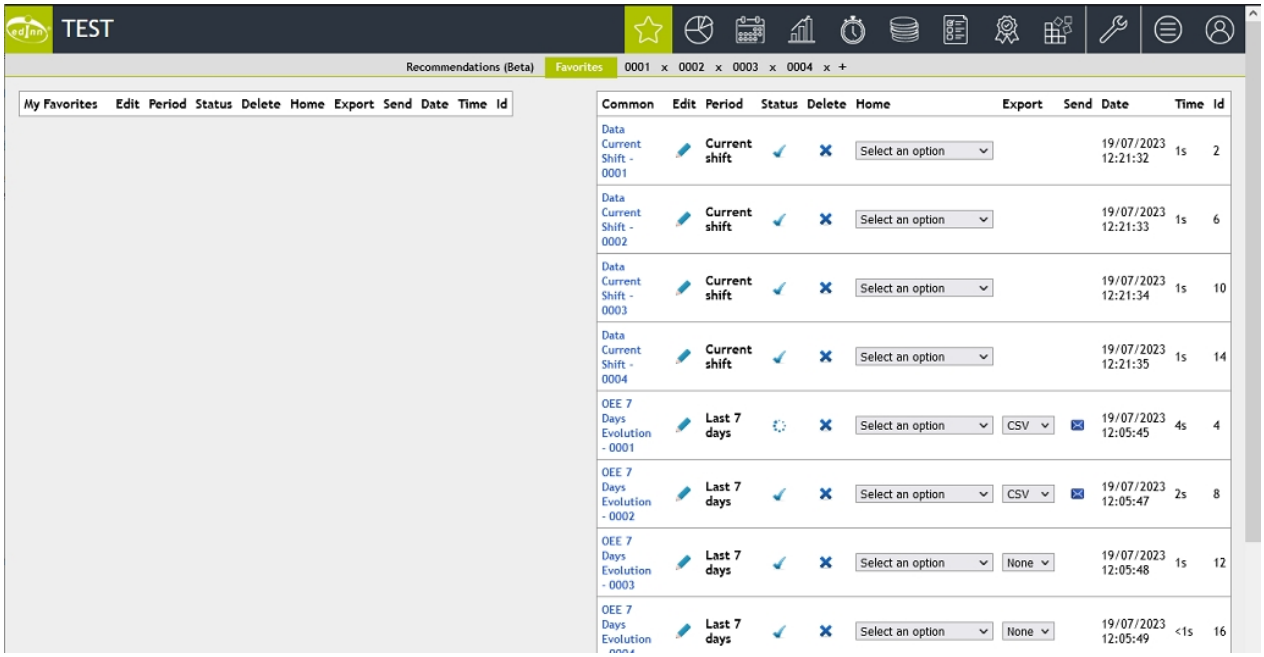


Favorites

The Favorites reports allows people to have their most frequently used reports in one screen, and those who do not know the reporting tool, to obtain the basic and most important reports in only 1 click, without needing any more information.

In addition, Favorite reports have the advantage that they are pre-calculated by the system in order to be shown faster, even if they imply the analysis of millions of records of data. For this, the option "Cache favorites" must be activated in the server.

Favorites shows the favorite reports of the user and the organization. This favorite reports are pre-calculated by the system.



Favorite reports can be added to dashboards by selecting in the "Home" column.

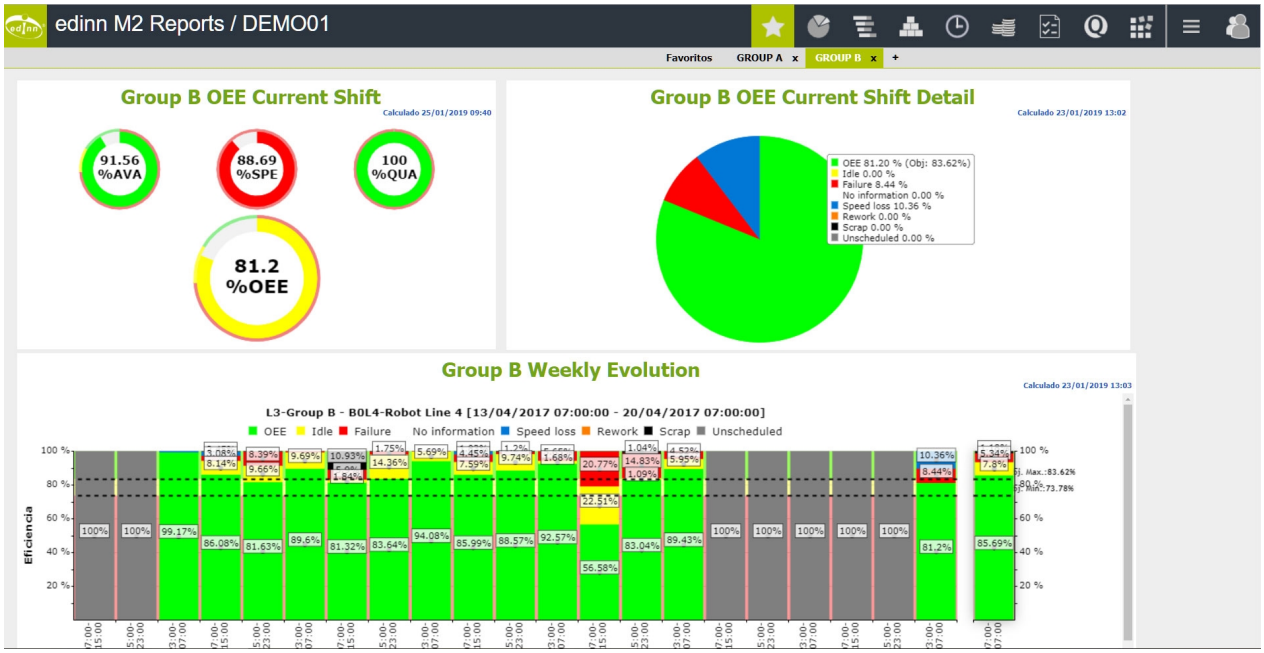
They can be automatically exported to CSV or PNG, by selecting the column "Export" list, when calculated by the system in a location that is configured from the server console in the reports tab. The name of the CSV or PNG file will match the name of the favorite followed by a generation date marker.

If the column "Send" is active, the exported file will be sent by email (email 1) to the user owner or, in the case of common favorites which do not belong to any user, to the email (email 1) of the user indicated in the section "Cache of favorites" in the server console in the reports tab.

Dashboards

Dashboards allow to configure customized dashboards using the common or user's favorite reports.

It also allows to assign, by clicking on the title of the tab, the dashboard to devices and processes and they will appear on the main screen of the Terminal and on the screensaver.



Analysis

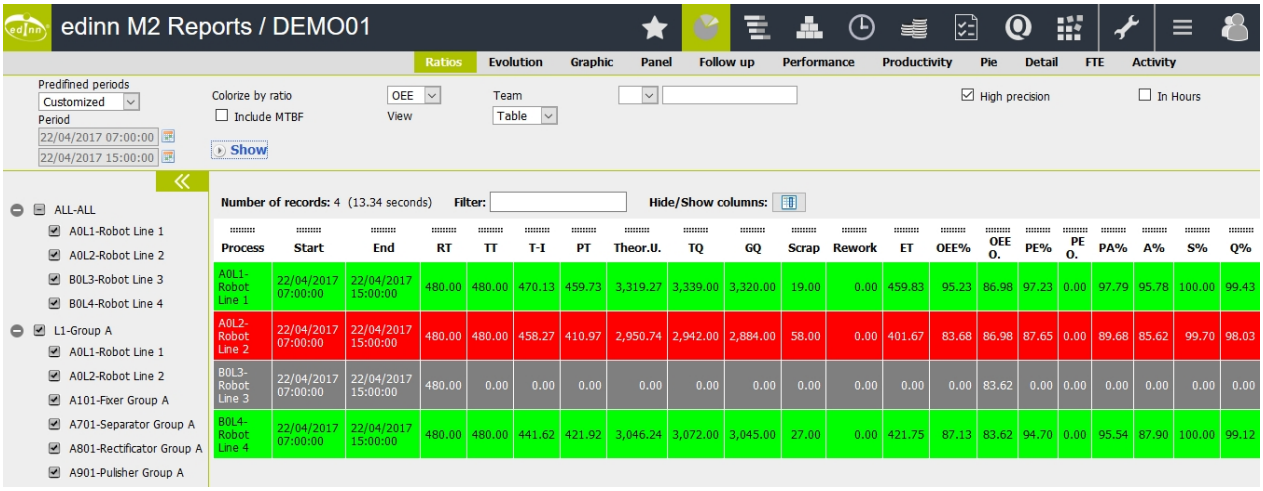
This is the section for the reports regarding analysis.

It is recommended to read the following documents production and energy and resources efficiency in this manual.

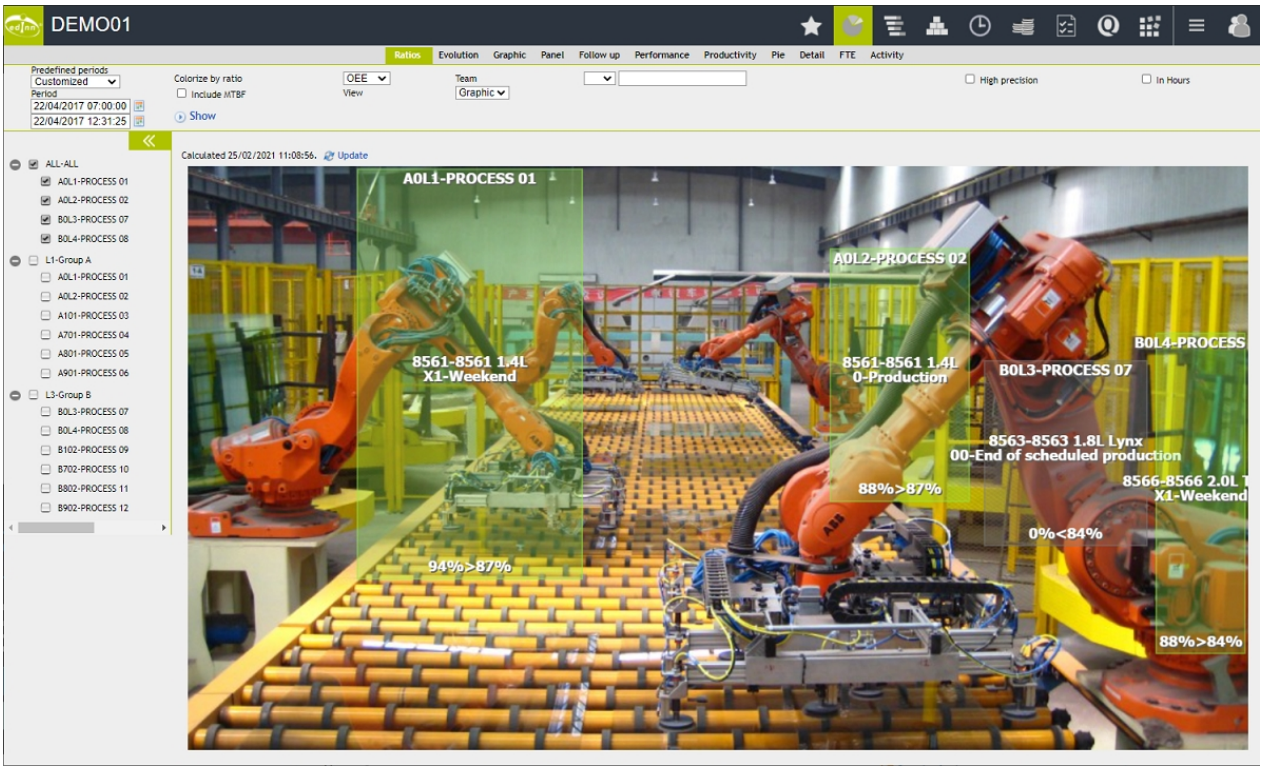
Ratios

This report shows the production efficiency ratios. It is one of the most useful reports regarding production efficiency. The reason is that it shows in colors how areas or processes are performing. The user can focus on analyzing only red areas and processes.

To find the reason why an area or process has losses (is red or yellow) simply click on the record.



On this report, graphical lay-outs can be shown as in the image below, by selecting view graphic of an area which has a graphic configured:



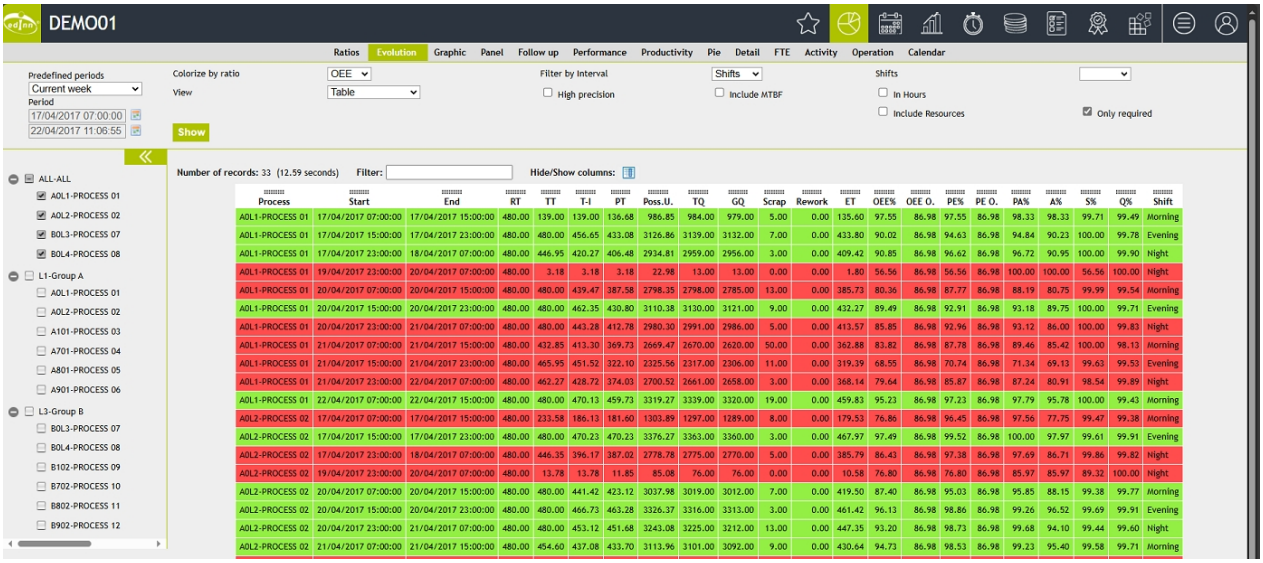
Selectors are:

- **Please see:** "Common aspects"
- **Colorize by ratio:** perform the division of colors based on the following ratios: OEE, OCE, OPCE.
- **Include MTBF**: MTBF information will be included.
- **Team**: if you want to see only information relative to the times where that specific team was working.
- **View:** allow to view the information in a table or graphic format.

Evolution

This report shows the evolution in time of the production efficiency ratios. It is, together with the report Ratios, one of the most useful reports regarding production efficiency. The reason is that it shows in colors how areas or processes are performing. The user can focus on analyzing only red areas and processes.

In addition to the functionality of the report ratios, this reports adds that it shows the evolution in time. That is useful to see if the organization is reducing certain losses or the results of certain actions or investments inside the organization.



Process	Start	End	RT	TT	T-I	PT	Poss.U.	TQ	GQ	Scrap	Rework	ET	OEE%	OEE O.	PE%	PE O.	PA%	A%	S%	Q%	Shift
A0L1-PROCESS 01	17/04/2017 07:00:00	17/04/2017 15:00:00	480.00	139.00	139.00	136.68	986.85	984.00	979.00	5.00	0.00	135.60	97.55	86.98	97.55	86.98	98.33	98.33	99.71	99.49	Morning
A0L1-PROCESS 01	17/04/2017 15:00:00	17/04/2017 23:00:00	480.00	480.00	456.65	433.08	3126.86	3139.00	3132.00	7.00	0.00	433.80	90.02	86.98	94.63	86.98	94.84	90.23	100.00	99.78	Evening
A0L1-PROCESS 01	17/04/2017 23:00:00	18/04/2017 07:00:00	480.00	446.95	420.27	406.48	2934.81	2959.00	2956.00	3.00	0.00	409.42	90.85	86.98	96.62	86.98	96.72	90.95	100.00	99.90	Night
A0L1-PROCESS 01	19/04/2017 23:00:00	20/04/2017 07:00:00	480.00	3.18	3.18	3.18	22.98	13.00	13.00	0.00	0.00	1.80	56.56	86.98	56.56	86.98	100.00	100.00	56.56	100.00	Night
A0L1-PROCESS 01	20/04/2017 07:00:00	20/04/2017 15:00:00	480.00	480.00	439.47	387.58	2798.35	2798.00	2785.00	13.00	0.00	385.73	80.36	86.98	87.77	86.98	88.19	80.75	99.99	99.54	Morning
A0L1-PROCESS 01	20/04/2017 15:00:00	20/04/2017 23:00:00	480.00	480.00	462.35	430.80	3110.38	3130.00	3121.00	9.00	0.00	432.27	89.49	86.98	92.91	86.98	93.18	89.75	100.00	99.71	Evening
A0L1-PROCESS 01	20/04/2017 23:00:00	21/04/2017 07:00:00	480.00	480.00	443.28	412.78	2980.30	2991.00	2986.00	5.00	0.00	413.57	85.85	86.98	92.96	86.98	93.12	86.00	100.00	99.83	Night
A0L1-PROCESS 01	21/04/2017 07:00:00	21/04/2017 15:00:00	480.00	432.85	413.30	369.73	2669.47	2670.00	2620.00	50.00	0.00	362.88	83.82	86.98	87.78	86.98	89.46	85.42	100.00	98.13	Morning
A0L1-PROCESS 01	21/04/2017 15:00:00	21/04/2017 23:00:00	480.00	465.95	451.52	322.10	2325.56	2317.00	2306.00	11.00	0.00	319.39	68.55	86.98	70.74	86.98	71.34	69.13	99.63	99.53	Evening
A0L1-PROCESS 01	21/04/2017 23:00:00	22/04/2017 07:00:00	480.00	462.27	428.72	374.03	2700.52	2661.00	2658.00	3.00	0.00	368.14	79.64	86.98	85.87	86.98	87.24	80.91	98.54	99.89	Night
A0L1-PROCESS 01	22/04/2017 07:00:00	22/04/2017 15:00:00	480.00	480.00	470.13	459.73	3319.27	3339.00	3320.00	19.00	0.00	459.83	95.23	86.98	97.23	86.98	97.79	95.78	100.00	99.43	Morning
A0L2-PROCESS 02	17/04/2017 07:00:00	17/04/2017 15:00:00	480.00	233.58	186.13	181.60	1303.89	1297.00	1289.00	8.00	0.00	179.53	76.86	86.98	96.45	86.98	97.56	77.75	99.47	99.38	Morning
A0L2-PROCESS 02	17/04/2017 15:00:00	17/04/2017 23:00:00	480.00	480.00	470.23	470.23	3376.27	3363.00	3360.00	3.00	0.00	467.97	97.49	86.98	99.52	86.98	100.00	97.97	99.61	99.91	Evening
A0L2-PROCESS 02	17/04/2017 23:00:00	18/04/2017 07:00:00	480.00	446.35	396.17	387.02	2778.78	2775.00	2770.00	5.00	0.00	385.79	86.43	86.98	97.38	86.98	97.69	86.71	99.86	99.82	Night
A0L2-PROCESS 02	19/04/2017 23:00:00	20/04/2017 07:00:00	480.00	13.78	13.78	11.85	85.08	76.00	76.00	0.00	0.00	10.58	76.80	86.98	76.80	86.98	85.97	85.97	89.32	100.00	Night
A0L2-PROCESS 02	20/04/2017 07:00:00	20/04/2017 15:00:00	480.00	480.00	441.42	423.12	3037.98	3019.00	3012.00	7.00	0.00	419.50	87.40	86.98	95.03	86.98	95.85	88.15	99.38	99.77	Morning
A0L2-PROCESS 02	20/04/2017 15:00:00	20/04/2017 23:00:00	480.00	480.00	466.73	463.28	3326.37	3316.00	3313.00	3.00	0.00	461.42	96.13	86.98	98.86	86.98	99.26	96.52	99.69	99.91	Evening
A0L2-PROCESS 02	20/04/2017 23:00:00	21/04/2017 07:00:00	480.00	480.00	453.12	451.68	3243.08	3225.00	3212.00	13.00	0.00	447.35	93.20	86.98	98.73	86.98	99.68	94.10	99.44	99.60	Night
A0L2-PROCESS 02	21/04/2017 07:00:00	21/04/2017 15:00:00	480.00	454.60	437.08	433.70	3113.96	3101.00	3092.00	9.00	0.00	430.64	94.73	86.98	98.53	86.98	99.23	95.40	99.58	99.71	Morning

Selectors are:

- **Please see:** "Common aspects"
- **Colorize by ratio:** perform the division of colors based on the following ratios: OEE, OCE, OPCE.
- **Interval:** the periods that will divide the records.
- **Include MTBF**: MTBF information will be included.
- **Shifts**: the shifts to show. This is useful to analyze the evolution of a certain shift.
- **Include Resources:** adds a column with the identifiers of the resources which did IN in the period.
- **Only required:** It only shows records which have demandable time, which is useful to hide periods not scheduled for production.

Graphic

This report shows the efficiency in a bars graphic, indicating the losses with a color code. The user can then click on the losses to see the detail.

This report is calculated both for processes and areas (groups of processes).

Selectors are:

- **Please see:** "Common aspects"
- **Type**: the type of ratio: OEE, PE.
- **Fields**: if you want fo filter production performed with a certain field with a certain value.
- **Contains**: what must be contained in the field selected in the previous selector.
- **Interval**: the periods that will divide the bars.
- **Shifts**: if you want to see just a certain shift.

Panel

This report shows different panels of aggregated information.

Selectors are:

- **Please see:** "Common aspects"
- **Type**: see below.
- **Team:** To see the information only regarding the time when a specific team

worked.

- **Device**: allows to select the device from which its customized data will be taken. This selector is only shown for certain types of panels.
- **Ratio Type:** allows to select the ratio for the bars graphic, from OEE, PE or both ratios.
- **Hide Header (Data)**: allows you to view the data table without the header.
- **View (Data)**: view the data table as a bar graph. Data with different units cannot be displayed together.

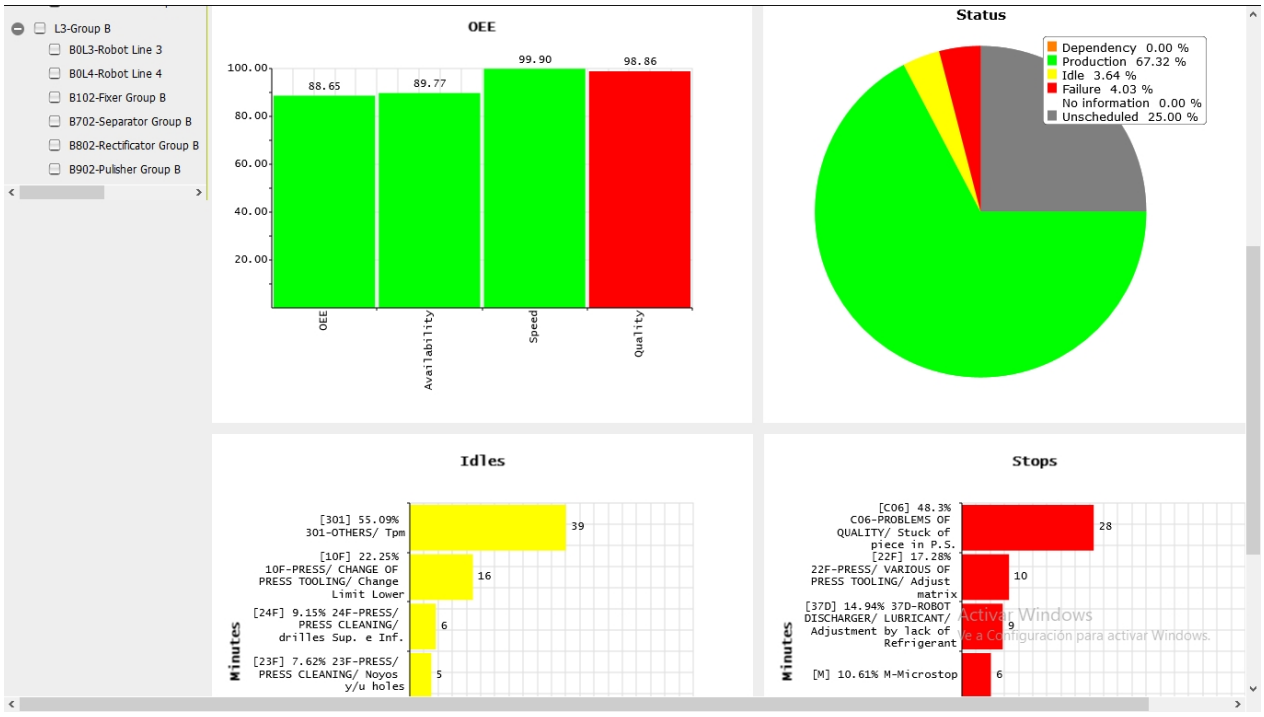
There are these types of panels:

- Complete
- Evolution
- Data

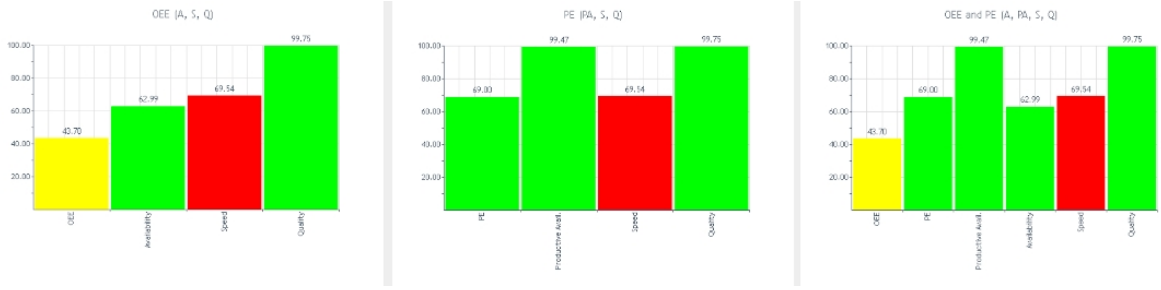
Let us see the different types of panels.

Complete

This report shows the main production data and the following graphics, in one page:



- OEE, PE or both ratios and their components.



- The losses pie.
- The paretto of the idle time usages.
- The paretto of the stops.

Evolution

This report shows the evolution per day and shift of production and scrap quantities.

		22/04/2017		21/04/2017		20/04/2017		17/04/2017		16/04/2017	
Process	Shift	Quantity	Scrap	Quantity	Scrap	Quantity	Scrap	Quantity	Scrap	Quantity	Scrap
A0L1-Press Line 1	Morning	3.339	19	2.670	50	2.798	13	984	5	1.501	8
A0L1-Press Line 1	Evening	2.457	5	2.317	11	3.130	9	3.139	7	2.461	24
A0L1-Press Line 1	Night	3.318	8	2.661	3	2.991	5	2.959	3	3.224	5
	TOTALS	9.114	32	7.648	64	8.919	27	7.082	15	7.186	37

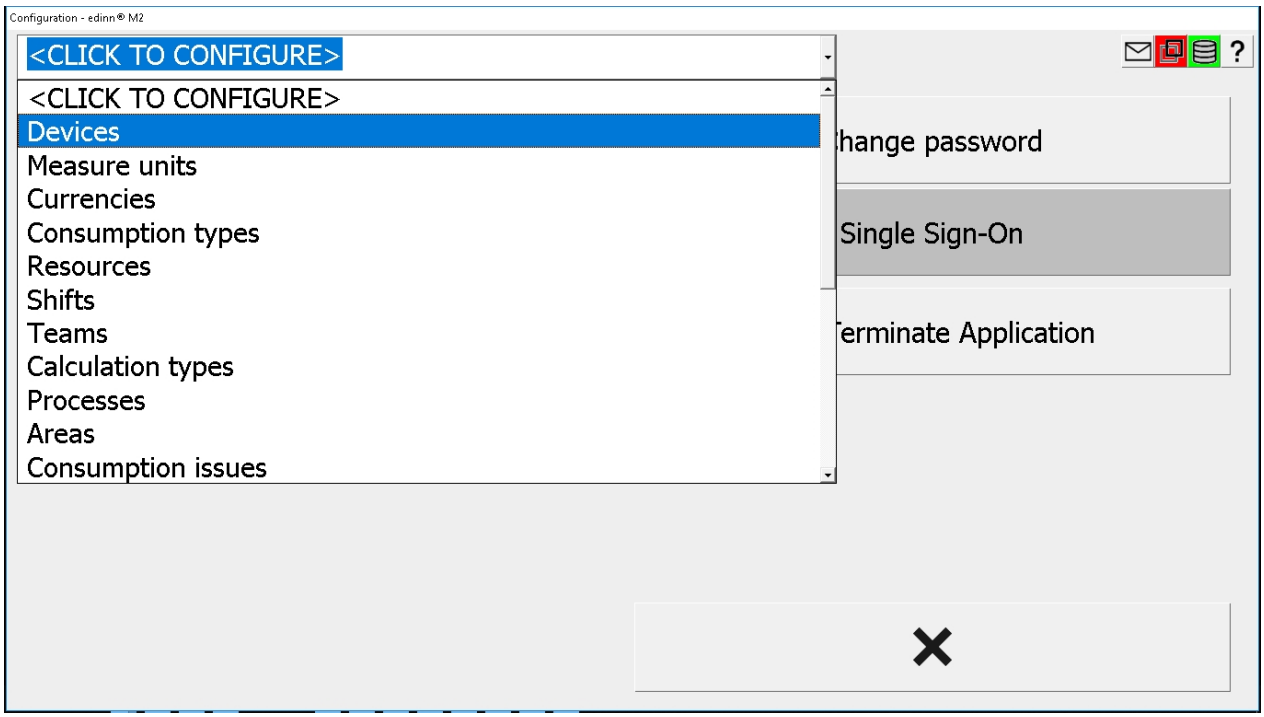
Data

This report shows different data which can be selected from a list.

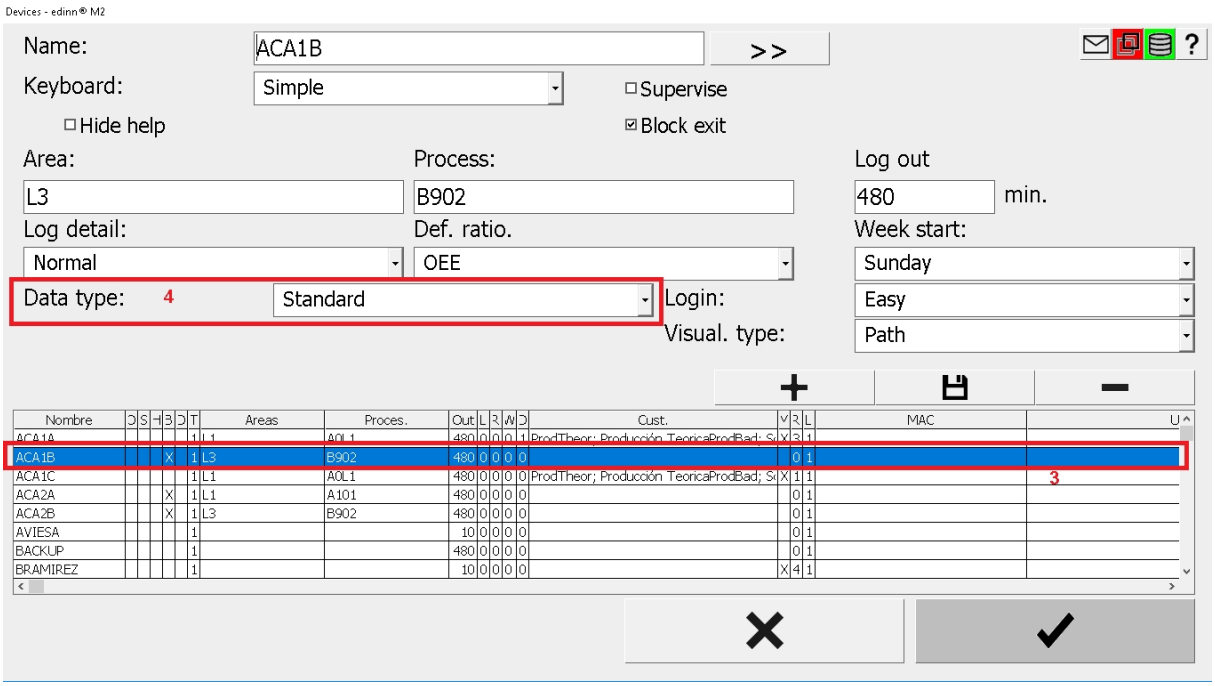
Customization of data

The data shown in the report Analysis, Panel, Data can be configured in the Configure Devices window. To do so:

1. Click on the button from the Main Window of the **Terminal**.
2. Select **Devices** in the **CLICK TO CONFIGURE** dropdown.



3. In the following window, choose the device whose data field you want to configure.



4. Modify the field **Data Type**. This field shows 2 options:

- **Standard**: it is the default configuration:

  ProdGood;Good (units)
  ProdTotal;Total (units)
  TimeProdMin;Productive (minutes)
  TimeNotProdMin;Not productive (minutes)

SpeedT;Nominal (un./min.)
Speed0;Average (un./min.)

- **Particular**: allows that each device (terminal) shows specific information.

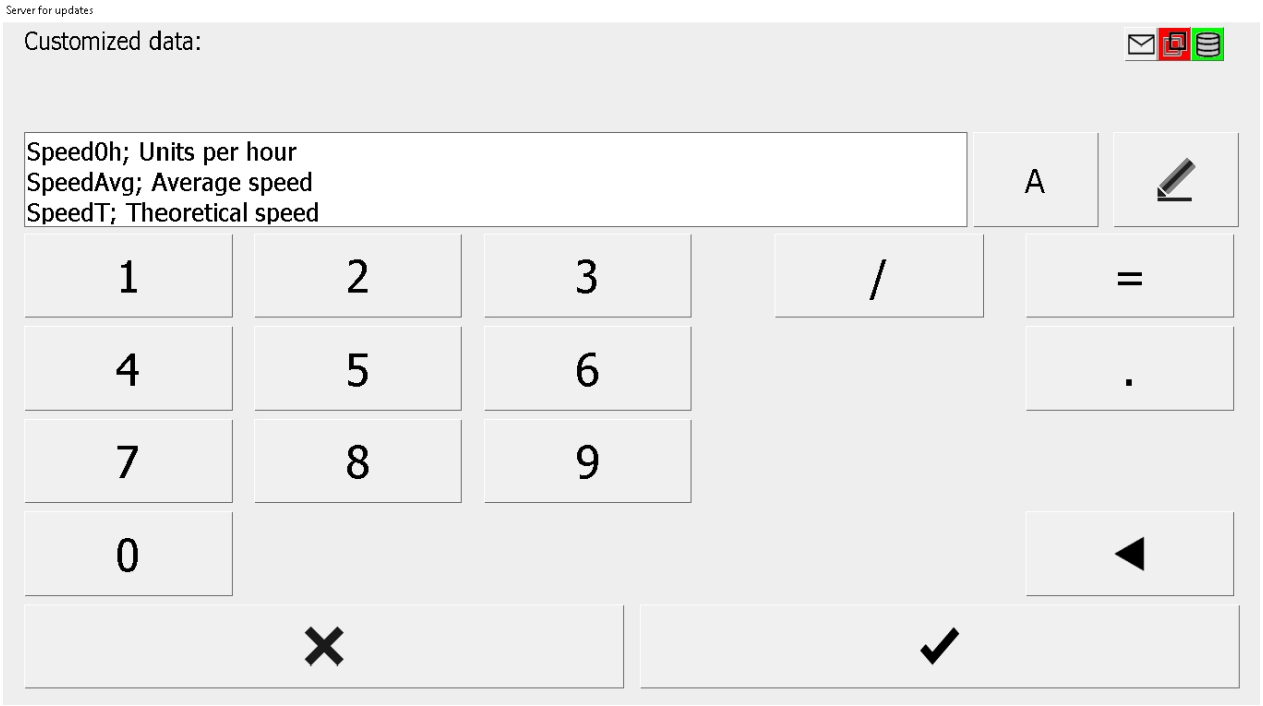
To configure the "Particular" option, use this syntax:

RESERVED TAG1; Free Text1
RESERVED TAG1; Free Text2
...
RESERVED TAG1; Free TextN

White lines can be introduced as separators. Type CTRL+ENTER to enter a new line.

When [0..2] is indicated, it means that the numbers 0, 1 and 2 can be appened to the tag. If omitted, it will consider all the period. If indicated, 0 means the period of the current result, 1 means the period of the previous result and 2 means the previous to the previous.

For example ProdTotal is the total production of the period, ProdTotal0 is the total production of the current result, ProdTotal1 is the total production of the previous result and ProdTotal2 is the total production of the previous of the previous result.



Available RESERVED TAGs, in **alphabetical order**, are:

- **CommentWO:** shows the comment of the current order.
- **CPK**: shows actual CPK for the selected measure.
- **CycleQ**: quantity produced that was read in the last monitoring cycle in units.
- **CycleT**: number of seconds needed to produce the CycleQ that were read in the last monitoring cycle.

- **CycleQTheor**: theoretical quantity to be produced in a cycle for a given moment, taking into account the active result and the work order.
- **CycleTTheor**: theoretical number of seconds of a cycle for a given moment, taking into account the active result and the work order.
- **CycleTEquiv**: shows the actual average cycle time, in seconds, to produce the number of cycle units configured in the nominal production capacity.
- **DueDateWO:** shows the due date of the current order.
- **EstimationWO:** shows the time that edinn has calculated that it will take to finalize the current order.
- **MonitorItemValue[id]:** shows the current value of a monitor signal.

NOTE: the MonitorItemValue[id] tag must indicate the identificator value of the monitor signal. Example: MonitorItemValue[15].

- **MTBF**: shows MTBF for the selected period in minutes.
- **MTBFLastUpdate**: shows date and time of last update of MTBF information
- **MTBFQ**: shows number of failures in the selected period.
- **MTTR**: shows MTTR for the selected period in minutes.
- **OEE**: OEE ratio of the current shift in percentage.
- **OEEA**: availability ratio of the OEE of the current shift in percentage.
- **OEEE**: speed ratio of the OEE of the current shift in percentage.
- **OEEQ**: quality ratio of the OEE of the current shift in percentage.
- **OrderWO:** shows the code current order.
- **PriorityWO:** shows the priority of the current order.
- **Id_Prod:** reference of the current result.
- **Prod[0..2]**: reference of result.
- **ProdBad[0..2]**: total bad production or scrap in units.
- **ProdBadKgs[0..2]**: total bad production or scrap in kilograms.
- **ProdBadWO:** shows the scrap quantity produce in the current order in units.
- **ProdGood[0..2]**:total good production in units.
- **ProdGoodKgs[0..2]**:total good production in kilograms.

- **ProdGoodWO:** shows the good quantity produce in the current order in units.
- **ProdReworkWO:** shows the rework quantity produce in the current order in units.
- **ProdTheor[0..2]**: total production in units that could have been produced.
- **ProdTheorKgs[0..2]**: total production in kilograms that could have been produced.
- **ProdTheorN**: total production in units that could have been produced, but calculated by multiplying production time by the current nominal speed of the proces.
- **ProdTheorTotalWO:** shows the theoretical production of the current order in units.
- **ProdTotal[0..2]**: total production in units.
- **ProdTotalKgs[0..2]**: total production in kilograms.
- **ProdTotalWO:** shows the production of the current order in units.
- **QaActivity**: is the activity of the process considering only the production of Qa variable. It is calculated as follows: (Total Production (Qa) - Scrap Production (Qa) - Rework Production (Qa)) / (Total time of Activity in hours). Where the total time of activity is the total time of all the statuses of the period, but multiplying their seconds by their activity percentage.
- **QbActivity**: is the activity of the process considering only the production of Qb variable. It is calculated as follows: (Total Production (Qb) - Scrap Production (Qb) - Rework Production (Qb)) / (Total time of Activity in hours). Where the total time of activity is the total time of all the statuses of the period, but multiplying their seconds by their activity percentage.
- **QcActivity**: is the activity of the process considering only the production of Qc variable. It is calculated as follows: (Total Production (Qc) - Scrap Production (Qc) - Rework Production (Qc)) / (Total time of Activity in hours). Where the total time of activity is the total time of all the statuses of the period, but multiplying their seconds by their activity percentage.
- **QtActivity**: is the activity of the process considering only the production of Qt variable. It is calculated as follows: (Total Production (Qt) - Scrap Production (Qt) - Rework Production (Qt)) / (Total time of Activity in hours). Where the total time of activity is the total time of all the statuses of the period, but multiplying their seconds by their activity percentage.
- **QaQbActivity[factor]**: is the activity of the process considering the value of the product of Qa and Qb variables. It is calculated as follows: (Total Production (Qa*Qb) - Scrap Production (Qa*Qb) - Rework Production (Qa*Qb)) / factor / (Total time of Activity in hours). Where “factor” is a user added value (equal or

greater than 1) and “total time of activity” is the total time of all the statuses of the period, but multiplying their seconds by their activity percentage.

- **QaQcActivity[factor]**: is the activity of the process considering the value of the product of Qa and Qc variables. It is calculated as follows: (Total Production (Qa*Qc) - Scrap Production (Qa*Qc) - Rework Production (Qa*Qc)) / factor / (Total time of Activity in hours). Where “factor” is a user added value (equal or greater than 1) and “total time of activity” is the total time of all the statuses of the period, but multiplying their seconds by their activity percentage.
- **QbQcActivity[factor]**: is the activity of the process considering the value of the product of Qb and Qc variables. It is calculated as follows: (Total Production (Qb*Qc) - Scrap Production (Qb*Qc) - Rework Production (Qb*Qc)) / factor / (Total time of Activity in hours). Where “factor” is a user added value (equal or greater than 1) and “total time of activity” is the total time of all the statuses of the period, but multiplying their seconds by their activity percentage.

NOTE: the system calculates production based on, potentially, up to 3 variables: Qa, Qb and Qc; combined by the formula of the calculation type of the process.

- **Speed0**: average current production speed in units / minute.
  - It is set to 0 if the machine stops.
  - Calculates the average speed by taking the last 2 results records and dividing the total amount produced by the elapsed time.
- **Speed0h**: average current production speed in units / hour. Same behaviour than Speed0.
- **SpeedAvgR:** average current speed in units / minute.
- **SpeedAvgRh:** average current speed in units / hour.

NOTE: SpeedAvgR, SpeedAvgRh tags are calculate from the last 10 records of the counter.

- **SpeedClass[id]:** speed of a interval time, taking into account only the states of the specified class, in units / minute.
- **SpeedClassh[id]:** speed of a interval time, taking into account only the states of the specified class, in units / hour.
- **SpeedWOClass[id]**: average speed of the current working order taking into account only the statuses of the specified class, in units / minute.
- **SpeedWOClassh[id]**: average speed of the current working order taking into account only the statuses of the specified class, in units / hour.

NOTE: identification values for each class are:
Productive = 0

Setup = 1
Teardown = 2
Failure = 3
Result change = 4
Tool change = 5
Corrective Maintenance = 6
Ordering and Cleaning = 7
Quality test = 8
Internal dependency = 9
External dependency = 10
No workload = 11
Audit = 12
Training = 13
Vacation = 14
Weekend = 15
Holiday = 16
Predictive Maintenance = 17
Preventive Maintenance = 18
Any class = 255

NOTE: SpeedClass[id], SpeedClassh[id], SpeedWOClass[id], SpeedWOClassh[id] tags must indicate the identificator value of the class. Example: SpeedWOClass[4].

- **SpeedR:** current speed in units / minute.
- **SpeedRh**: current speed in units / hour.
- **SpeedT**: theoretical speed (Nominal Capacity) or maximum speed in units / minute.
- **SpeedTh**: theoretical speed (Nominal Capacity) or maximum speed in units / hour.
- **SpeedGTh:** the theoretical speed (Nominal Capacity) or maximum speed in units / minute for the green objective.
- **SpeedYTh:** the theoretical speed (Nominal Capacity) or maximum speed in units / minute for the yellow objective.
- **SpeedT*S%**: shows average speed, multiplying theoretical speed (Nominal Capacity) by the speed ratio of the OEE (%), in units / minute.
- **SpeedT*S%h**: shows average speed, multiplying theoretical speed (Nominal Capacity) by the speed ratio of the OEE (%), in units / hour.
- **StartedWO:** shows the start time of the current order.
- **TimeMin**: total interval time (minutes).
- **TimeNotProdHr**: non productive time (hours).
- **TimeNotProdMin**: non productive time (minutes).

- **TimeNowMin**: total time from the start of the interval until current moment (minutes).
- **TimeProdHr**: productive time (hours).
- **TimeProdMin**: productive time (minutes).
- **TimeRelativeToShift**: percentage of time elapsed with respect to the time of the current shift.
- **TimeProdRelativeToShift**: percentage of time in production elapsed with respect to the time of the current shift.
- **TimeResMin[0..2]**: productive time in minutes.
- **TimeSetupWO:** shows the setup time of the current order in seconds.
- **TimeStatusWO[id]:** shows the time spent in one state in the current order in seconds.

WARNING: You need to have a Scheduling Module license to obtain the data with termination in "WO".

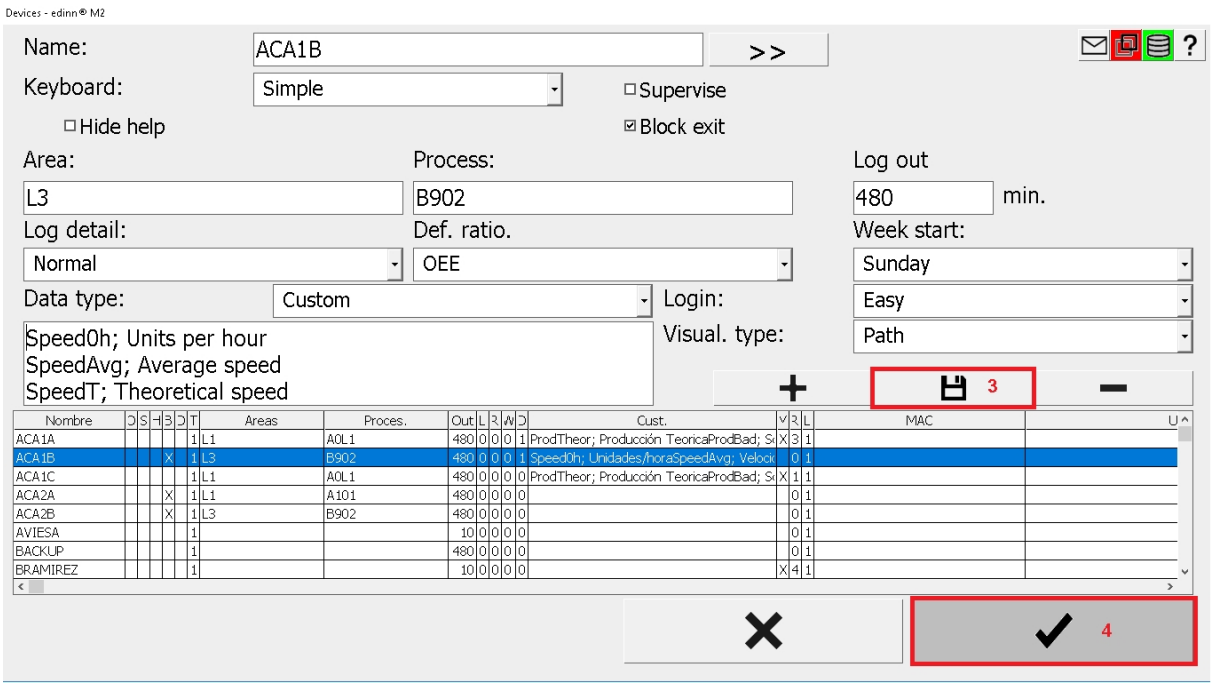
- **TimeStatus[id]:** shows the time spent in one state in seconds.

NOTE: TimeStatusWO[id], TimeStatus[id] tags must indicate the status code. Example: TimeStatus[FAI].

- **TimeTeardownWO:** shows the Teardown time of the current order in seconds.

If you do not find the label you are looking for, please contact edinn's technical support.

5. Once the changes have been done, save the changes by clicking 💾 , then click ✔ to confirm them.



6. Now, the data panel report should show the data specified by the new configuration.

Configuration of Data Screen in main window

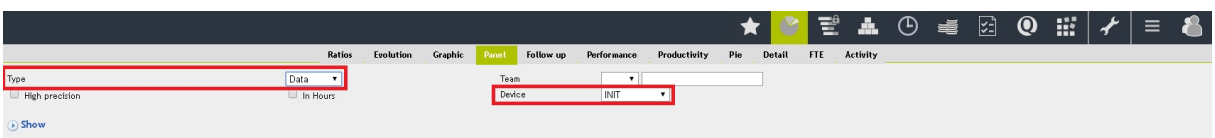
From edinn 2019-11 version build 330, the process data screen is not configured by default in the main screen of the edinn application, so if you want to have this screen accessible you will have to configure it manually. This procedure details the steps to configure screens with the data report for specific processes.

Step 1: Customization of data

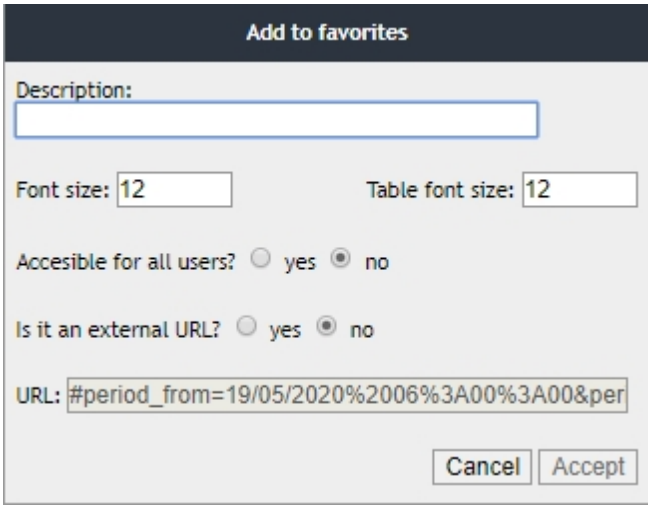
To configure the fields that we want the data report to show, we must follow the indications of the following link: Customization of data. You can choose to keep the standard configuration or to choose the data you want to see manually.

Step 2: In the reporting tool, add the Data Report to Favorites

1. From the reporting tool, in the **Analysis -> Panel** tab, configure a new **Data Report** (see Analysis -> Panel -> Data). Remember to Select **Type: Data**, as well as to select the **Device** on which we have configured the Customization of data that we want (see Step 1).



2. Add this report to Favorites (see Favorites). The following window will be displayed:

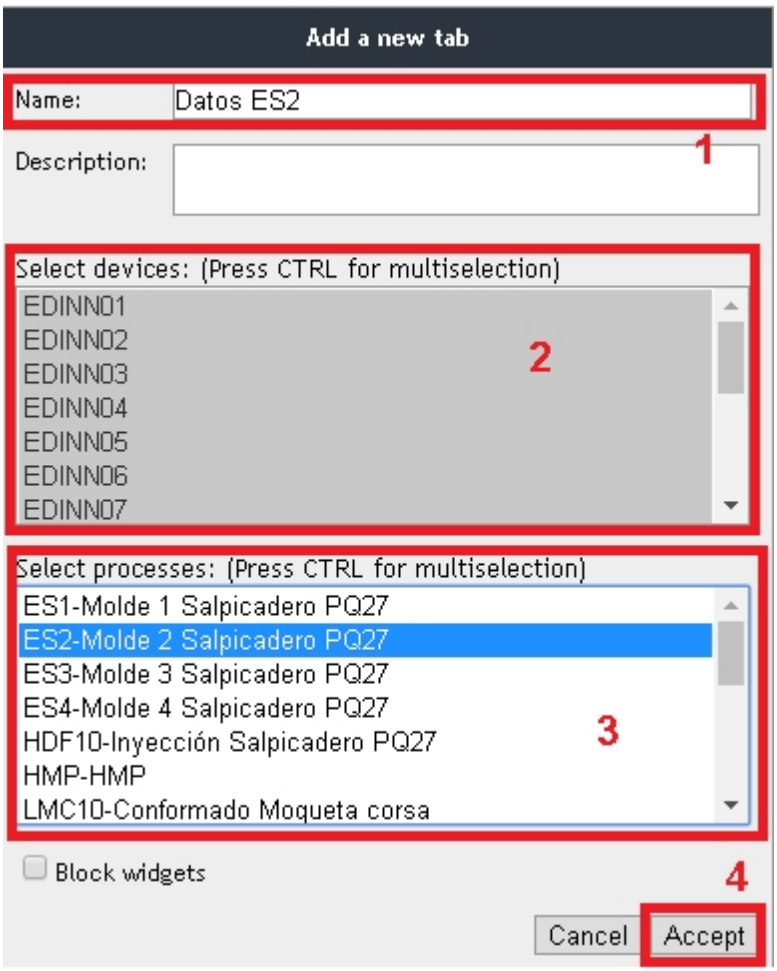


Enter the name of the report in the **Description** field. The **Font size** field indicates the font size of the title of the widget. The **Table font size** field indicates the size of the font inside the tables of data. The **Accessible for all users?** field indicates whether the report will be accessible to all users or only to those who created it. **Is it an external URL?** field indicates if the URL of the report is external or not; in this case it is not since we have created it with a widget from the edinn reports system.

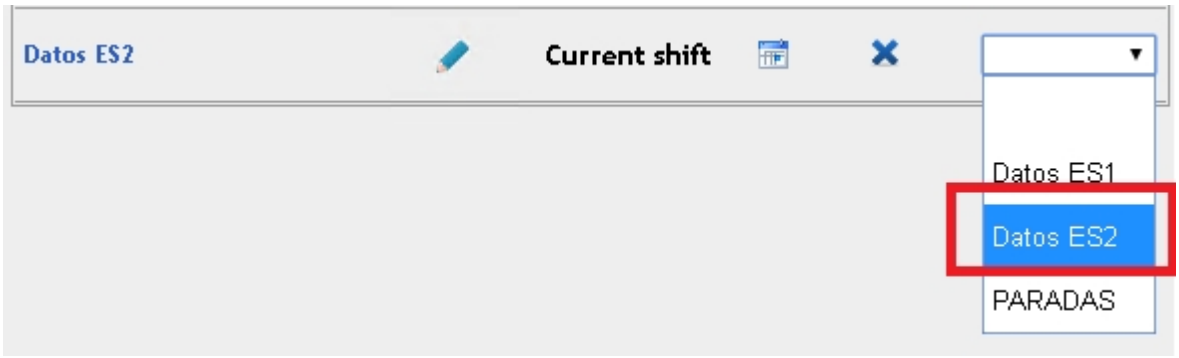
Step 3: In the reporting tool, create a new Dashboard including the created report.

1. Create a new Dashboard (see Favorites): To do this, go to the Favorites screen of the report tool and click on the "+" tab.

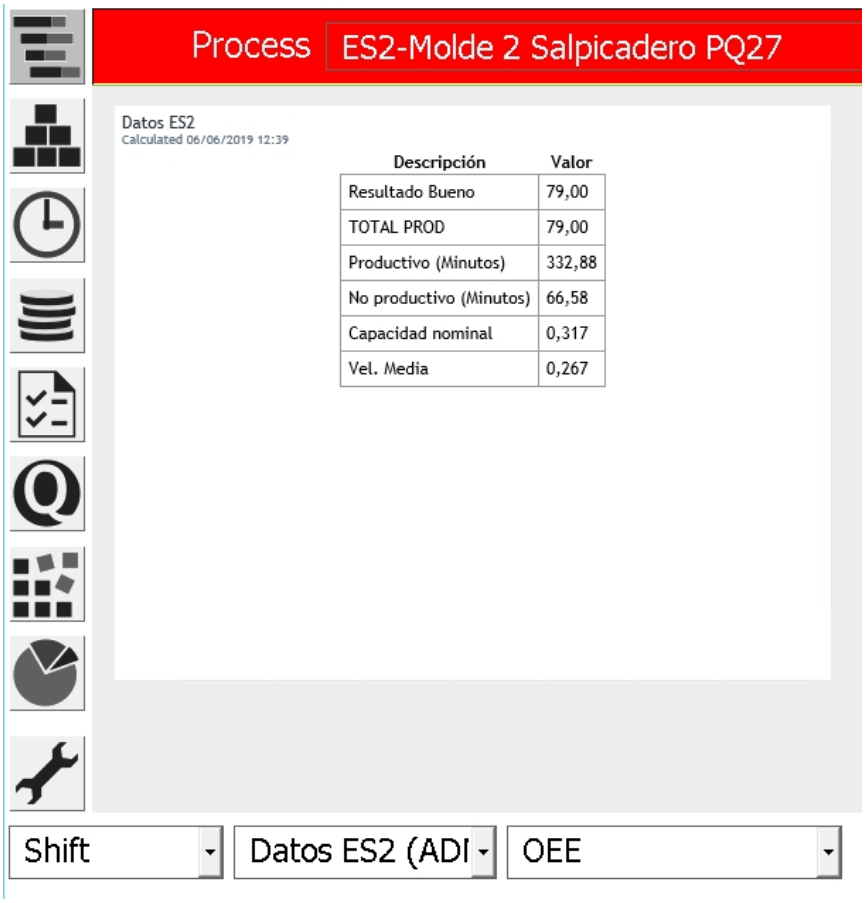
2. A window will appear in which you will have to enter the **name** of the control panel, the **devices** from which the control panel can be visible and the **processes** from which it can be viewed. Multiple devices or processes can be selected by holding down the CTRL key. Press OK to create the dashboard.



3. On the Favorites tab, add the report to the new Control Panel by selecting it from the drop-down list.



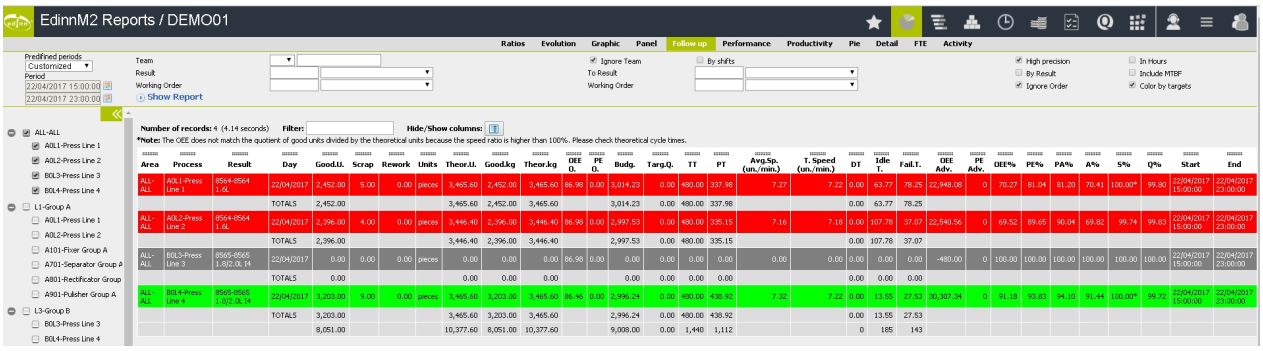
4. Now the report can be selected on the main screen of edinn by the users and within the processes selected in point 2 of **Step 3**.



Follow up

This report allows an aggregated analysis of the elapsed time.

Mark the option **Colors by targets** if you want to see each individual line of the report in colors. Green will indicate that the ratios of that result where above the organization targets; yellow and red will indicate they were not. By this, you can gain a lot of time by only focusing your efforts where there are losses. For this option to work you need to have product targets configured at results configuration or the relation process-results configuration.

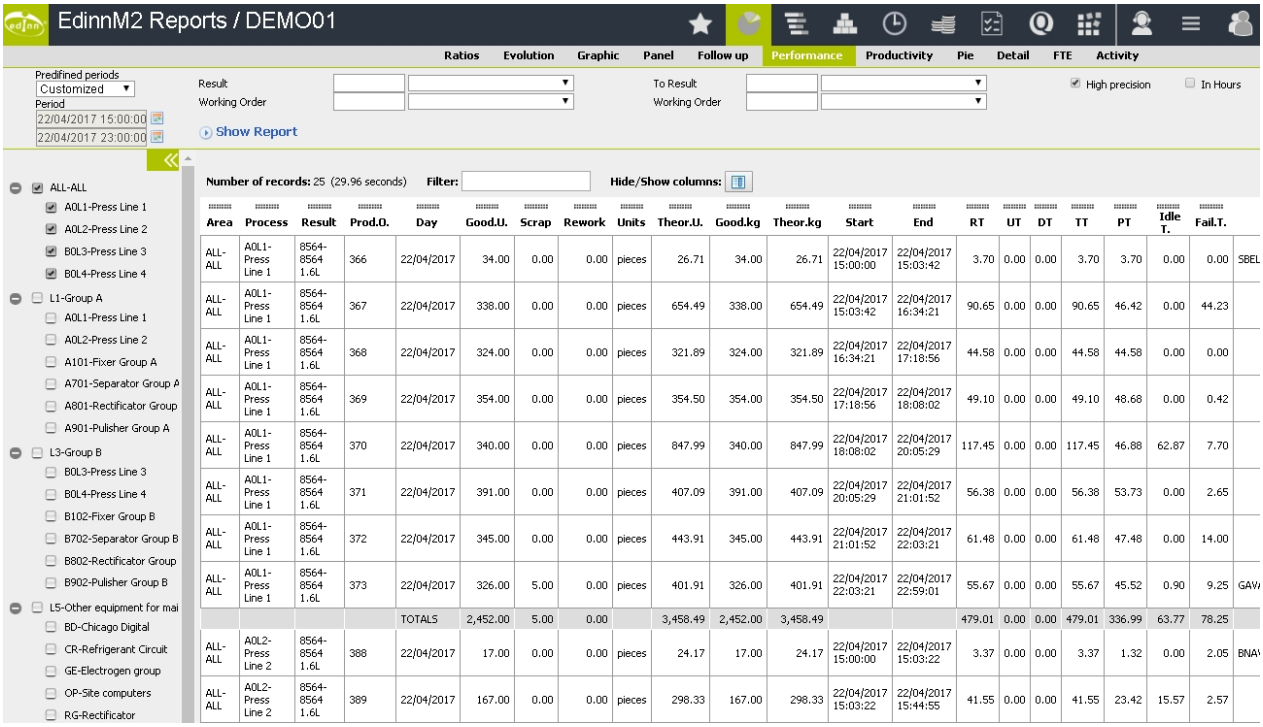


Selectors are:

- **Please see:** "Common aspects"
- **Team**: if you want to see only information relative to the times where that specific team was working.
- **Result from and until**: select the result from and to that you want to focus your report on.
- **Order from and until**: select your desired from and to order.
- **Ignore Team**: do not show nor consider the team.
- **By Shifts**: aggregates by shift.
- **By Results**: aggregates by results.
- **Ignore Order**: ignore order.
- **Include MTBF**: includes MTBF information.
- **Color by targets**: shows records in colors (green, yellow, red) considering the product targets defined in configuration of relations Process-Results.

Performance

This report allows an aggregated analysis of the results and orders produced in a period.

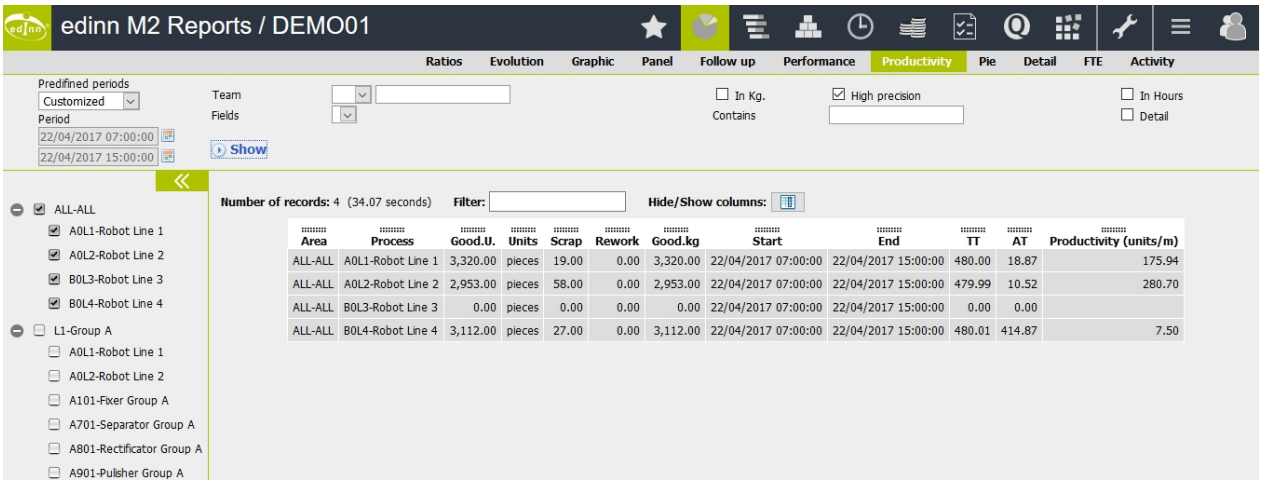


Selectors are:

- **Please see:** "Common aspects"
- **Result:** select the result that you want to focus your report on.
- **Order from and to:** select your desired from and to order.

Productivity

This report allows an aggregated analysis of the productivity of the works that have occurred in a period.

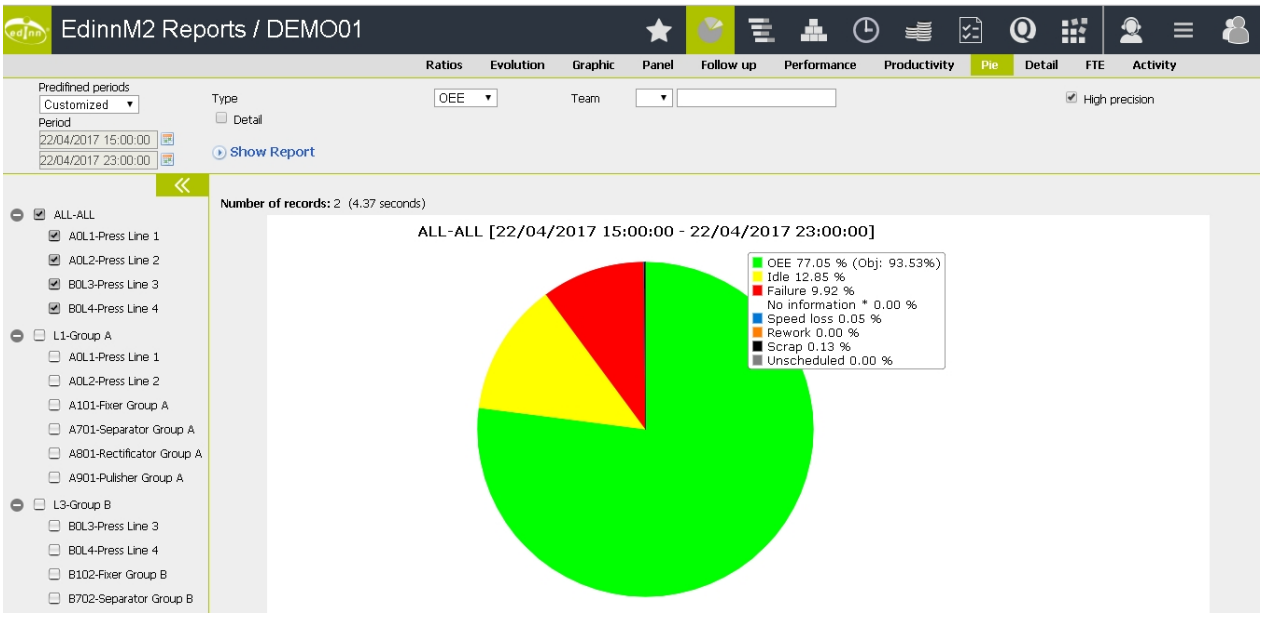


Selectors are:

- **Please see:** "Common aspects"
- **Team:** if you want to see only information relative to the times where that specific team was working.
- **Fields:** if you want fo filter production performed with a certain field with a certain value.
- **Contains:** what must be contained in the field selected in the previous selector.
- **In kg:** if you want to show results in kgs.

Pie

This report shows in a graphic pie the distribution of the production efficiency losses. This report can also be accessed by clicking on the records of the ratios report.



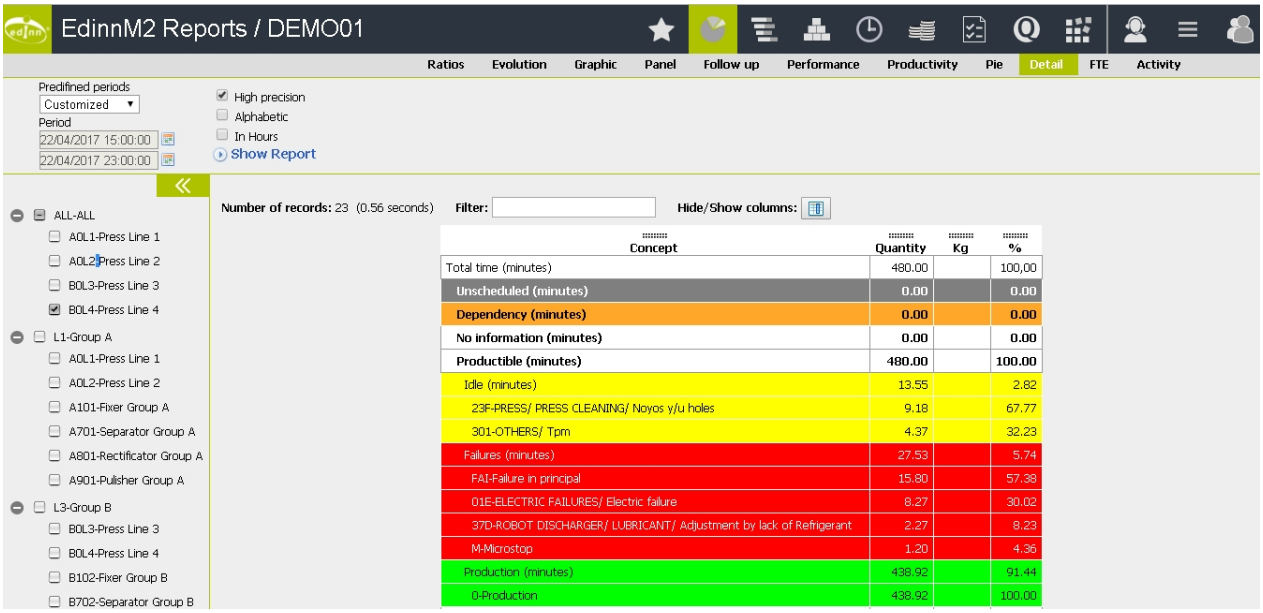
Selectors are:

- **Please see:** "Common aspects"
- **Type**: OEE, Status (time usage) Results.
- **Team**: if you want to see only information relative to the times where that specific team was working.

NOTE: to understand the 'No information' label, please refer to the 'Not recorded' report.

Detail

This report shows 100% detail of the losses of a process, regarding its production efficiency. It details all losses that are in the red boxes of losses of the document OEE in this manual.



Selectors are:

- **Please see:** "Common aspects"
- **Alphabetic**: orders the report in alphabetical order instead of in quantitative order.

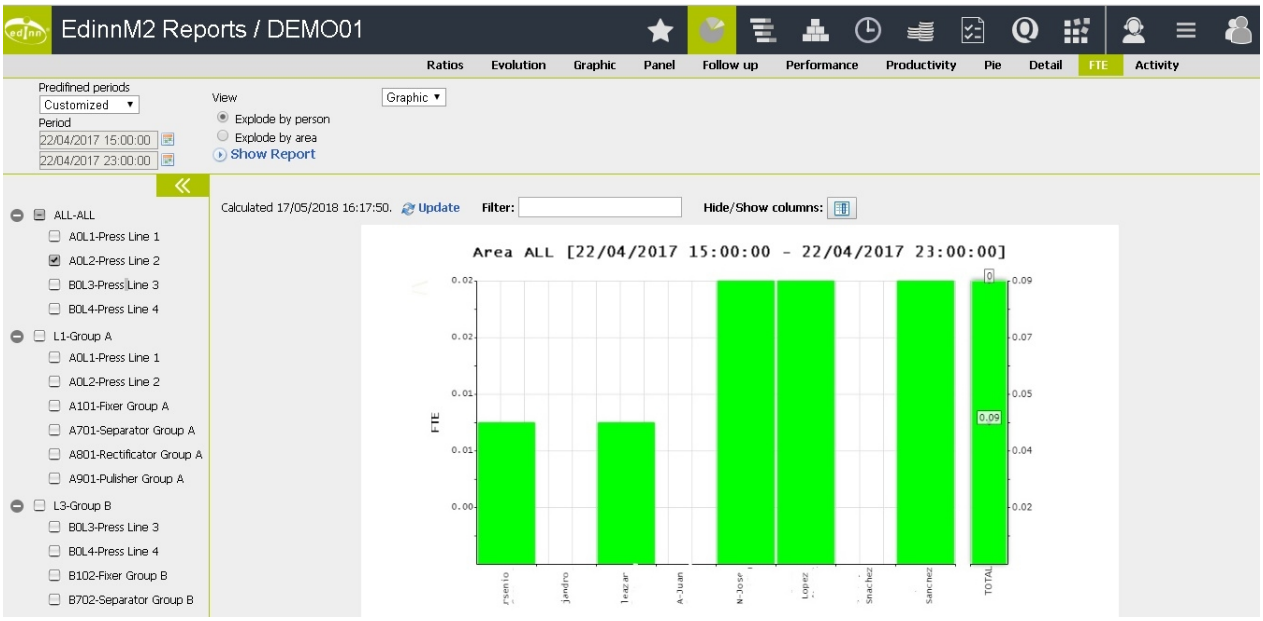
NOTE: to understand the 'No information' label, please refer to the 'Not recorded' report.

FTE

FTE (*Full Time Equivalent*) is a measure used in economical or human resources matters. It is obtained by dividing the working hours or various part-time employees by the quantity of hours of the complete legal labour period.

This report shows the FTEs in a bar graphic, indicating using colors if the working time was used for production or services.

This report is calculated both for processes and areas (groups of processes).

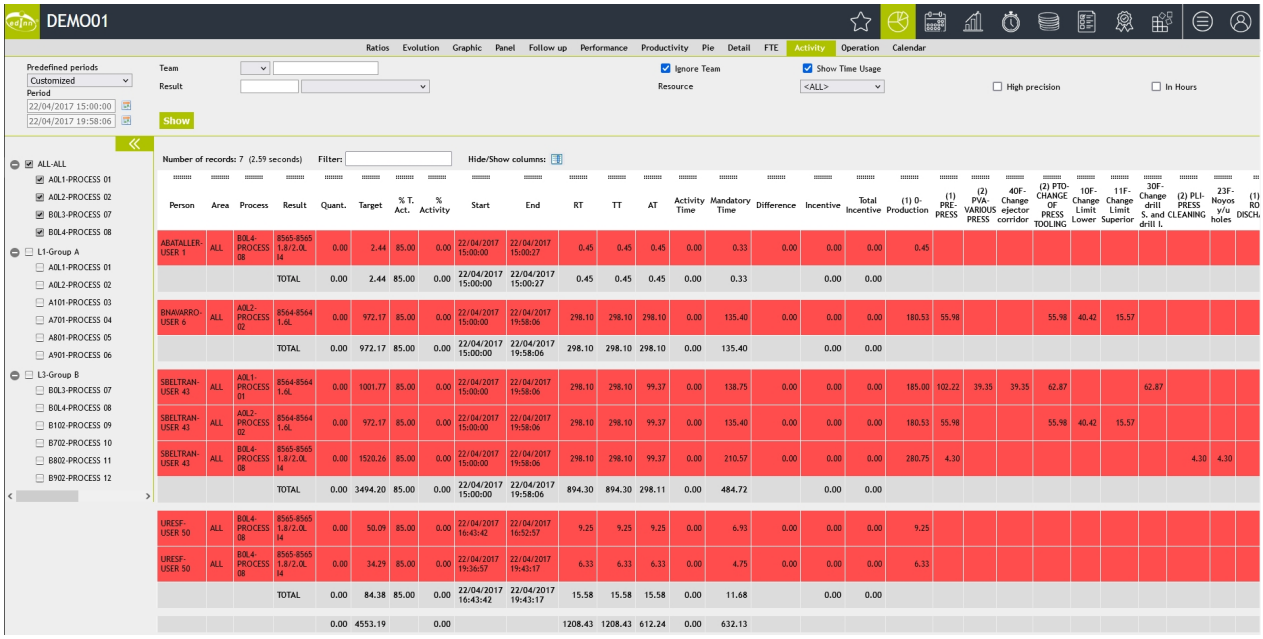


Selectors are:

- **Please see:** "Common aspects"
- **Explode by area or by person**: allows to differentiate if the analysis is needed for persons or for areas.

Activity

This report shows the activity ratios of the working personnel. It is useful because as other edinn reports do, it shows the aggregated ratios and results of every working person in colors. By this, we only have to dedicate time to analyze the losses of the red and yellow records.

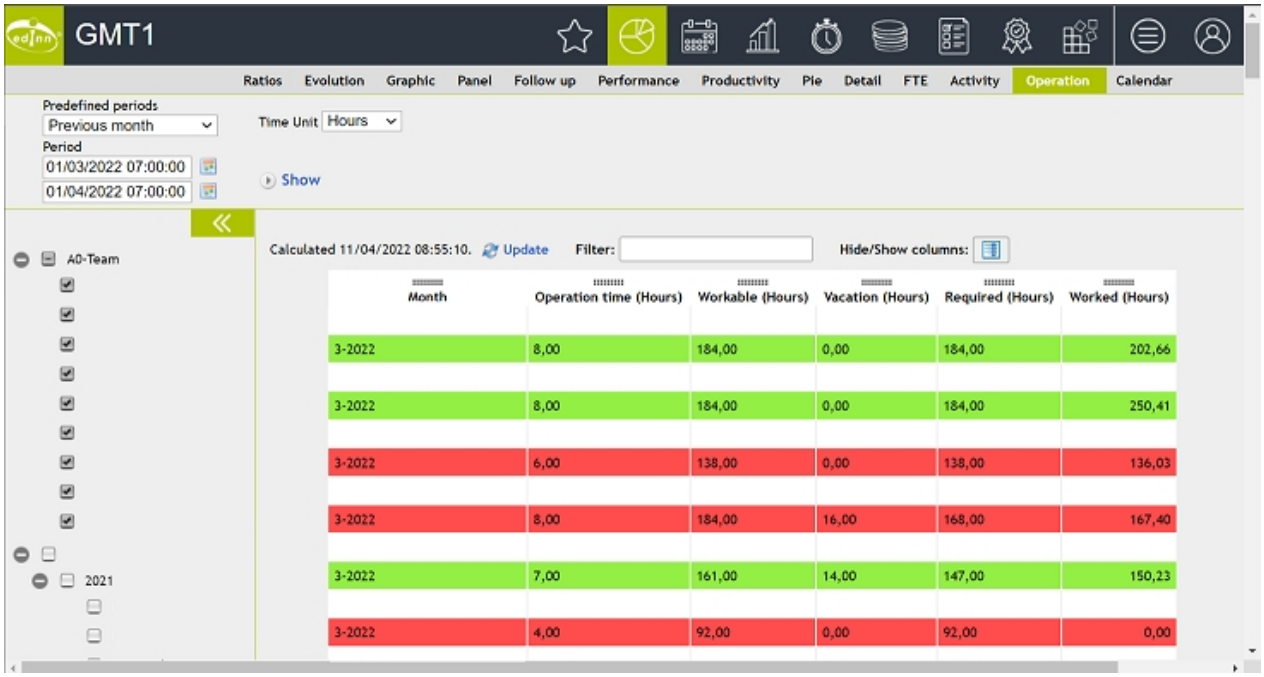


Selectors are:

- **Please see:** "Common aspects"
- **Team**: if you want to see only information relative to the times where that specific team was working.
- **Result**: select the result that you want to focus your report on.
- **Ignore Team**: do not show nor consider the team.
- **Show Time Usages:** if checked, it will show the states the process was in while the resource was in it.
- **Person**: focus the analysis in one person.

Operation

This report explains the availability of the processes which have Operation Time, when it was configured in the processes or in the PSR Relations. Normally these processes will be persons.



Selectors are:

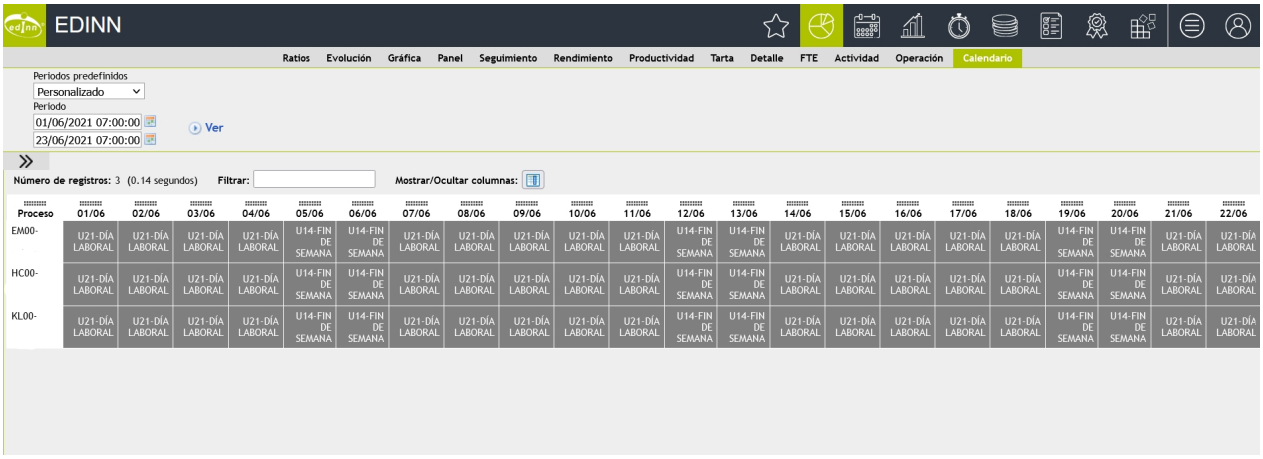
- **Please see:** "Common aspects"
- **Time unit**: unit in which the data will be shown.

Columns are:

- **Month / Name**: of the process.
- **Operation Time**: as configured in PSR Relations or in the Process.
- **Workable**: considering the operation time, is the sum of all the statuses of the calendar which, in the statuses configuration, do not belong to classes neither weekend nor holiday.
- **Vacation**: considering the operation time, is the sum of all the statuses which, in the statuses configuration, belong to the class holiday.
- **Required**: is the difference between Workable minus Vacation columns.
- **Worked**: is the sum of all the statuses configured as of the productive class.

Calendar

This report shows te processes calendar as configured in the calendar.



Selectors are:

- **Please see:** "Common aspects"

Schedule

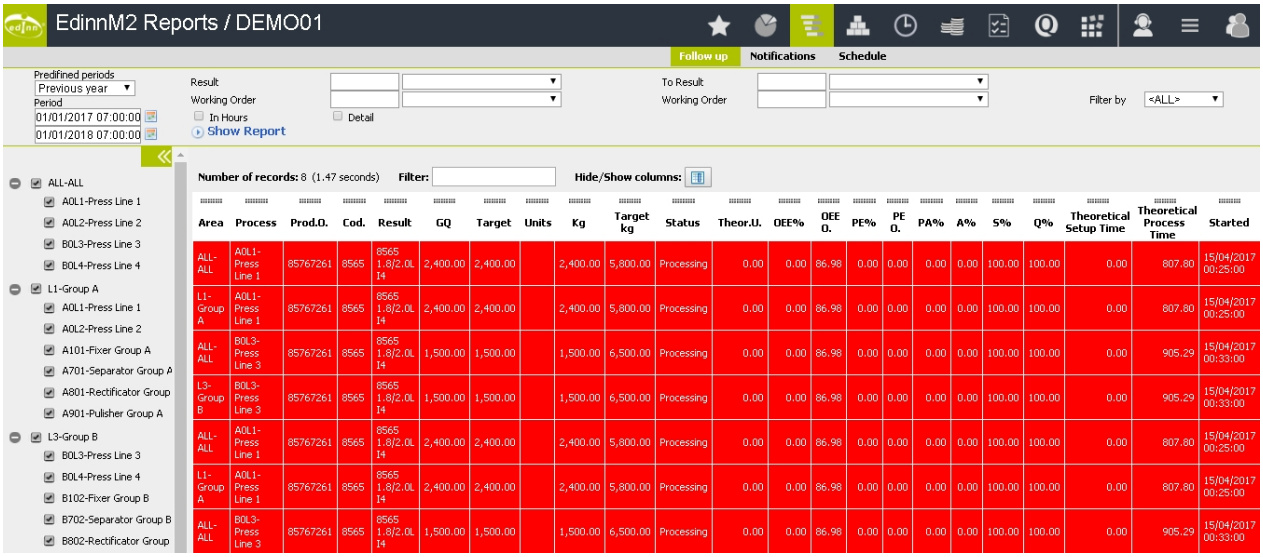
This is the section for the reports regarding Schedule.

Follow up

This report shows aggregated information of the orders, their status and main KPIs.

> **NOTE**: In the reports of the Schedule section, the selected areas filter orders/tasks which belong to those areas. Therefore, if you do not know to which area does an order or task belong, do not select any area.

> **NOTE**: This report shows only work orders/tasks which have activity, by the selected processes, in the period.



Selectors are:

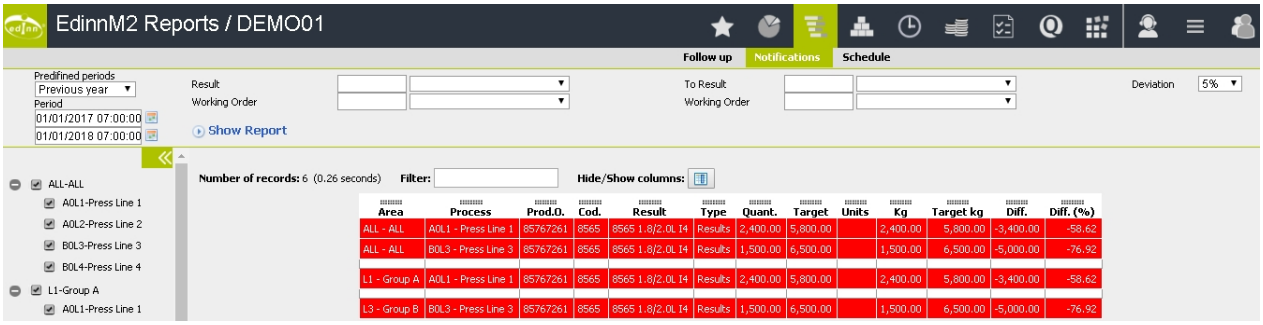
- **Please see:** "Common aspects"
- **Result from and until**: the date and time range of results to be analyzed.
- **From and until order**: the date and time range of orders to be analyzed.
- **Filter by**: status of the orders.
- **In hours**: shows times in hours.
- **Detail**: shows additional detail.

Notifications

This report shows the notifications of the schedule, this is, production, inputs and outputs.

NOTE: In the reports of the Schedule section, the selected areas filter orders/tasks which belong to those areas. Therefore, if you do not know to which area does an order or task belong, do not select any area.

NOTE: This reports shows only orders/tasks which have activity, by the selected processes, in the period.



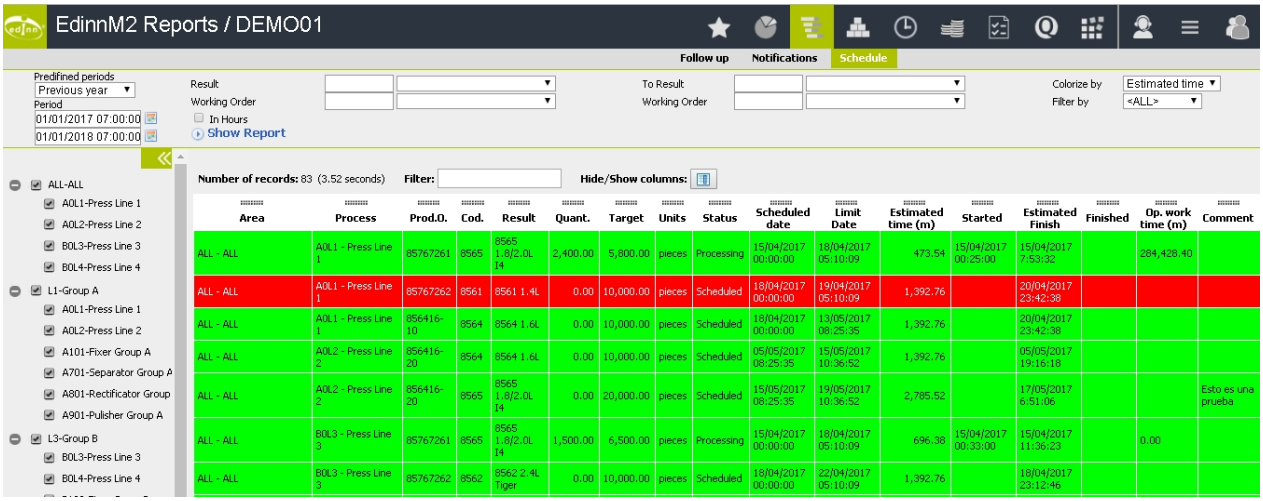
Selectors are:

- **Please see:** "Common aspects"
- **Result from and until**: the date and time range of results to be analyzed.
- **Deviation**: the % of deviation comparing with the scheduled values in order to show the order in red.
- **From and until order**: the date and time range of orders to be analyzed.

Schedule

This report shows the scheduled orders, allowing to see them by colours upon their status or their delay.

> **NOTE**: In the reports of the Schedule section, the selected areas filter orders/tasks and show only those which belong to those areas. Therefore, if you do not know to which area does an order/task belong, do not select any area.



Selectors are:

- **Please see:** "Common aspects"
- **Result from and until**: the range of results to be analyzed.

- **Colorize by**: estimated time (shows in red those orders that are going to be delayed according to their estimated time) or by status (shows the orders in colors according to their status).
- **From and until order**: the date and time range of orders to be analyzed.
- **Filter**: filters orders according to their status.

Results

This is the section for the reports regarding Results.

Path

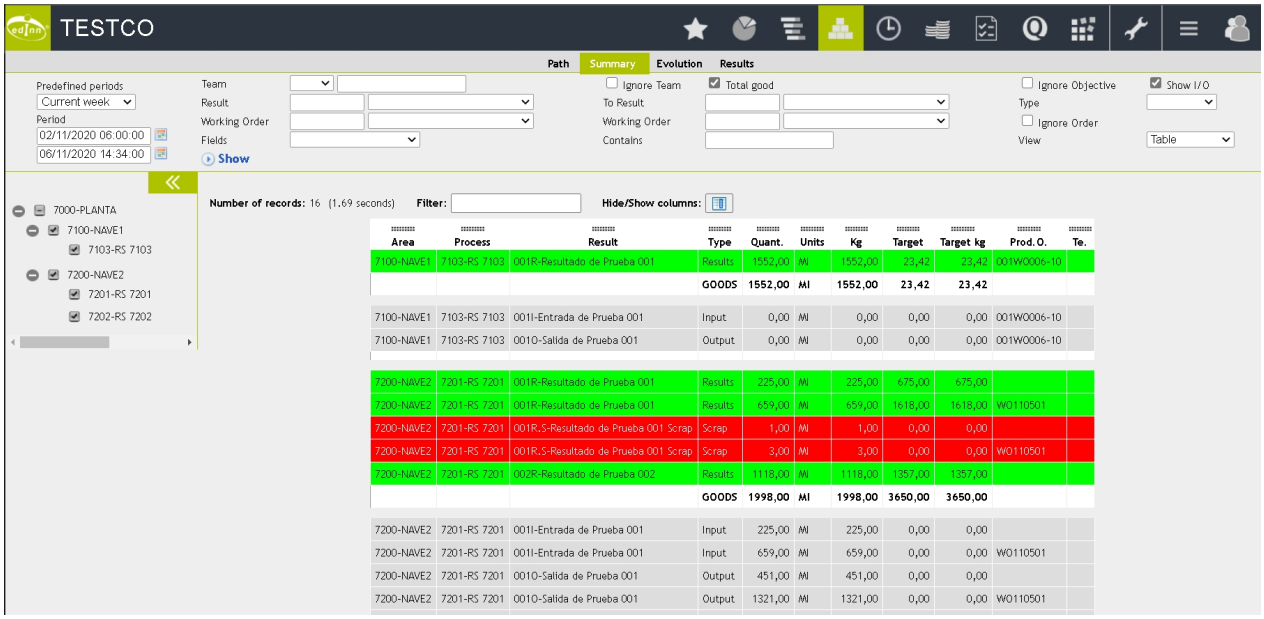
Shows the "path" of production. Please refer to the "Path".

Selectors are:

- **Please see:** "Common aspects"

Summary

Shows a quick summary of the Results obtained.

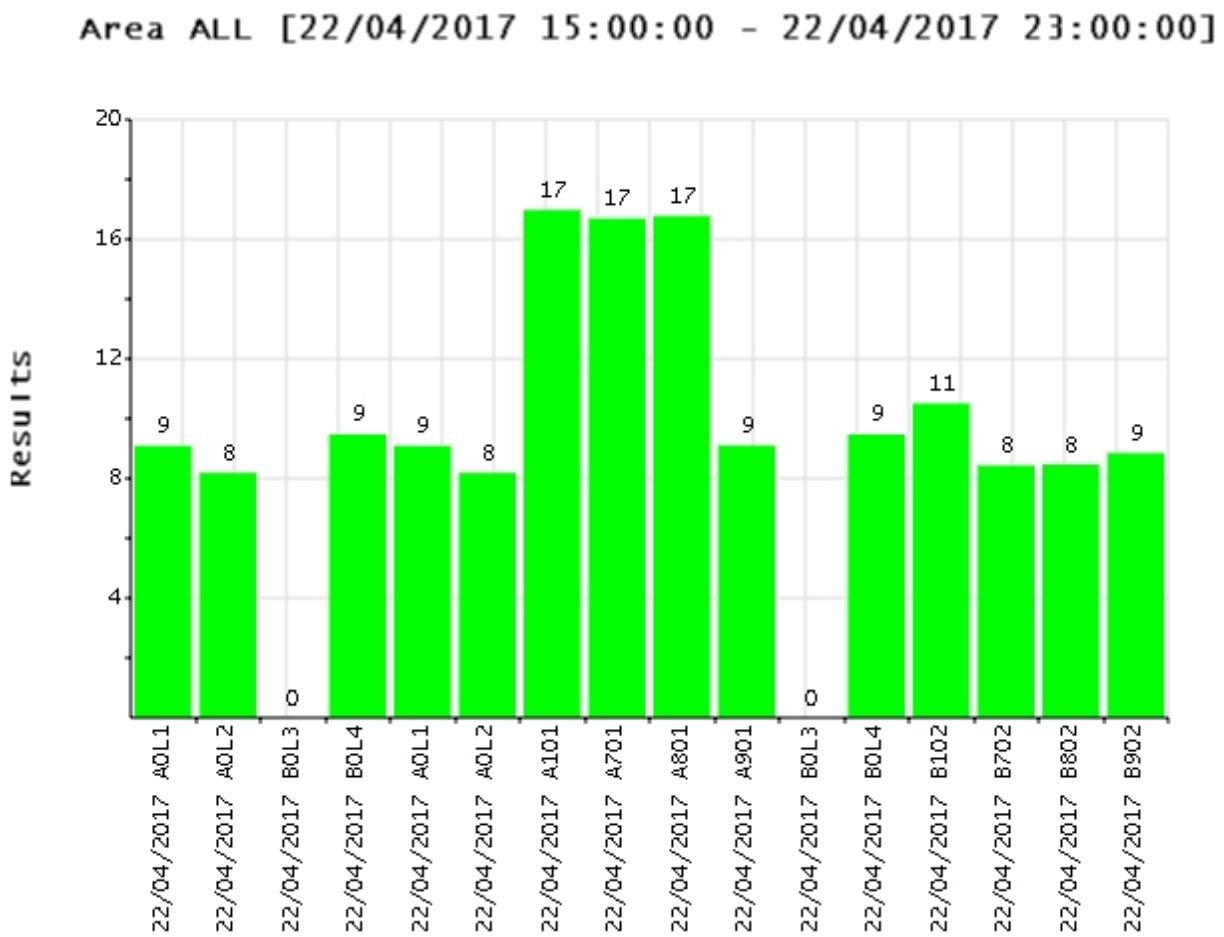


Selectors are:

- **Please see:** "Common aspects"
- **Team**: if you want to see only information relative to the times where that specific team was working.
- **Ignore Objective**: do not show objectives (targets) of results. By marking this, will faster the report.
- **Show I/O**: shows inputs and outputs of each result.
- **Result from and until**: select the result from and to that you want to focus your report on.
- **Order from and until**: select your desired from and to order.
- **Type**: type of result.
- **Ignore Team**: mark it to ignore and hide the team column.
- **Total good**: totalize only the *good results*.
- **Ignore Order**: ignore the order.

Evolution

This report shows the evolution of the good results produced by a process.

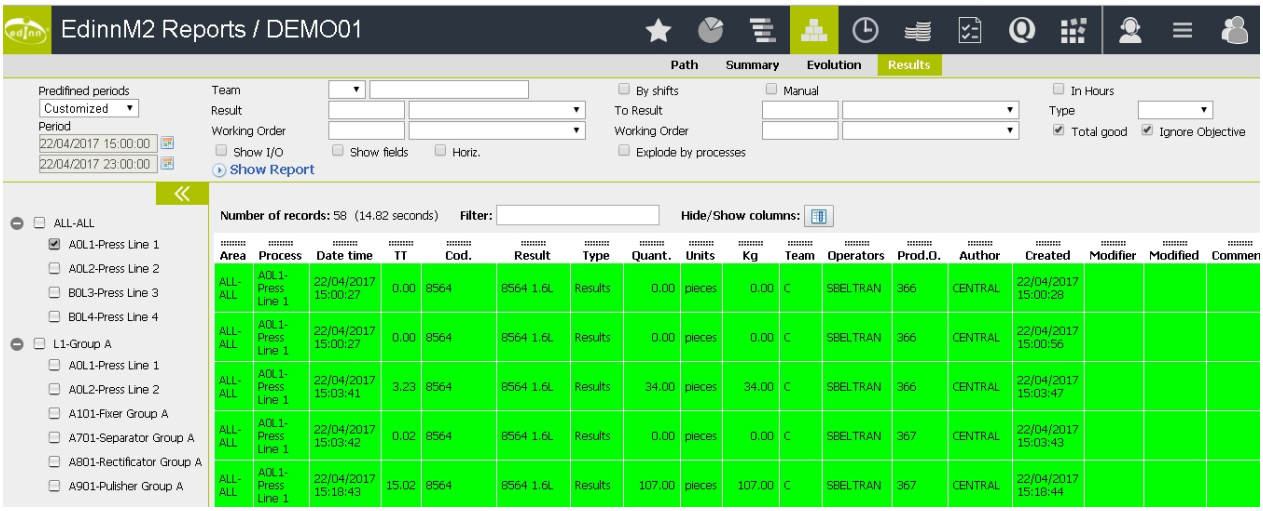


The selectors are:

- **Please see:** "Common aspects"
- **By Shifts**: aggregates by shift.
- **Explode by process:** shows the results of each process in the area.

Results

This report shows the most detailed list of results obtained. As it shows all the records in colors, it is very useful to know in just a quick view, all the results that were produced, their type, etc. As all the detailed reports, if there are user comments on a record, the comment will be shown at the last column on the right.



Selectors are:

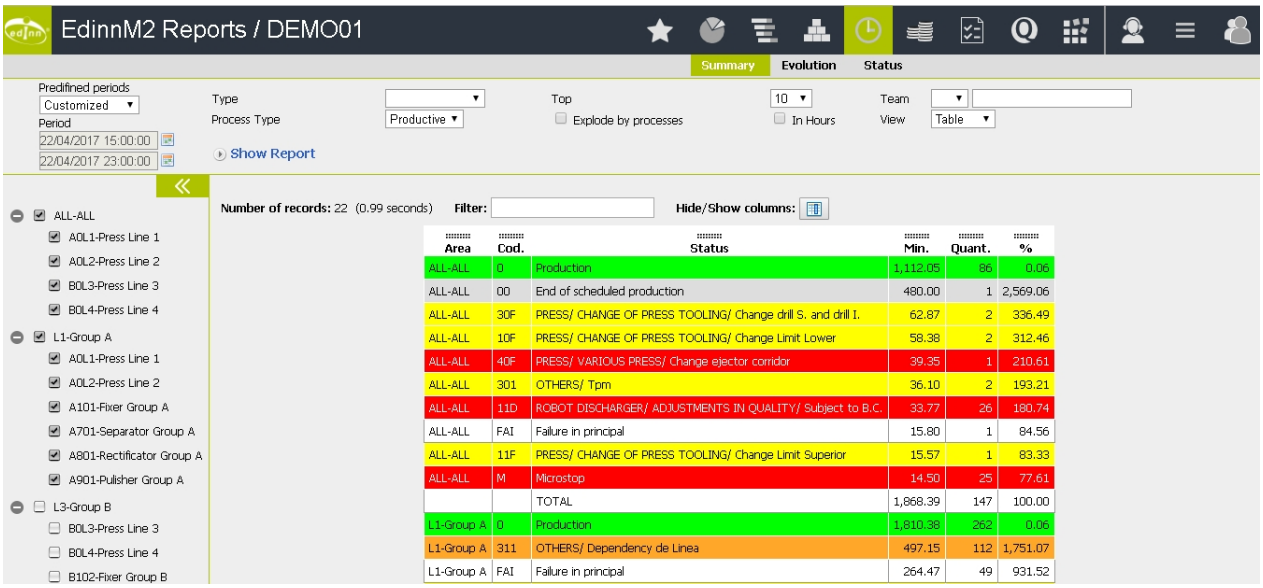
- **Please see:** "Common aspects"
- **Team**: if you want to see only information relative to the times where that specific team was working.
- **Result from and until**: select the result from and to that you want to focus your report on.
- **Order from and until**: select your desired from and to order.
- **Show I/O**: shows inputs and outputs of each result.
- **Show fields**: show information fields associated with each result.
- **Horiz.**: shows the fields in horizontal format. This is useful for excel sheets analysis.
- **By Shifts**: separates information by shift.
- **Manual**: shows records that have been modified manually by a worker.
- **Type**: type of result.
- **Total good**: totalize only the good results.
- **Ignore Objective**: do not show objectives (targets) of results. By marking this, will faster the report.

Status

This is the section for the reports regarding Status.

Summary

This report shows the aggregation of all the statuses that a process has passed.



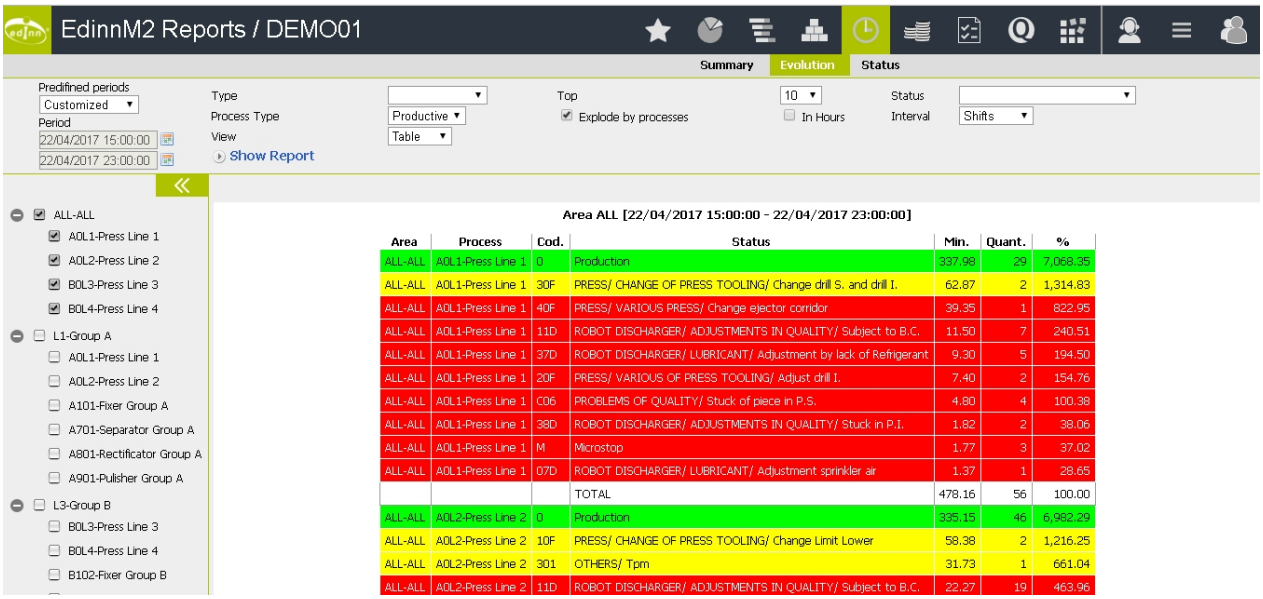
Selectors are:

- **Please see:** "Common aspects"
- **Team**: if you want to see only information relative to the times where a specific team was working.
- **Type**: type of status you want to filter.
- **Top**: maximum number of statuses to show. For example: selecting type = 'Not productive' and Top = 5 is useful to focus the analysis on the main 5 problems of a process.

WARNING: Total times of reports Status\Summary and Status\Status could not match because Status\Status is a detailed report and includes all the time of the records shown, whilst Status\Summary only includes the time of each record which is really inside the period, and not all the time of the record.

Evolution

This report shows the evolution of the aggregation of all the statuses that a process has passed. It is useful to see how the process is improving and thus certain bad statuses are reducing.

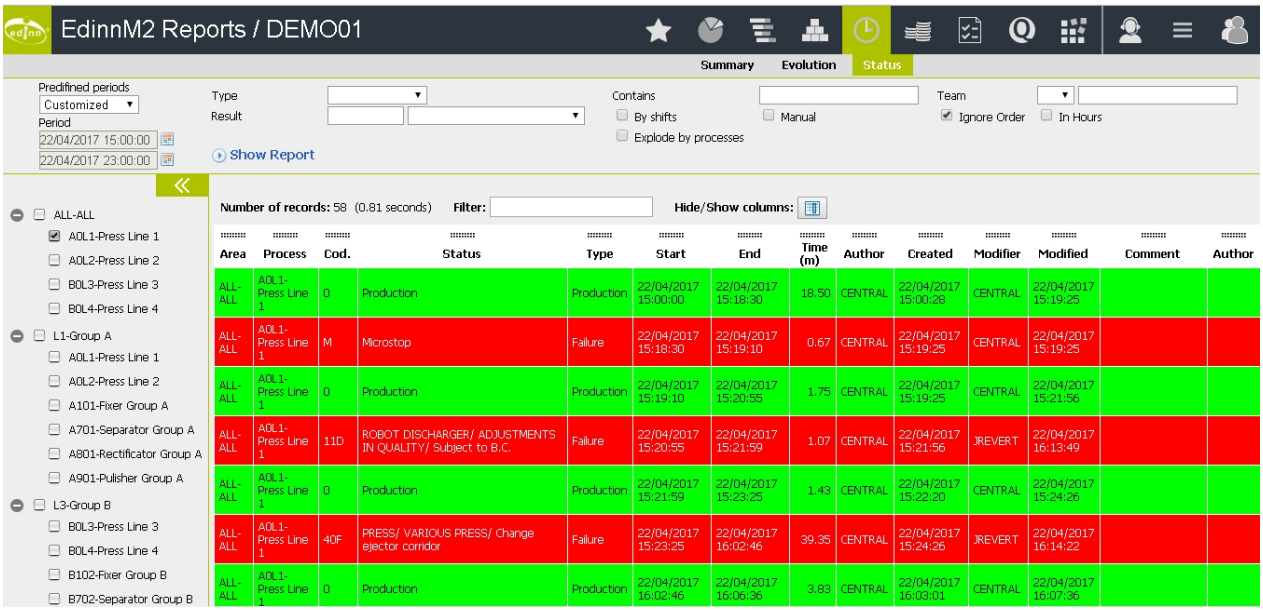


Selectors are:

- **Please see:** "Common aspects"
- **Type**: type of status you want to filter.
- **Process type:** if it is productive or is a service.
- **Top**: maximum number of statuses to show. For example: selecting type = 'Not productive' and Top = 5 is useful to focus the analysis on the main 5 problems of a process.
- **Explode by process:** shows the results of each process in the area.
- **Status (Time usage)**: useful to focus the analysis in only one status.
- **Interval**: allows to analyze the evolution of statuses by shift, day, week, month, etc.

Status

This report shows the most detailed list of statuses that the process has passed. As it shows every record in colors, it is very useful to know in a quick view, the statuses that the process passed, any incidence, etc. As all the detailed reports, if there are user comments on a record, the comment will be shown at the last column on the right.



Selectors are:

- **Please see:** "Common aspects"
- **Type**: type of status you want to filter.
- **Contains**: show only those statuses that contain this text in their id or their description. You can put != at the beginning to indicate to select those records which do not contain the indicated text.
- **Result**: select the result that you want to focus your report on.
- **By shifts**: separate information by shifts.
- **Manual**: show only records that have been manually modified by a user.
- **Explode by process:** shows the results of each process in the area.
- **Team**: if you want to see only information relative to the times where a specific team was working.
- **Ignore Order**: ignore the order.

WARNING: Total times of reports Status\Summary and Status\Status could not match because Status\Status is a detailed report and includes all the time of the records shown, whilst Status\Summary only includes the time of each record which is really inside the period, and not all the time of the record.

Consumption

This is the section for the reports regarding Consumption.

Please see previous document to correctly distinguish between consumptions and inputs.

Panel

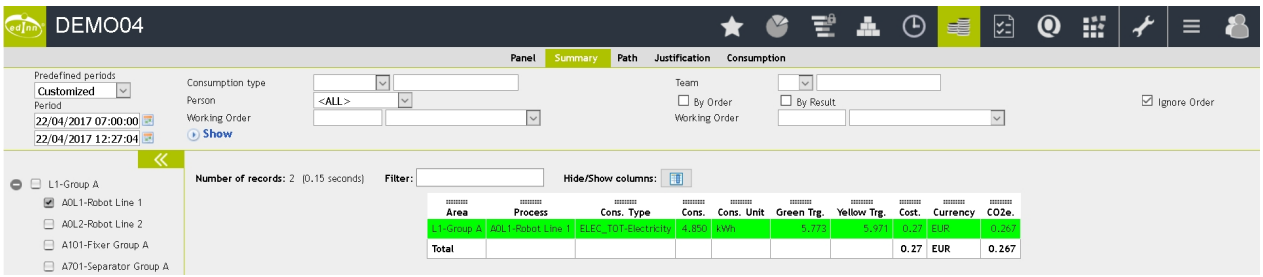
This report shows the actual consumption of a process for a certain type of consumption. It is useful to view in real time how the different changes in the process affect its consumption. The report will self-refresh according to the rate of refreshment of data on the server. Technical support can tell you this rate.

Detail is as follows:

- **Please see:** "Common aspects"
- **Consumption type**: select the consumption type.

Summary

This report shows an aggregation of all consumptions that occurred. Using the exclusive edinn technology and combining consumption and status of process, this report shows in colors each summary of consumption. By this, the user must only analyse those records which appear in red.

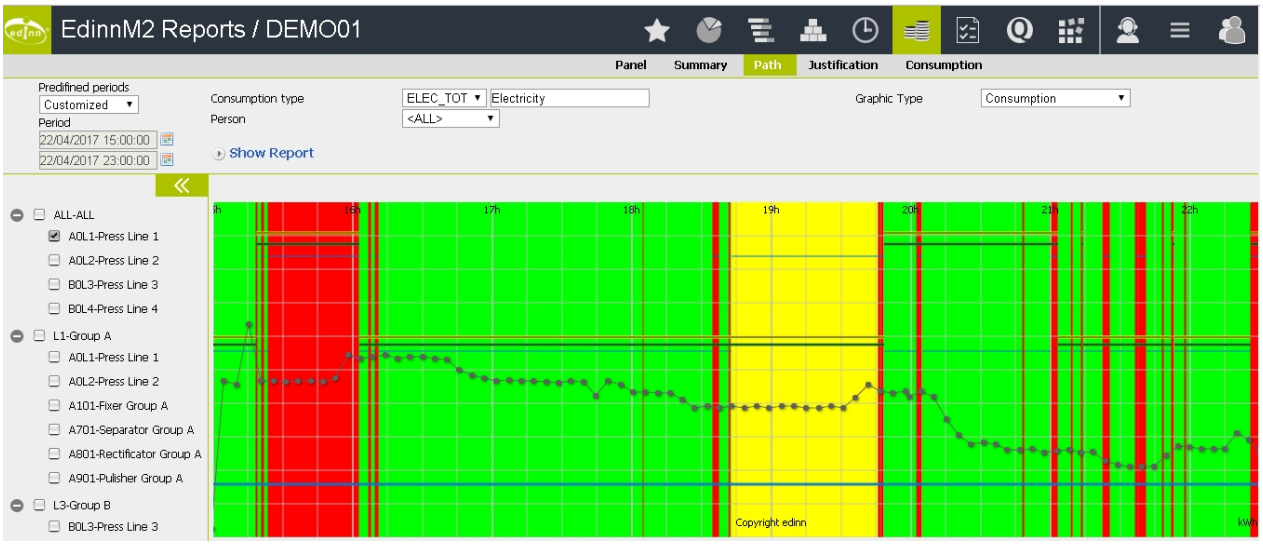


Selectors are:

- **Please see:** "Common aspects"
- **Team**: shows information relative to the periods when a specific team was working.
- **Consumption type**: focuses on a specific consumption type.
- **Person**: shows only consumptions associated with a person.
- **By Order:** groups and totalizes consumptions by the order.
- **Order:** filters by a range of orders.
- **Ignore Order:** ignores the order and does not show its column.

Path

This report shows the path of consumption. This report is very useful to understand what is happening in the process regarding each type of consumption. As shown below, it combines the consumption values with the status of the process. Nominal and target consumptions vary according to the status of the process.



Selectors are:

- **Please see:** "Common aspects"
- **Person**: if you want to see only consumptions associated with a person.
- **Consumption type**: if you want to focus on a specific consumption type.
- **Graphic type**: allows to see just the consumption path, or a combination of the consumption and production path. This is very useful to visually and completely understand the patterns of efficiency of a process.

Justification

This report shows the justifications to excessive consumptions introduced by users.

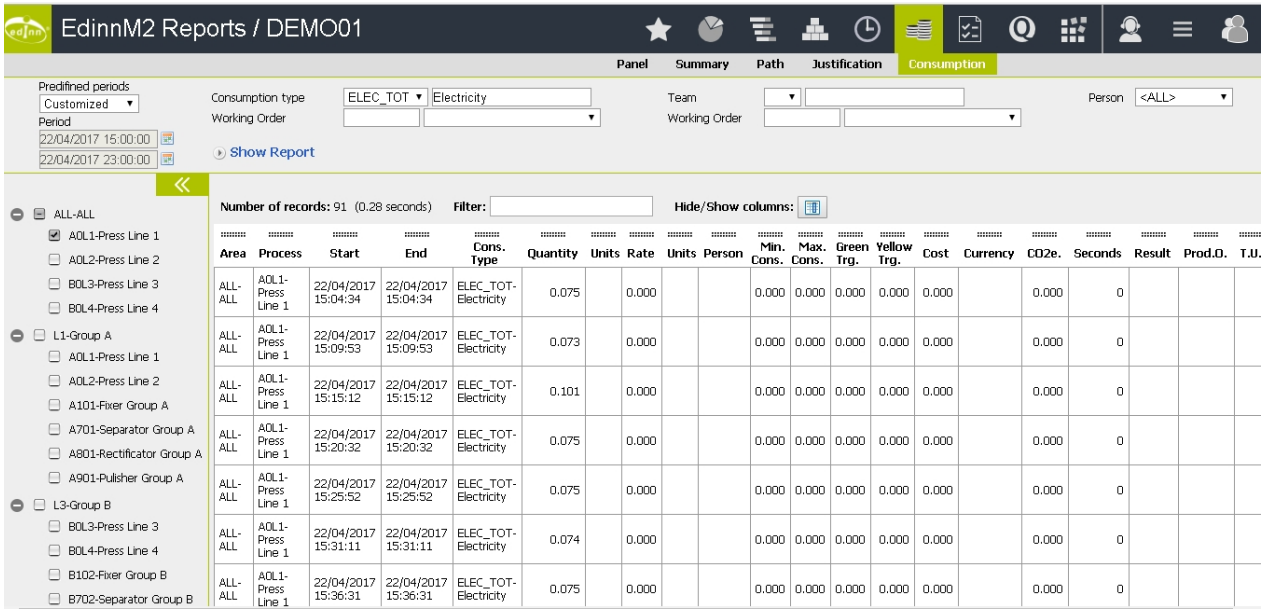
Selectors are:

- **Please see:** "Common aspects"
- **Consumption type**: if you want to focus on a specific consumption type.
- **Person**: if you want to see only consumptions associated with a person.

- **Justification type**: normal or excessive consumption.

Consumption

This report shows the detail of the consumptions that occurred in a period. As it shows the information in colors, it is very useful to know consumptions issues in a quick view. The colors take into account, among others, the active result and status of the process, so that it means truly profitable when is green and losses when is red. As all the detailed reports, if there are user comments on a record, the comment will be shown at the last column on the right.



Area	Process	Start	End	Cons. Type	Quantity	Units	Rate	Units	Person	Min. Cons.	Max. Cons.	Green Trg.	Yellow Trg.	Cost	Currency	CO2e.	Seconds	Result	Prod.O.	T.U.
ALL-ALL	A0L1-Press Line 1	22/04/2017 15:04:34	22/04/2017 15:04:34	ELEC_TOT-Electricity	0.075		0.000			0.000	0.000	0.000	0.000	0.000		0.000	0			
ALL-ALL	A0L1-Press Line 1	22/04/2017 15:09:53	22/04/2017 15:09:53	ELEC_TOT-Electricity	0.073		0.000			0.000	0.000	0.000	0.000	0.000		0.000	0			
ALL-ALL	A0L1-Press Line 1	22/04/2017 15:15:12	22/04/2017 15:15:12	ELEC_TOT-Electricity	0.101		0.000			0.000	0.000	0.000	0.000	0.000		0.000	0			
ALL-ALL	A0L1-Press Line 1	22/04/2017 15:20:32	22/04/2017 15:20:32	ELEC_TOT-Electricity	0.075		0.000			0.000	0.000	0.000	0.000	0.000		0.000	0			
ALL-ALL	A0L1-Press Line 1	22/04/2017 15:25:52	22/04/2017 15:25:52	ELEC_TOT-Electricity	0.075		0.000			0.000	0.000	0.000	0.000	0.000		0.000	0			
ALL-ALL	A0L1-Press Line 1	22/04/2017 15:31:11	22/04/2017 15:31:11	ELEC_TOT-Electricity	0.074		0.000			0.000	0.000	0.000	0.000	0.000		0.000	0			
ALL-ALL	A0L1-Press Line 1	22/04/2017 15:36:31	22/04/2017 15:36:31	ELEC_TOT-Electricity	0.075		0.000			0.000	0.000	0.000	0.000	0.000		0.000	0			

Selectors are:

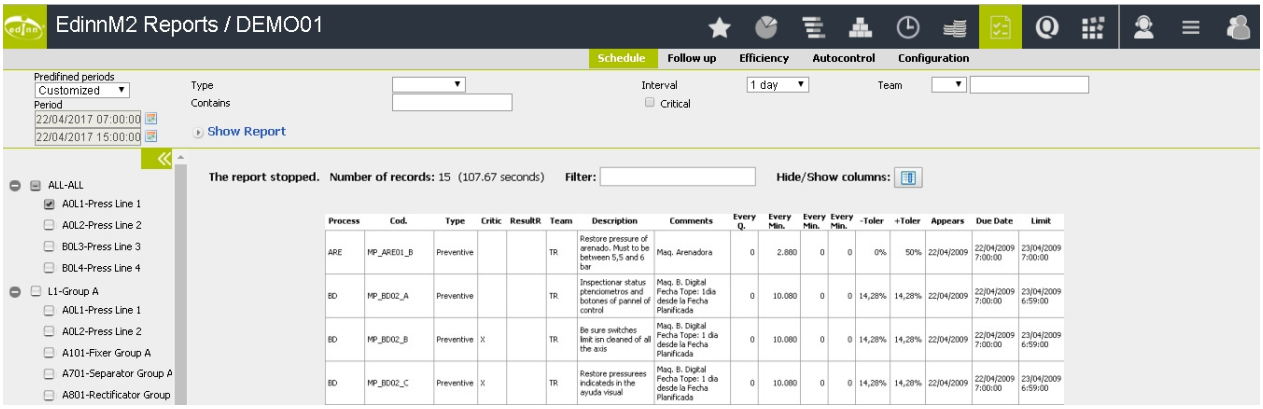
- **Please see:** "Common aspects"
- **Consumption type**: if you want to focus on a specific consumption type.
- **Team**: if you want to see only information relative to the times where a specific team was working.
- **Person**: if you want to see only consumptions associated with a person.

Autocontrol

This is the section for the reports regarding autocontrol.

Schedule

This report performs a simulation and shows the autocontrol tasks that will be launched in the future. It is very useful to predict the work load of tomorrow, next week, next month, a special period, etc.



Selectors are:

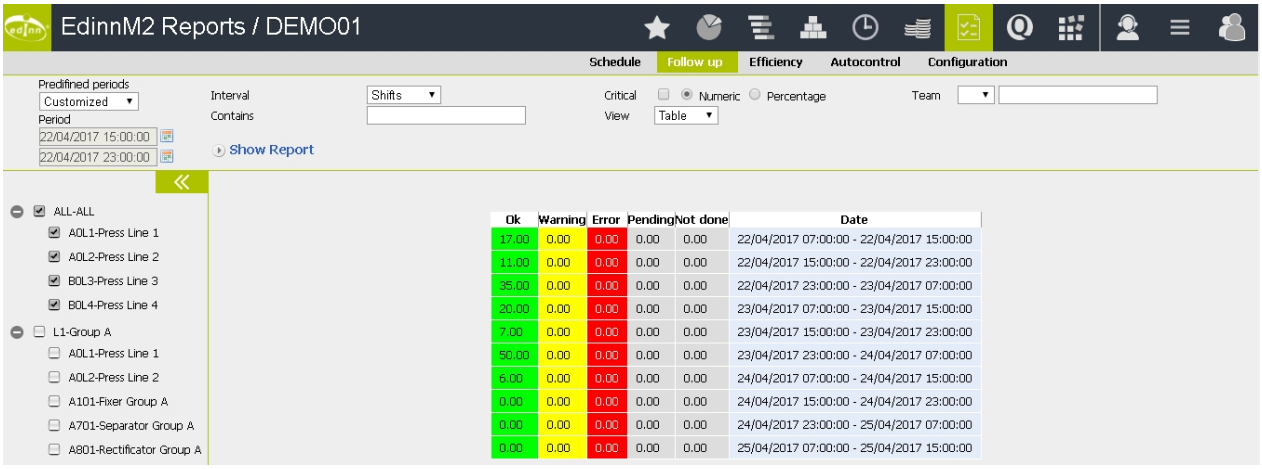
- **Please see:** "Common aspects"
- **Type**: the type of autocontrol task that you want to simulate.
- **Contains**: simulate only those tasks that contain certain text. You can put != at the beginning to indicate to select those records which do not contain the indicated text.
- **Interval**: to simulate future periods, the report needs to know the interval of the simulation. The shorter the period, the more precise the simulation, but it will take much longer to calculate the report.
- **Critical**: if the tasks to simulate are only the critical ones.
- **Team**: if you want to simulate only tasks relative to a specific team.

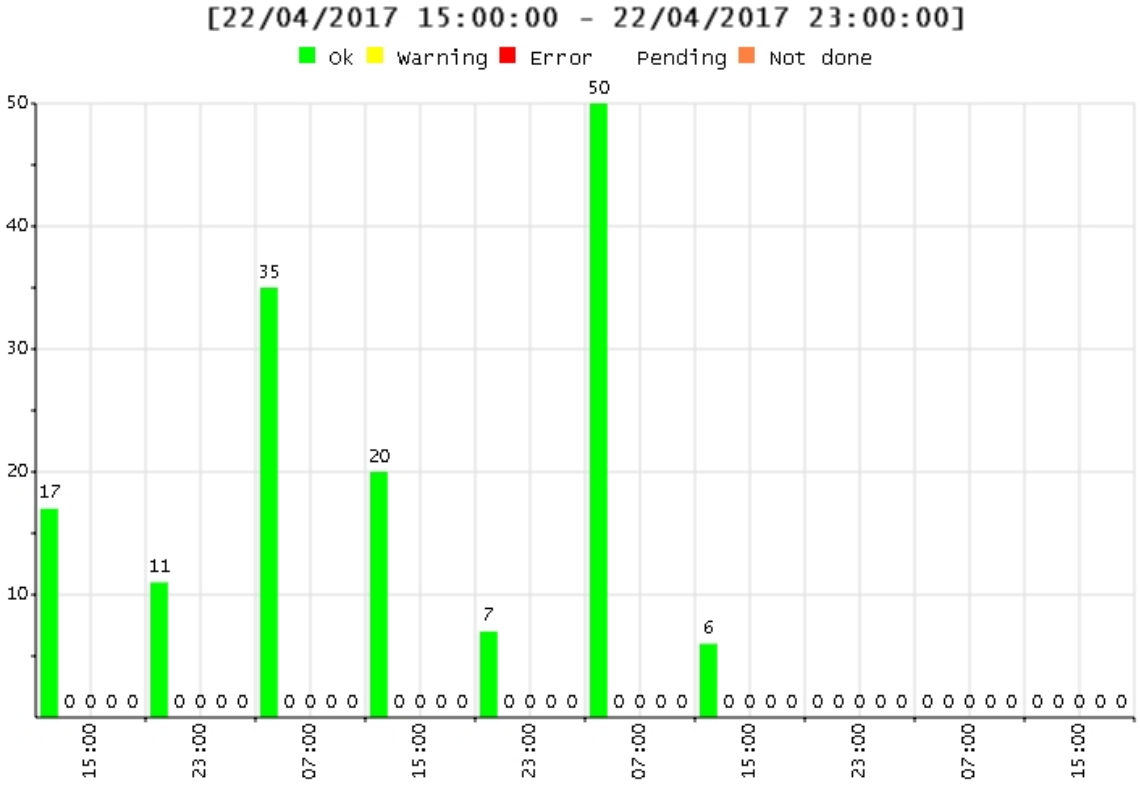
> **Note**: For those tasks which are programmed by production quantity, the system will take into account the nominal speed multplied by the OEE, different than zero, of the longest period.

Follow up

This report shows is useful to know the aggregated of the status of all the autocontrol tasks. It is very useful to know in a few seconds how many tasks have been performed, how many had incidents and how many were Not Done (note that the Not done category includes the Expired tasks).

> **NOTE**: Tasks become 'Expired' when they are 'Not done' and another task of the same Id is created.





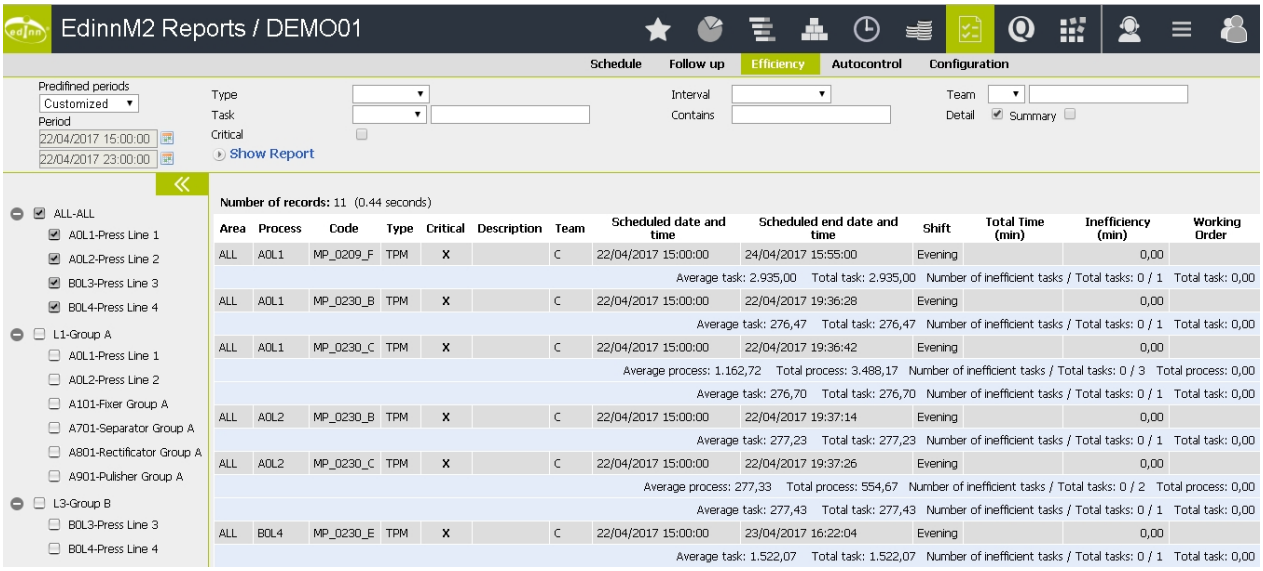
Selectors are:

- **Please see:** "Common aspects"
- **Interval**: period to aggregate.
- **Contains**: simulate only those tasks that contain certain text. You can put != at the beginning to indicate to select those records which do not contain the indicated text.
- **Critical**: if the tasks to show are only the critical ones.
- **Numeric or percentage**: if the results must be shown in absolute numbers or in percentage.

- **Team**: if you want to simulate only tasks relative to a specific team.

Efficiency

This report shows the efficiency of the autocontrol management. It is very useful to know how efficient is our maintenance and follow up of procedures.

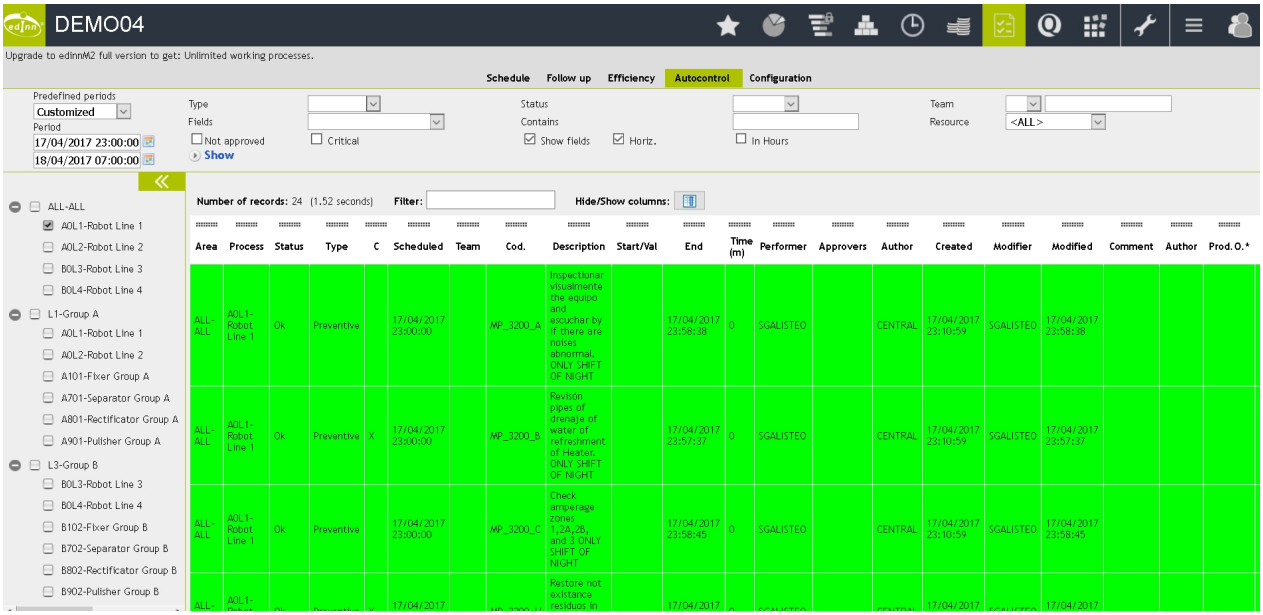


Selectors are:

- **Please see:** "Common aspects"
- **Type**: the type of autocontrol task you need to analyze.
- **Fields**: to show only tasks that contain an specific field.
- **Task**: to analyze only a certain autocontrol task.
- **Critical**: to show only the critical tasks.
- **Interval**: period to aggregate.
- **Contains**: show only those tasks that contain certain text. You can put != at the beginning to indicate to select those records which do not contain the indicated text.
- **Team**: to show only tasks relative to a specific team.
- **Detail**: to show details of results.
- **Summary**: to show the summary of the results.

Autocontrol

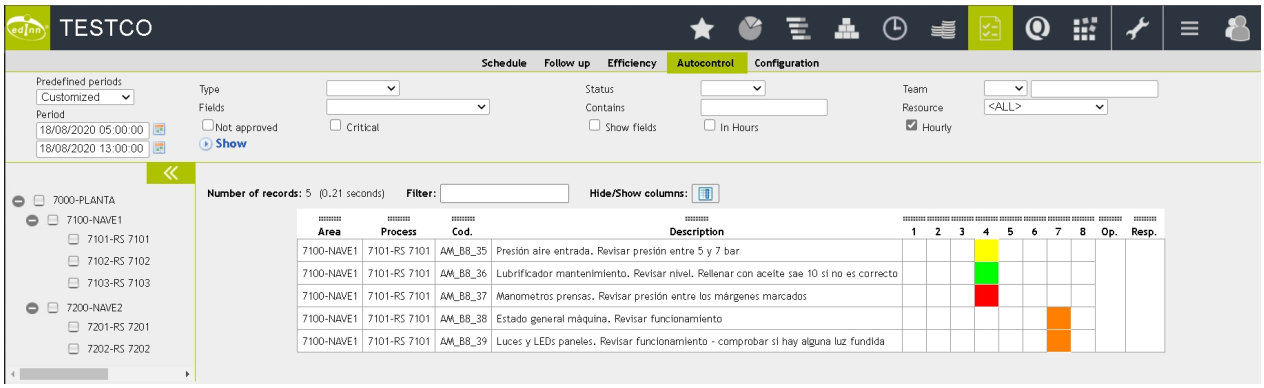
This report shows the most detailed list of the autocontrol tasks. As it shows every record with colors, it is very useful to know if an autocontrol task was not performed, if it had trouble, or if everything was correct. As all the detailed reports, if there are user comments on a record, the comment will be shown at the last column on the right.



Selectors are:

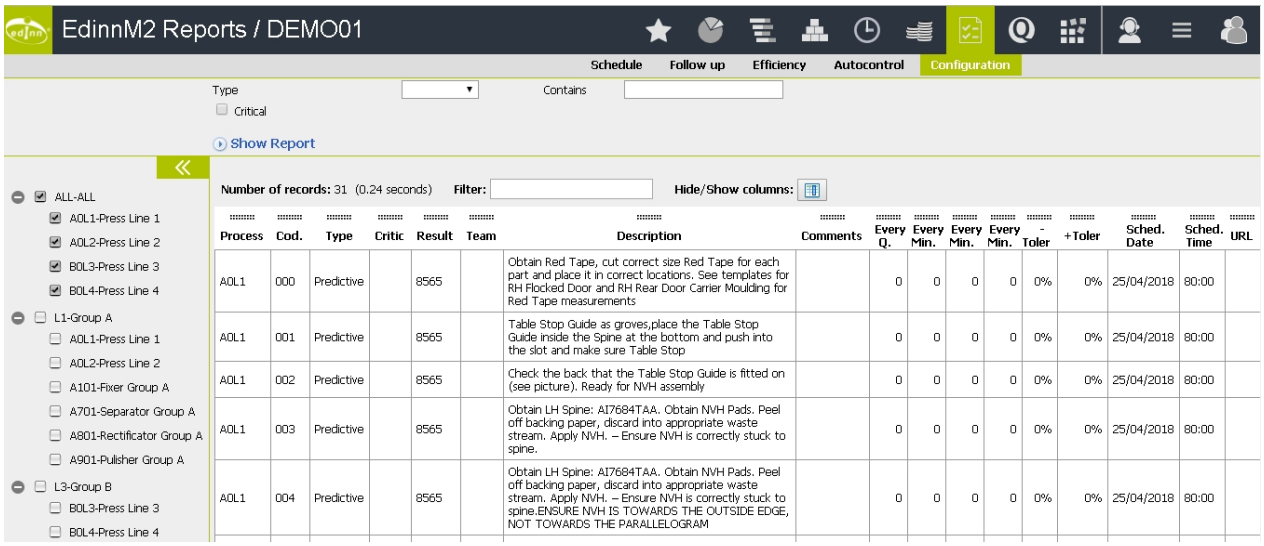
- **Please see:** "Common aspects"
- **Team**: to show only tasks relative to a specific team.
- **Critical**: to show only the critical tasks.
- **Status**: status of the autocontrol tasks.
- **Type**: the type of autocontrol task you need to analyze.
- **Not approved**: to see only not approved tasks.
- **Person**: to see only tasks associated with a certain person.
- **Fields**: to show only tasks that contain an specific field.
- **Contains**: to show only tasks containing certain text. You can put != at the beginning to indicate to select those records which do not contain the indicated text.
- **Only in the period**: will show tasks that where scheduled and finished in the selected period.
- **Show fields**: if fields associated with the tasks must be shown.

- **Horiz.:** to show the register fields on the same line.
- **In hours:** to show the times in hours instead of minutes.
- **Hourly:** to show an hourly table with the status of each task. The colored boxes matches with te finishing, modifying or scheduled hour, depending on the current status.



Configuration

This report shows the configuration of autocontrol. It is useful to know the scheduled tasks and their characteristics. Combined with the Schedule report, the user can have a fine control of the autocontrol module.



Selectors are:

- **Please see:** "Common aspects"
- **Type**: select the type of autocontrol tasks.
- **Contains**: select the text that the tasks must contain in their id, description or comments. You can put != at the beginning to indicate to select those records

which do not contain the indicated text.

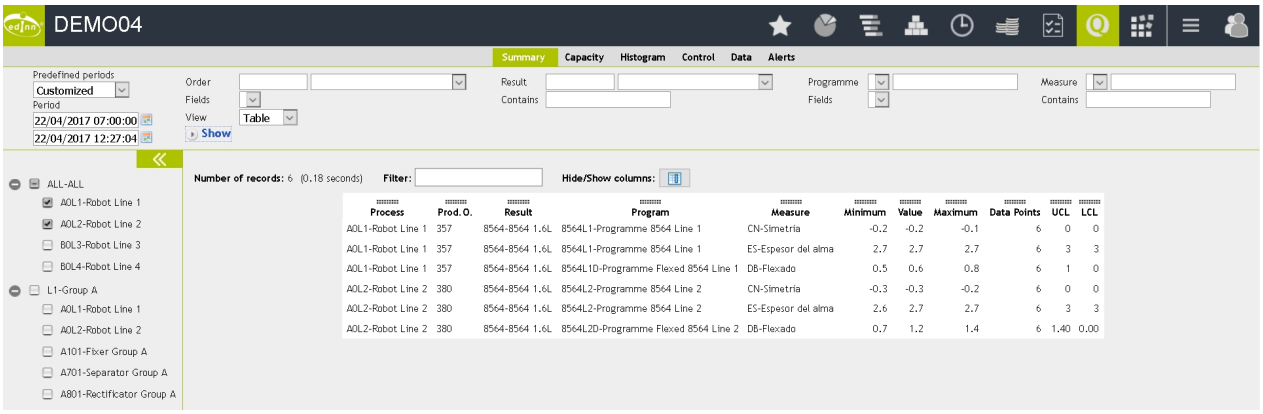
- **Critical**: select if only the critical tasks must be shown.

Quality

This is the section for the reports regarding quantitative quality (SPC).

Summary

This report shows the aggregation of all the quality measures of the processes.

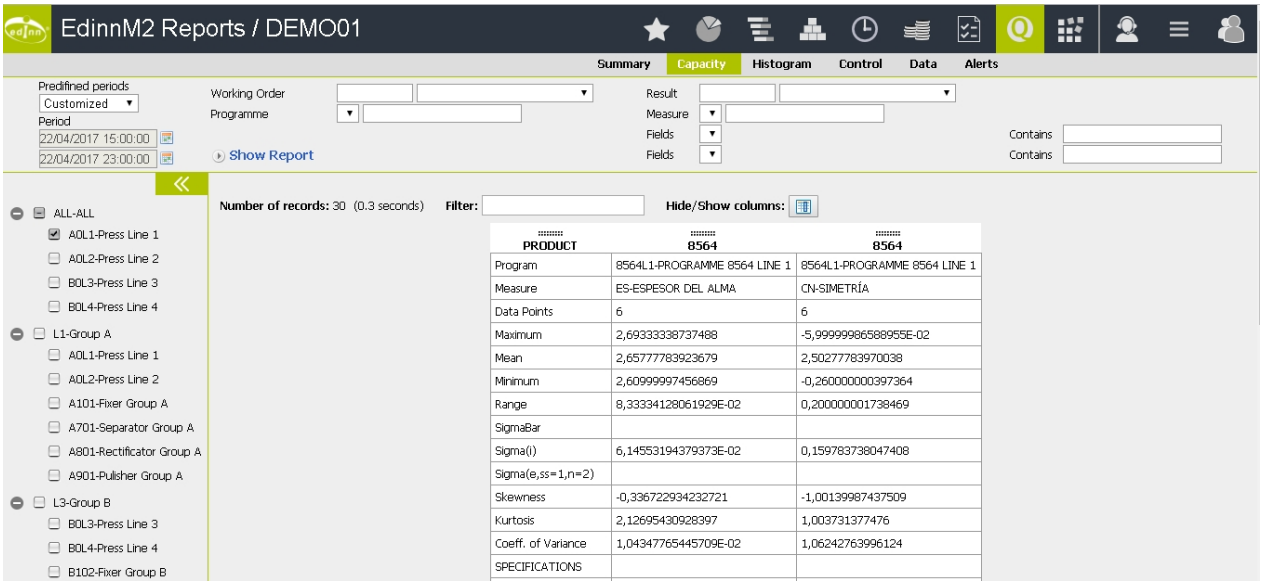


Selectors are:

- **Please see:** "Common aspects"
- **Order**: filter and select the order.
- **Result**: filter and select the result.
- **Programme**: select the programme.
- **Measure**: select the measure.
- **Fields**: select the field you want to filter with.
- **Contains**: type the content of the previous field to filter.
- **View:** select the visualization method. View as a table data or a graphic.

Capacity

This report shows the quantitative quality (SPC) capacity parameters of the process.



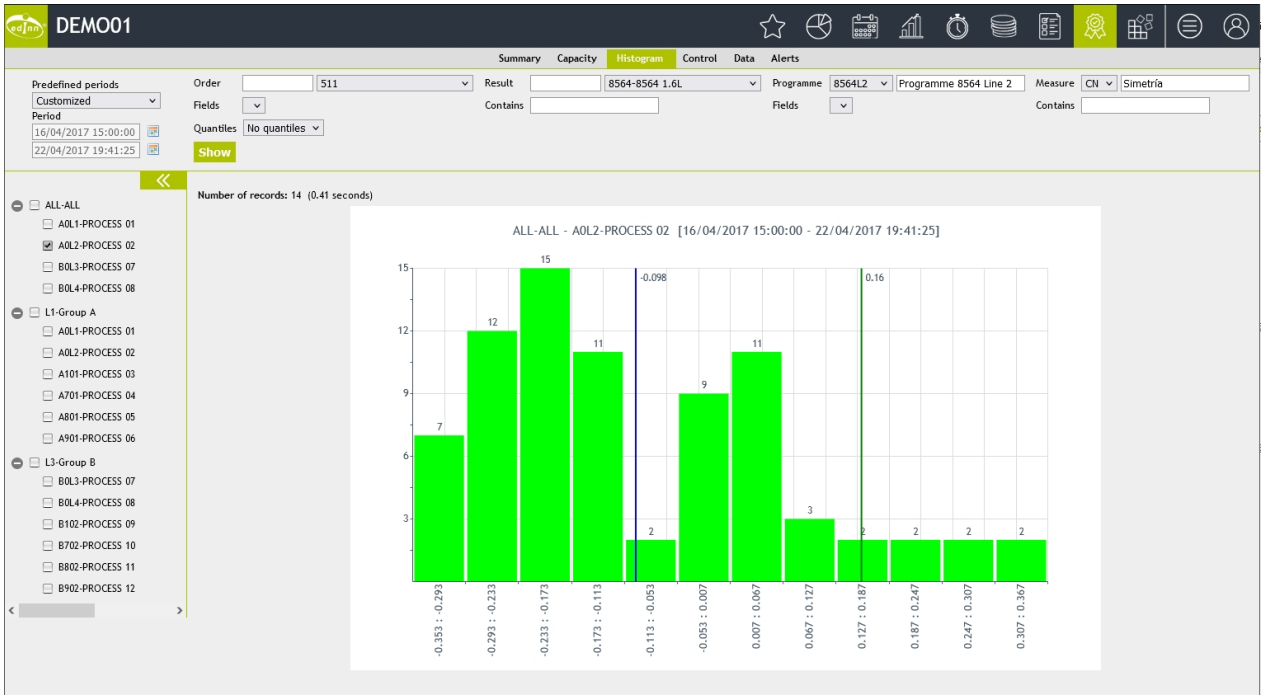
Selectors are:

- **Please see:** "Common aspects"
- **Order**: filter and select the order.
- **Result**: active result.
- **Programme**: select the programme.
- **Measure**: select the measure.
- **Fields**: select the field you want to filter with.
- **Contains**: type the content of the previous field to filter.

Note that the Sigma(i) calculation is performed for capacity related with quality, and not related with performance of a process.

Histogram

This report shows the SPC Histogram graphic from data of the quantitative quality analysis (SPC) and its limits.

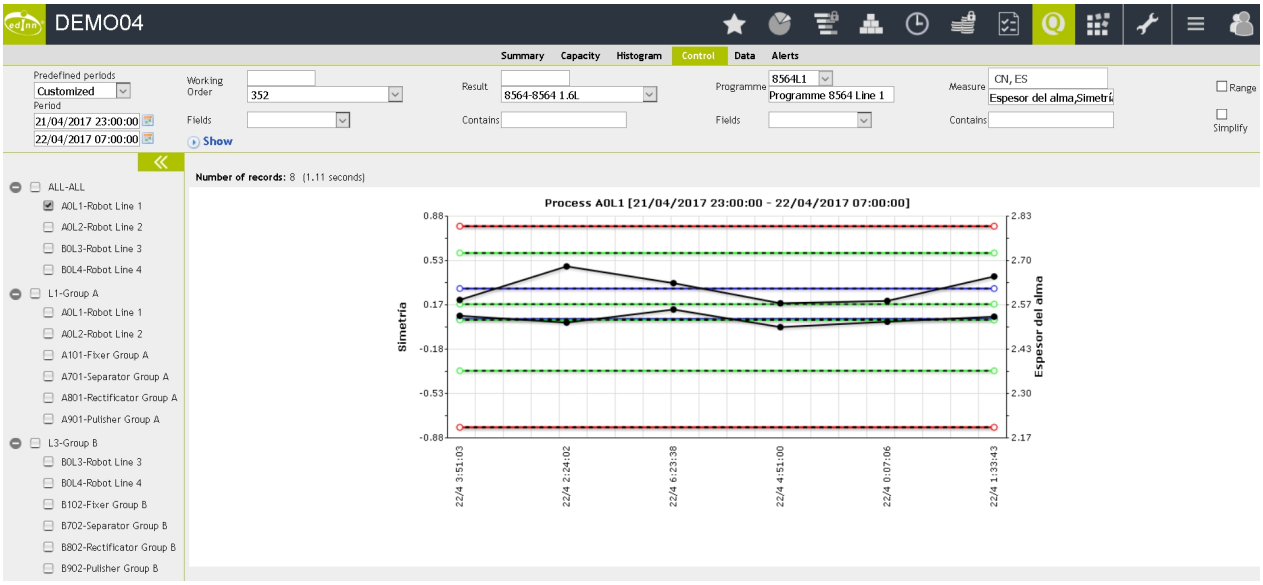


Selectors are:

- **Please see:** "Common aspects"
- **Order**: filter and select the order.
- **Result**: active result.
- **Programme**: select the programme.
- **Measure**: select the measure.
- **Fields**: select the field you want to filter with.
- **Contains**: type the content of the previous field to filter.
- **Quantiles:** allows you to select whether you want to represent the values as quantiles. The options are: Without quantiles (frequency histogram), Quartiles, Quintiles and Deciles.

Control

This report shows the SPC Control graphic from data of the quantitative quality analysis (SPC).

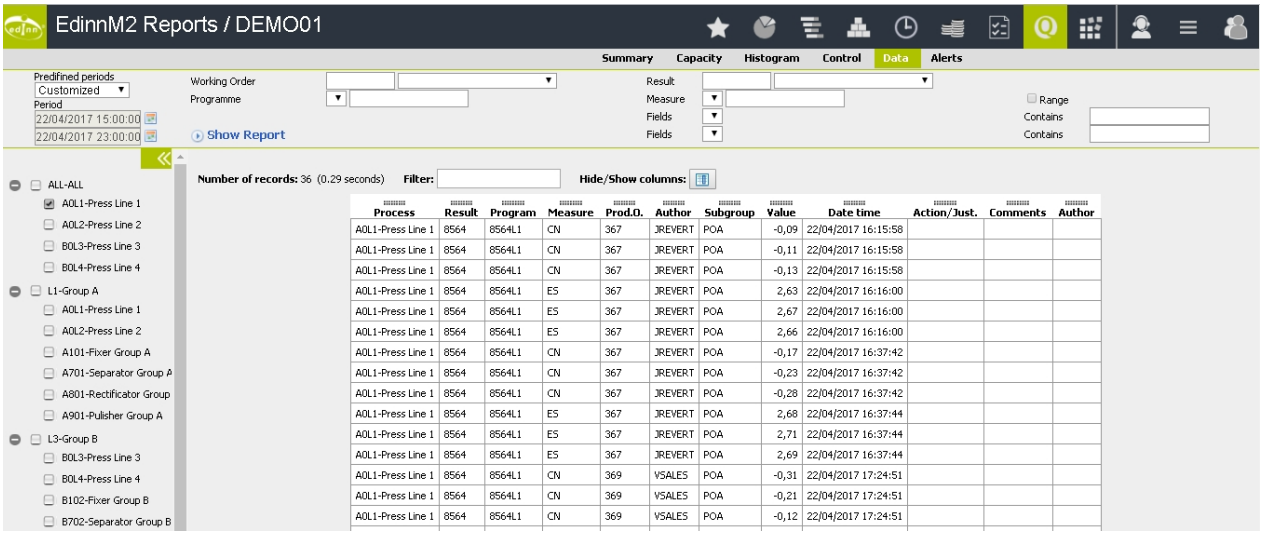


Selectors are:

- **Please see:** "Common aspects"
- **Order**: filter and select the order.
- **Result**: active result.
- **Programme**: select the programme.
- **Measure**: select the measures.
- **Range**: if only the range of data must be shown.
- **Fields**: select the field you want to filter with.
- **Contains**: type the content of the previous field to filter.
- **Simplify**: Show the graphic with the client limits and the real value.

Data

This report shows the detailed data of quantitative quality analysis (SPC).

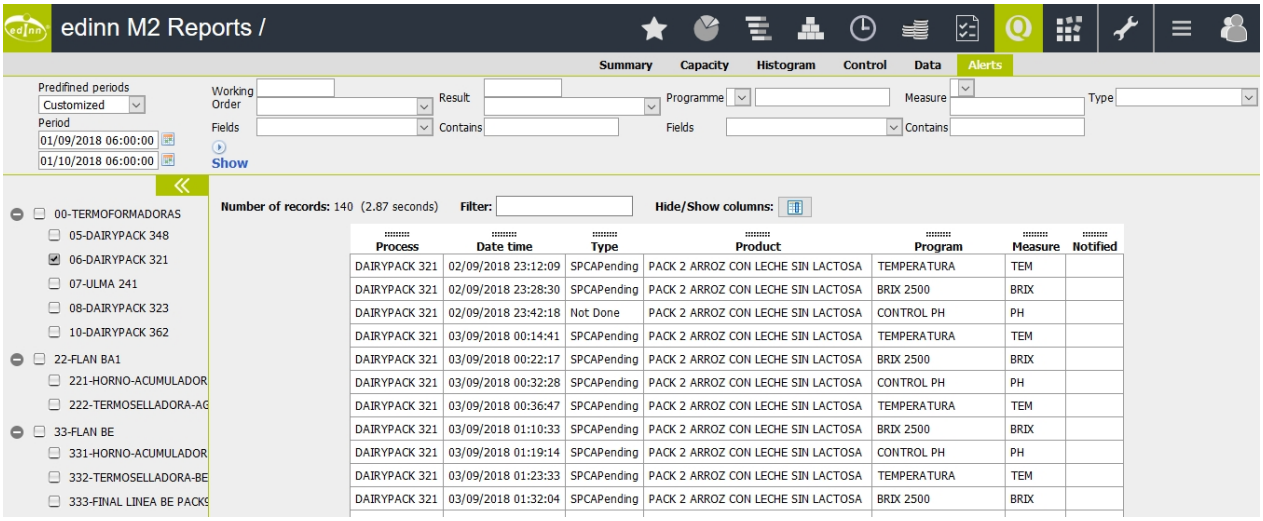


Selectors are:

- **Please see:** "Common aspects"
- **Order**: filter and select the order.
- **Result**: active result.
- **Programme**: select the programme.
- **Range**: if only the range of data must be shown.
- **Measure**: select the measure.
- **Fields**: select the field you want to filter with.
- **Contains**: type the content of the previous field to filter.

Alerts

This report shows the alerts of quantitative quality analysis (SPC).



Selectors are:

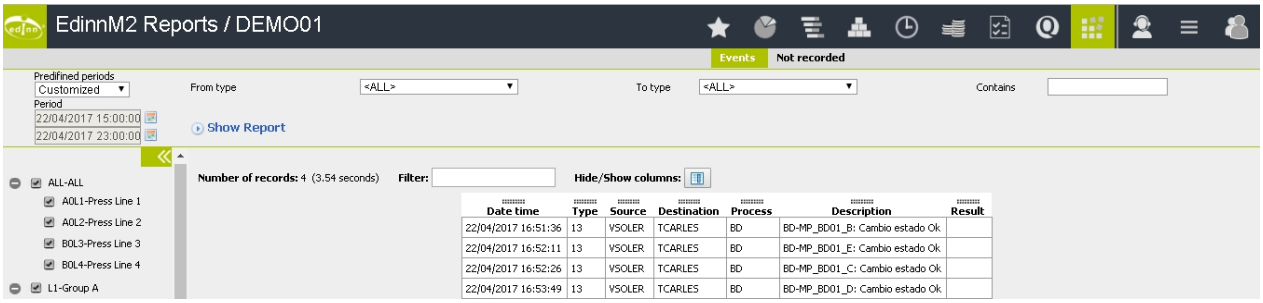
- **Please see:** "Common aspects"
- **Order**: filter and select the order.
- **Result**: active result.
- **Programme**: select the programme.
- **Range**: if only the range of data must be shown.
- **Measure**: select the measure.
- **Type**: of alert.
- **Fields**: select the field you want to filter with.
- **Contains**: type the content of the previous field to filter.

Registry

This is the section for the reports regarding the analysis of certain variables which are more associated with pure control.

Events

This report shows certain events that occur in the system.



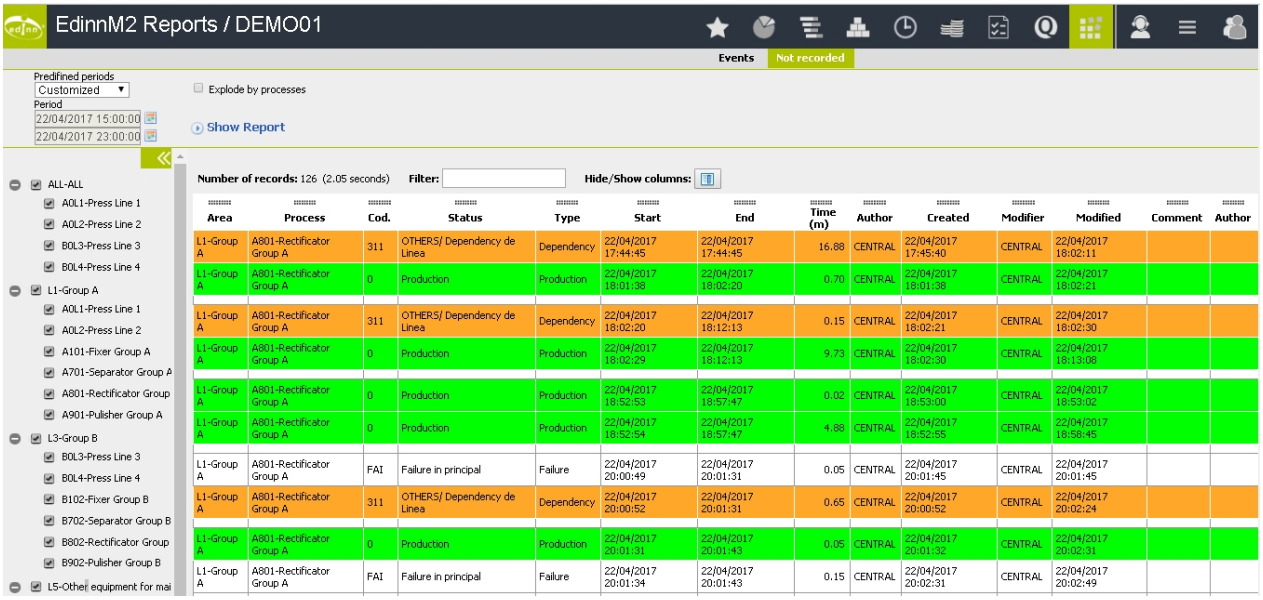
Selectors are:

- **Please see:** "Common aspects"
- **From type to type**: useful to filter and view only certain events. See the events document on this manual for more information.
- **Contains**: filters to only those events which text contains that text. You can put != at the beginning to indicate to select those records which do not contain the indicated text.

Not recorded

This report shows the periods when the system does not have any information of what happened. This can occur when a user with supervisor role deletes information manually or when the system is not capable of monitoring, for example when power line or network are down.

In some reports (for example the Losses Pie) a '**No information**' label will appear. This values must be always zero (0). If they are not zero, keep the selectors (period, process, etc.) and switch to this report and show it. This report will tell what periods must be fixed.



Selectors are:

- **Please see:** "Common aspects"

Integration

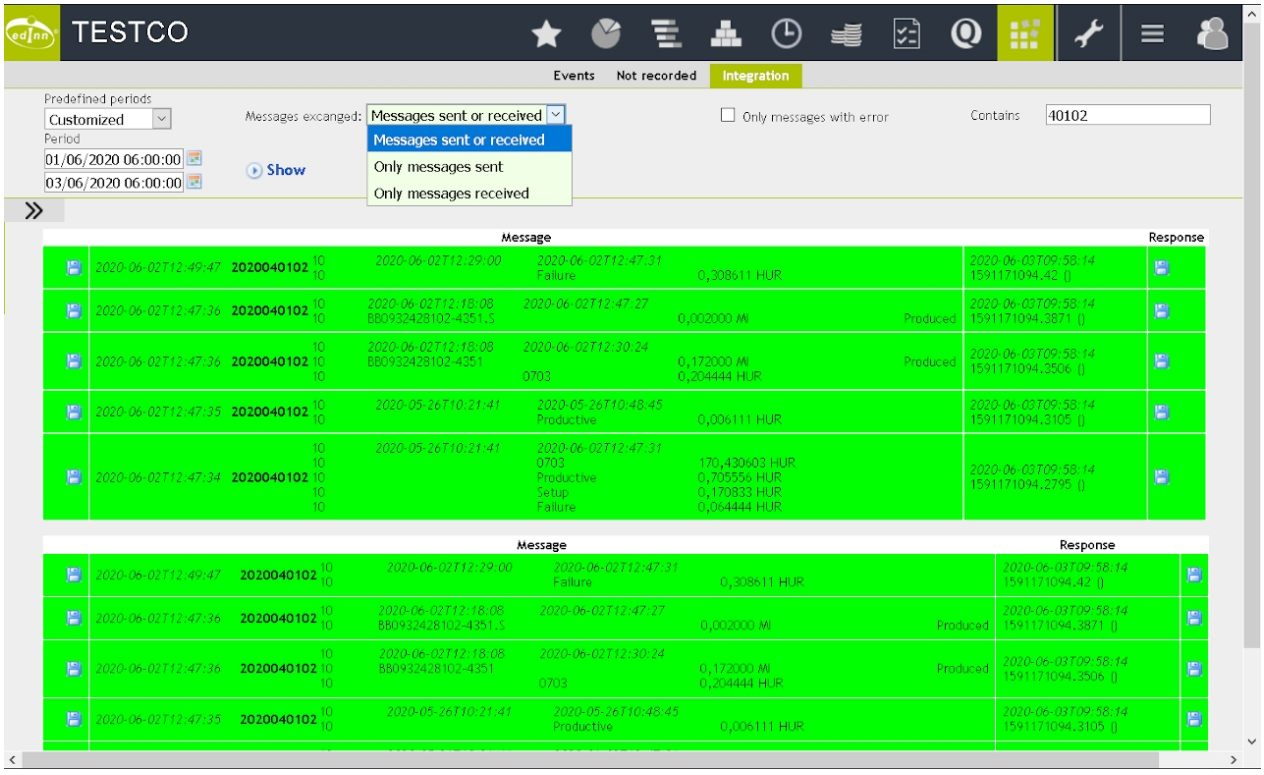
This report shows the summary of the messages that edinn exchanges with other software.

The colors of the records indicate:

- Green: Messages received and processed correctly.
- Orange: Messages with error.
- Gray: Messages that are being sent (only for messages sent from edinn).
- Yellow: Messages returned with a timeout error (only for messages sent from edinn). These messages can be automatically forwarded from edinn. Automatic resending can be configured in the server configuration.

There is a disk icon before the message with which you can show the whole XML message. There is also a disk icon after the response with wihch you can show the whole response XML message.

You can manually resend a returned message with error (oranges) by clicking on the resend icon.



Selectors are:

- **Messages exchanged:**
  - **Messages sent or received:** Shows all messages both sent and reveived.
  - **Only messages sent:** Shows the messages sent from edinn to the connected software.
  - **Only messages received:** Shows the messages sent from the connected software and received by edinn.
- **Only messages with error:** Shows only the messages returned with an error.
- **Contains:** Shows only the messages containig the text specified.
- **Please see:** "Common aspects" for information about the other selectors.

Advanced guide

This advanced guide explain parts of the system which are more complex.

Advanced concepts

Maximum speed

It is very important for the correct calculation of production efficiency (mainly OEE) to correctly configure the maximum production speed (also known as nominal capacity) of the process.

This is because these ratios are calculated on the basis of theoretical maximum production. This theoretical production is calculated by multiplying the time that the process should be working by the nominal capacity or maximum production speed.

Examples:

- In the case of a **machine**: the user's guide of a machine indicates that the machine is capable of producing 50 units per minute. We will introduce 50 as Cycle Unit and 60 seconds as Cycle Time. Remember that cycle time must be in seconds.
- In the case of **persons,** or other more complex **processes**: generally Cycle Unit

will be 1 and Cycle Time will be the time in seconds that the person or process will need to produce 1 unit of result.

It is possible to allow the worker to indicate the results that he will produce and the time that he will need to produce them. For that specific use please contact your technical support.

"Cycle Units" and "Cycle Time" can be configured in different places of the system, and it will try to find them informed (different than zero) following this order. Once the system finds "Cycle Units" and "Cycle Time" informed, it will not continue to read these fields in the next level:

1. The fields of the active Result.
2. In the active work order.
3. For the active process, status and result at Configure Processes-Statuses-Results relations.
4. For the process at Configure Processes.

How to setup maximum Cycle Time and Cycle Units?

What are the fields "Cycle Units" and "Cycle Time"?

- **Cycle Units:** refer to the number of units of results to be obtained in a Cycle Time.
- **Cycle time:** is the time of a cycle to obtain the defined number of Cycle Units.

In other words: maximum speed of production is calculated as Cycle Units and/or Cycle Time.

Speed depending on product parameters:

In some cases, the production cycle time of a product (and therefore, the production speed) may depend on the parameters defined for a given product. This can be parameterized by the user in the More options () button of the Configure Processes-Statuses-Results relations window.

(As this function is complex, please request for more information at your technical support)

Automatic update of the maximum production speed:

This is useful when we do not know exactly the maximum speed of production for a given product in a process.

When the system registers a **higher** production speed than the **configurated** maximum production speed, and this is maintained for more than the specified number of units, it will

update the speed set with the new production speed.

For this, you will have to click in the More options () button of the Configure Processes-Statuses-Results relations window and indicate a number in the **Auto** field. It is convenient to combine these results with the filter of the minimum Cycle Time configured for each process on the **server**.

(As this function is complex, please request more information in the technical support)

Maximum times for status and autochanges

It is possible to configure a **maximum time** for each status, as well as define which will be the "Next" status right after the automatic closing, in the configuration of statuses and in the Processes-Statuses-Result relation.

If a status has a maximum time configured:

- If this status is active and opened (no end time yet), when this maximum time is exceeded, the status will be automatically closed and a new one will be generated. That one will be the default status FAI unless a 'Next' status has been indicated at configuration. Subject to exceptions.
- When a default 'FAI' status is justified to a status that has maximum time, if this maximum time is exceeded, then the status will be justified but closed to the maximum time, and a new default 'FAI' will be generated, which also requires justification. Subject to exceptions.
- When the "Next" status is set in the maximum time setting, when the maximum time is exceeded, instead of creating a 'FAI' status, the defined state will be created. Subject to exceptions.

Particularization through relations

As you can see in the configuration chapter, you can configure (among other things):

- Processes
- Status
- Results

But also, you can relate this items, by the **configuration of relations:**

- Relation Process-Results

- Relation Process-Status-Results

This is **useful** for 2 reasons:

- Users can only select a status or result if it is related to the process they are working with. By this, you can have all the status and results available at your organization, but only make a certain subset available for users for a certain process. For example: if one status is not applicable any more, but you want to keep the historic data, you just remove the relation Process-Status-Results, but you leave the status at the status configuration to be able to see the description, type, etc. of the status when viewing the reporting.
- When the systems needs to obtain any configuration data, it will first search first on the relation and later, if no relation is found, it will search on the configuration master file. This allows, for example, the configuration of specific values for a certain process, when it is in a certain status, producing a certain result.

Envision: failure forecast

The edinn® Envision technology is a tool capable of **predicting failures**, based on a combination of statistical methods and neural networks.

The **Envision module** does not require maintenance and learns automatically, based on historical data. This produces a great cost saving, since it notifies the user through the terminal of possible failures, which can be avoided, in real time.

In order to allow the user to verify the effectiveness of this technology in their industry, the server has a built-in verification function, which can only be used with predictions that are not notified to the user.

ATTENTION: This module works with historical data to perform, combining technical statistics and neural networks, a forecast of the future. It can never be guaranteed that the failure will occur or that all failures that actually occur will be predicted by the module.

Difference between consumptions and inputs

This systems distinguishes between consumptions and inputs.

- **Consumptions:** are consumed in a flow and they can be infinite, like the sun light, or the wind.
- **Inputs:** are consumed individually and can never be infinite. Inputs are the subparts that form a result.

For example: a process that mounts cars, takes the wheels, the chasis and the engine as inputs; and takes money and energy as consumptions.

Read the document about Inputs and Outputs for more information.

How good results are accounted?

The edinn® system count good results in the following order:

1. Registering the total number of results produced in a process, including those that had quality problems and were discarded (scrap) or reworked.
2. Registering the number of results that had quality problems, and the cause for which they were discarded (scrapped) or reworked.

These records do not need to be consecutive. The important thing about this is that these records belong to the same shift, which will allow the system to decrease the production efficiency of the shift.

For example:

Let us imagine we have a process that has produced 1000 results units of light bulbs. And of those, 6 have been discovered to have a production failure and thus have been discarded. At the Results window's grid we should have this:

- A green record showing 1000 results of reference LIGHT BULB.
- A red record showing 6 results of rederence LIGHT BULB WITH QUALITY PROBLEM.

Events

Certain events that occur in the system are registered. They can be managed from the Registry window.

Events from number 100 and higher can be created in configure events.

The predefined events are:

1. **Commentary**: commentaries of users.
2. **Session**: logs in and outs of users.
3. **Status auto change**: auto changes of statuses (time usages).

4. **Password 3 failures**: 3 wrong password attempts of login into the system.
5. **Trace events**: deprecated.
6. **Team auto change**: automatic changes of teams.
7. **Monitor down**: indicates when the edinn monitoring service was down.
8. **SPC justification**: justifications of SPC.
9. **SPC no data**: indicates where no data of SPC was taken and should have been taken.
10. **SPC alerts**: indicates alerts of SPC.
11. **SPC context**: contains context data of SPC.
12. **Author change**: deprecated.
13. **Item desc.**: item description.
14. **On/Off change**: indicates when an On/Off signal has been sent by a user.
15. **Interface error**: there has been an error with the integrated interface.
16. **Audit of manual deletions**: indicates the records that have been manually deleted by users.
17. **Subscription**: events where a user was informed of the stops of a process he was subscribed to.
18. **Favorites deletion**: informs a user that another user with permission to delete has deleted a favorite report.
19. **Maximum time**: a status changed automatically to another status because its maximum time was reached.
20. **Recommendation**: automatic recommendation sent to a user.
21. **Congratulation**: automatic congratulation sent to a user.
22. **Improve**: automatic guidance for improvement sent to a user.

Difference between production lines and areas

Processes that are grouped in production areas, as we can see in Areas Configuration, do not have any additional restriction. Areas are only a visual group and these groups can be changed anytime without any negative consequences.

However, when processes are linked through the field '**Next**' as shown in process configuration, it is indicating which processes follow the current process in the production

process. In this way, a production line is defined which is different from the visually configured area in areas configuration. By using the field '**Next**', the configured line has the following restrictions or functions:

- All processes of this line are always producing the same result.
- If we manually introduce a status to a process within the line, we are allowed to spread this status to all processes of the line (if they are marked to propagate at the PSR relations configuration) or we can simply transfer it as a dependency.

Things we need to set up a demo

To set up a demo of edinn® M2 including **sensorization**, we need these things:

1. **An interested client**:

   In addition to allowing the use of the facilities for testing, it is essential that the client agrees to cooperate and provide feedback to carry out the demo successfully.

2. **A process**:

   It can be fully manual (just people), fully automated (just a machine) or a mix. For a quick test it is easier with automated processes. Almost every machine of every industry can be monitorized with edinn® M2. In order to show the client the potential of edinn® M2, it is better if the selected process is causing efficiency problems like low productivity, uncertainty or excessive consumption.

3. **An edinn monitoring BOX or eBOX**:

   Monitoring hardware that gathers the data and sends it to a central edinn server.

4. **A production and optionally a consumption signal**:

   You can practically control all the outputs and inputs of any process, but first time tests, we typically monitor only 2 things, 1 production and 1 consumption (electricity)

   By this, we can quickly see the powerful effect of combining production and energy efficiency. The sensors come with the eBOX.

5. **The client should indicate a point in the process to attach a production sensor and optionally an electrical wire to attach the consumption sensor**:

   In order to sensorize the process, we will need to attach to it 1 piece-pass sensor, which will detect every product that is produced; and optionally 1 open toroid which will be installed surrounding one wire of one electrical phase of the process.

   It is important that the production sensor measures only real and total products.

You should discard points where some products do not pass or points that may indicate that product is passing, but it is not.

6. **A time of 2 to 4 h. when the process is stopped**:

   This sensors have to be installed when the process is totally powered off, so a moment has to be found for that. Total installation takes typically 2 h. Installing the production sensor is just attaching a little "mirror" on one side of where the product passes by and the light generator and photocell on the other side.

   Then the sensor is directly connected to the standard connector of the eBOX. The installation of the consumption sensor is just opening the electrical box of the process, locating the wire of one of the electrical phases and attaching an open toroid surrounding it. Then the sensor is directly connected to the standard connector of the eBOX. The eBOX is designed so that the user can not connect a sensor to the wrong connector.

7. **This does not affect the process**:

   The production sensor is a light barrier, so it does not affect the process. The consumption sensor is an open-toroid wich is closed surrounding the wire, so it does not affect the process neither. There is no need to harm the wire or its plastic cover at all.

8. **An authorized installer**:

   For demo installations, we can do it ourselves, specially because we are using an eBOX. But for commercial installations, we recommend to always work with our authorized partners which have the experience, the knowledge and the needed certifications.

Notification priorities

To notify of an event, the system follows a sequence of priorities, depending whether the systems has the email of the responsibe of:

- First, the team.
- Second, the shift.
- Finaly, the process.

Any notification will be sent if the emails of the corresponding responsibles has not been specified in the system.

Quality Validations

This section describes the available Quality validations.

RD1801/2008

This document describes how the the spanish quality norm RD 1801/2008 is supported in the system.

If you requiere any other international norm, certification or quality validation, please request a quotation to support.

For a better understanding, we recommend to have available the official document from the spanish Government "BOE número 266 de martes 4 noviembre de 2008 (43707)".

If this validation is activated, it will affect to the following functions:

- In the introduction of quality measurements in the quality window.
- When selecting another value in the quality window.
- When creating or changing and saving the value of the field, both in the results and autocontrol window.
- When changing the result in the results window.

In all these cases, the system:

1. Requires a number of measurements according to the size of the lot:
    - If less than 100: all the units of the lot.
    - Between 100 and 500: 30 units of the lot.
    - Between 501 and 3200: 50 units of the lot.
    - 3201 or more: 80 units of then lot.
2. Checks, according to what is indicated in the norm, if the measurements are not defectuous which means that they are inside the client limits.
3. **IMPORTANT: For the alerts to be shown in the terminal, the process must be configured with Trace to the level of devices.**

Inputs and Outputs

Inputs and Outputs (I/O) work as follow:

- When using schedule:
    - When creating an order:
        - If there are I/O relations in Results configuration (Template), then the target quantity of this relations expresses the number of units of I/O for each produced unit, and the target of I/O is calculated as follows: Quantity of I/O of Results configuration (Template) * Target Quantity of the Order. The I/O can be modified manually.
        - If there are no I/O relations in Results configuration (Template): I/O will not be automatically generated in the order. The target of E/S will be assigned manually.
    - When starting an order, it will inherit to the Result the I/O of the order. The quantity will become 0 and the target quantity will be the equivalent to 1 result produced.
    - When generating results, the target quantity and the quantity will be identical and calculated like this: Target Quantity of I/O of the order * Results Quantity / Target Quantity of the order.
- Without using Schedule:
    - If there are I/O relations in Results configuration (Template), the quantity expresses the number of units of I/O per produced unit.
        - When changing the result, the I/O quantity sets to 0 and the target quantity will set to the equivalent of 1 produced result.
        - When generating results, the target quantity and the quantity will be identical and calculated as follows: Target Quantity of I/O * Results Quantity.
    - Without I/O relations in Results configuration (Template):
        - When changing result, I/O records will not be generated.
        - When generating results, I/O records will not be generated.
- When in the last record of results, I/O records are added, modified or deleted, those changes are not inherited to the following records of results.
- Scrap and rework:
    - When generating scrap and rework, the target quantity and the quantity will be identical and calculated as follows: Target Quantity of I/O * Results Quantity.
    - If the “Auto insert bad results” option is activated in the console, when initiating an order, it will insert the equivalent result of scrap and rework. In this case, it

will insert as inputs and outputs of these bad results what is in result settings for those references.

Configuration

How to start

There are 3 points where you can configure the platform:

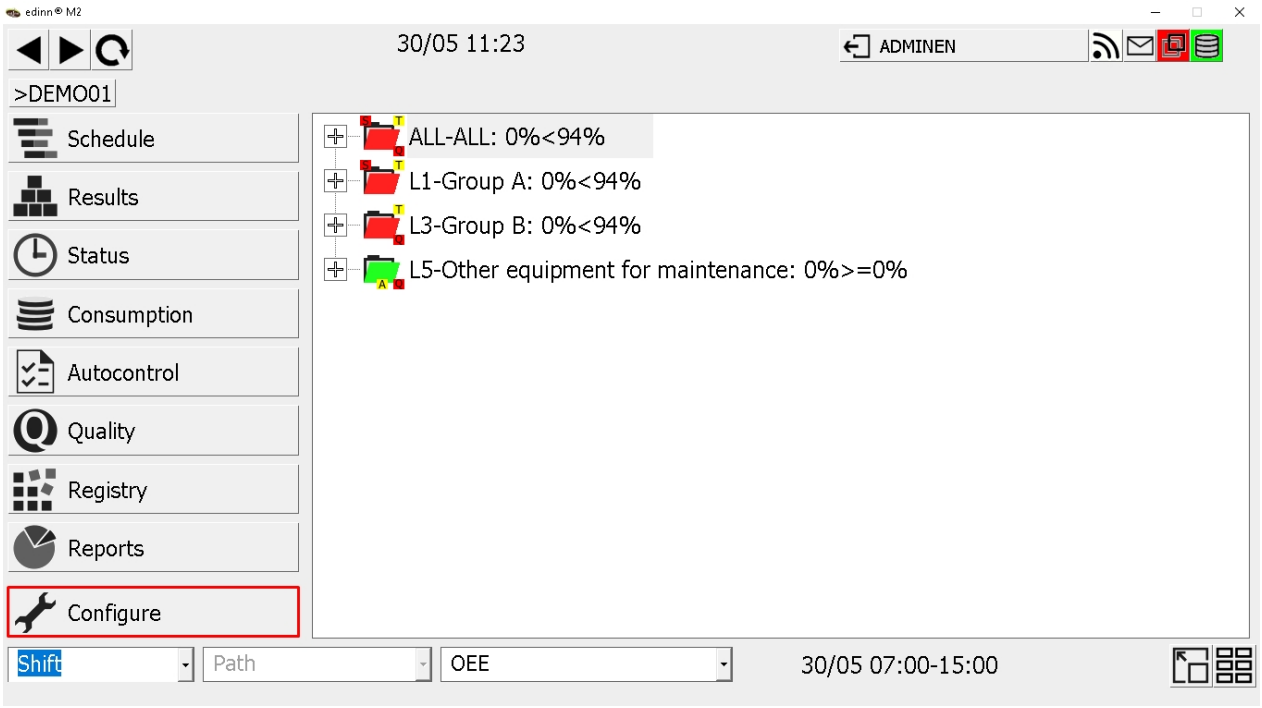
- The web wizard, for a basic and initial configuration.
- The terminal, for a detailed configuration but without including server subjects (monitoring, backup copies, etc.). This configuration is explained here in the following pages.
- The server.

To access the **terminal** and start the configuration, follow these steps:

Step 1. Start terminal and login.

Step 2. Access the configuration window

Press  to access the Configuration window:



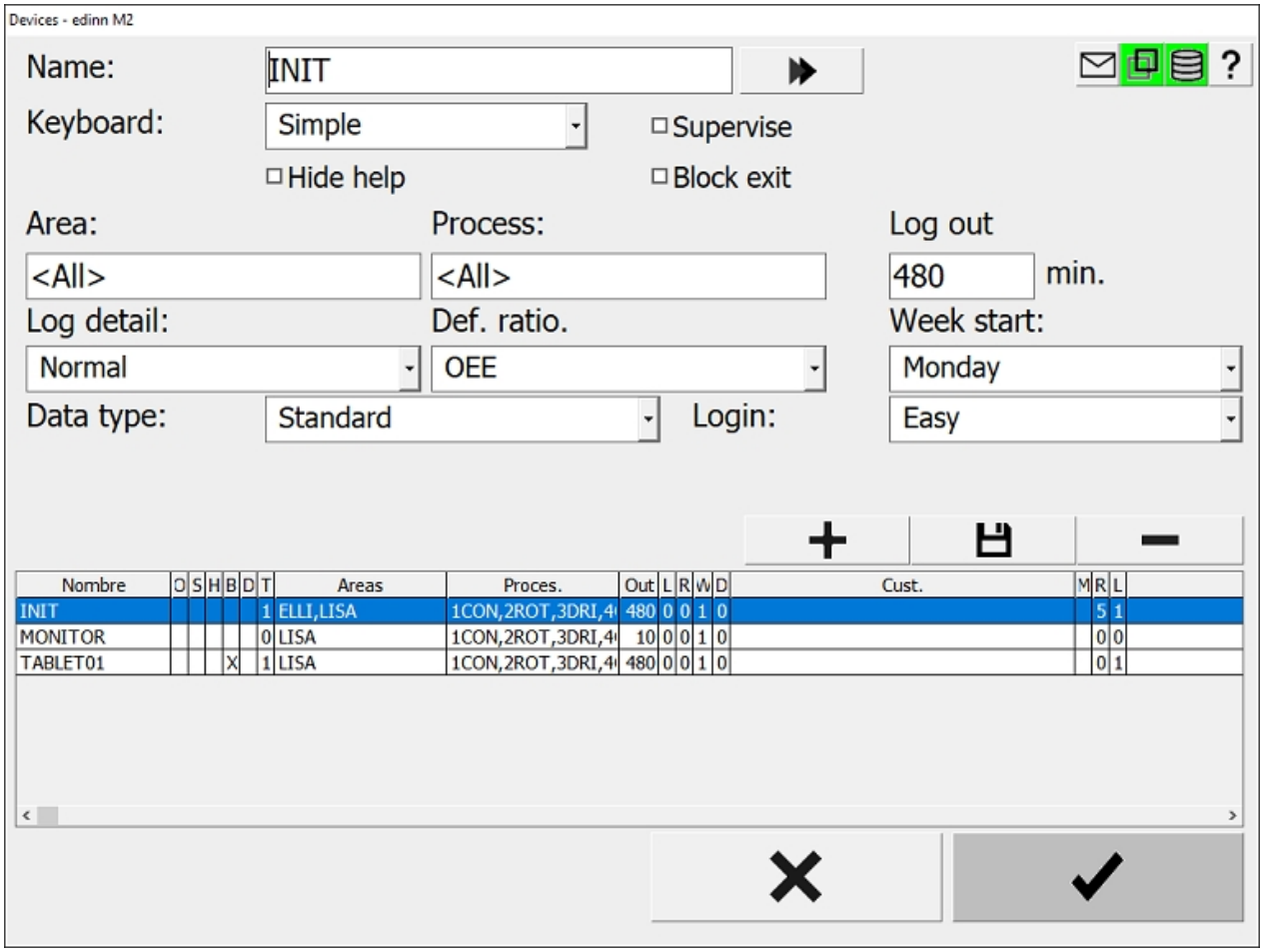
Devices

Click to configure > Devices

A **device** is a terminal or equipment that allows physical access to the system. Examples of devices are: *touch tablets, touch terminals, personal computers, sensors, etc*.

In the lower part: you will see a grid with the devices already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

In the upper part: you will see the configuration of each record, where you can configure the parameters of a new insertion or edition.

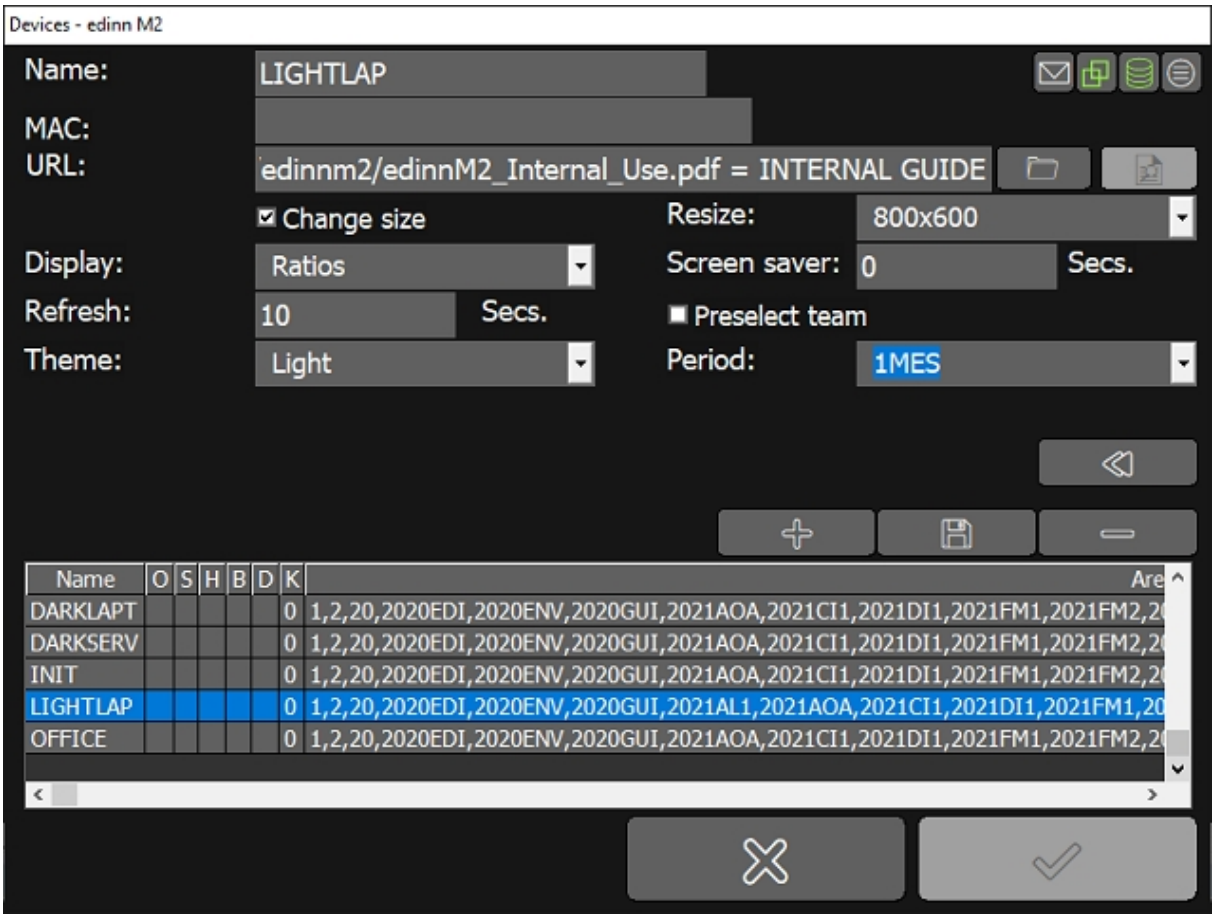


CONFIGURATION PARAMETERS:

- **Name:** name of the device.
- **Keyboard:** to indicate the keyboard configuration (none, simple or multilanguage)
- **Supervise**: if the device should be supervised by the **Supervisor Service.** It will send email alerts if the device is disconnected.
- **Hide help**: activate if you want the contextual help icon (?) to not appear.
- **Block exit**: does not allow a user to end the application.
- **Area**: working area to which the device (terminal) is associated. The terminal will only be able to see the areas indicated here. Leave **blank** to see all areas.
- **Process**: process to which the device (terminal) is associated. The terminal will only be able to see the processes indicated here. Leave **blank** to see all processes.
- **Log out**: every x (minutes), of inactivity of the user, the device should automatically end the user's session.
- **Log detail:** how many log trace records will generate for debugging. Usually has

no effect, as this field is only used by edinn system engineers. Choose between: *Normal, Detail and High detail*.

- **Def. ratio**: ratio shown by default on the main window. Choose between: *OEE, PE, OCE, OPCE, MTBF*.
- **Week start**: indicates at what day the week starts.
- **Data type**: type of data seen at "Data" at the main window. Choose between: *Standard, Custom*.
  - **Custom:** when choosed, a blank space will appear, where particularization sequence should be introduced. To get to know the particularization options please consult the *Particularization of data grid*.
- **Login**: type of entrance or login to the system. Choose between: *Standard* (indicated for devices with physical keyboard) *or Easy* (indicated for touch terminals).
-  will show more options:



  - **MAC**: the MAC of the network card that connects this device to the system. By using this MAC, activating the option Supervision (see above) and if the device supports the Wake On Lan function, the system can automatically power on the device. Please contact your technical support

service for more information.

- **URL**: a URL, following the URL syntax of the system, that can be launched by the user from the login window. URLs can be selected from  and its syntax can be cheched from the button  .
- **Change size**: allows the user to change the size and to minimize the application.
- **Resize**: indicates how to rescale the client application in the device. Choose between: *Max (full size screen), Min (240x480), 640x480, 800x600,1024x768*.
- **Display:** to select the **type of visualization that the device will show by default.** Choose between: Path, Speed, Consumption, Path&Consumption, CO2e, Cost, Ratios, Data, Control, Histogram, MTBF, MTBFP, In&Out, Team In&Out, Dashboard (customized with the API) and Dashboards Web. In the last option, this device will have available, in the main window, the dashboards of the reports that have been assigned to this device from title tab of the dashboard on the web reports or from the processes configuration.
- **Screensaver ***: number of seconds of inactivity in the login screen that the system will wait to activate the screen saver, which consists in the previously selected dashboard, full screen. Type 0 to disactivate.
- **Refresh**: number of seconds that the screen saver will wait to show the next process associated to this device and refresh the dashboard.
- **Preselect team**: indicates if the team of the user should be preselected in the team drop down lists.
- **Theme**: select the default color theme.
- **Period**: select the period that the main window will show by default for this device.

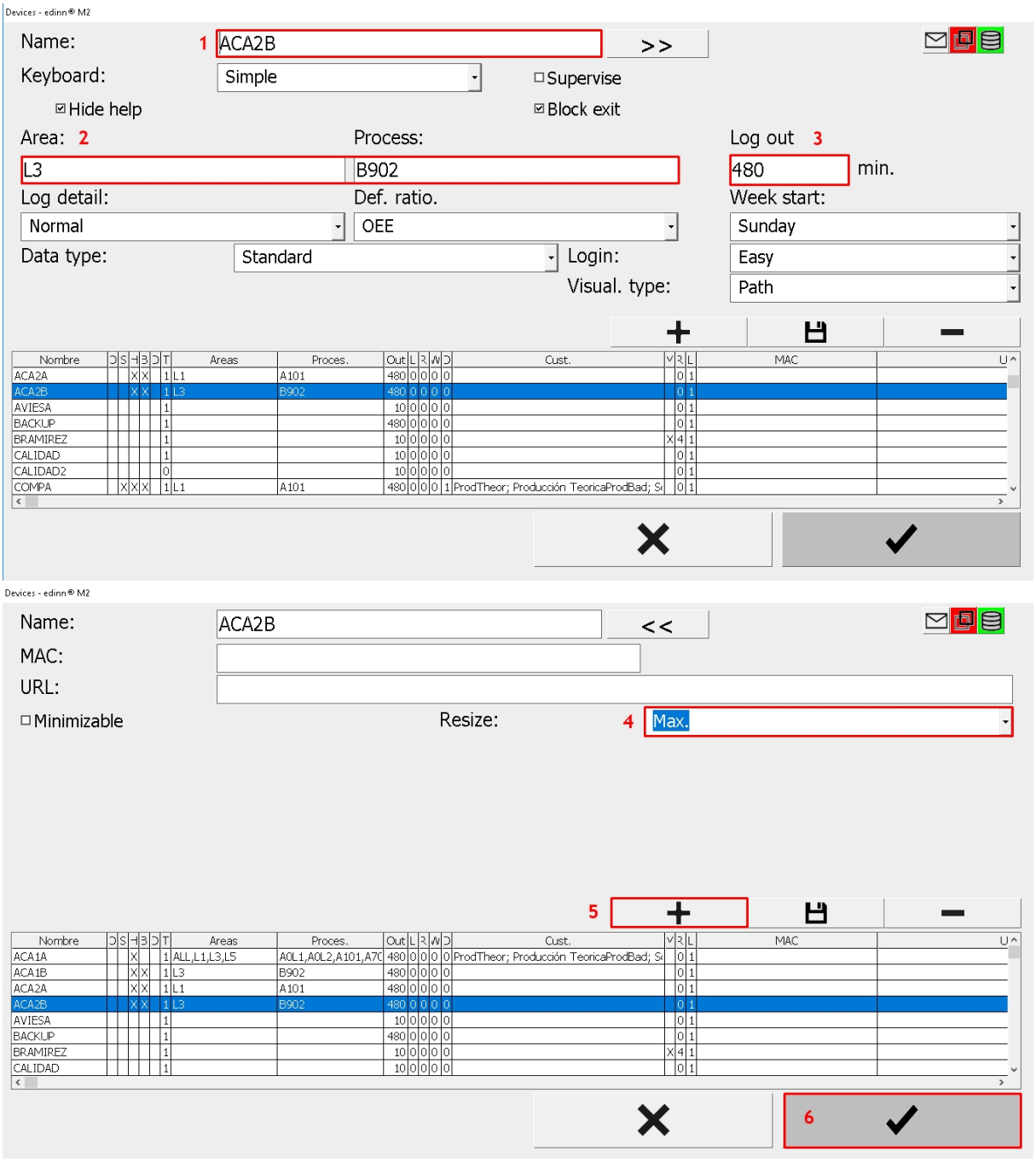
* The screensaver requires:

- In the terminal which is going to show it: Microsoft Internet Explorer, in their most updated versions (version 11.4XX or higher).
- A valid user to cache reports, configured as a resource, and configured in the Console, Reports tab, Cache of favorites section fields Username and Password. If you change these fields, you need to restart the services and, after, the edinn terminal.

How to configure a device step by step

Once the General Options have been defined, we will configure the basic characteristics of each Terminal. In **[Click to configure]** select **[Devices]**.

Follow these steps for **each Terminal** to be configured (3 terminals = 3 configurations):

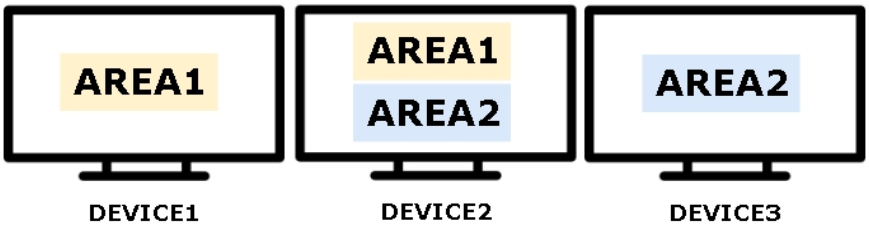


Step 1. Configuration for each device

1. **Name:** write the name or the identifier of the Terminal to be configured, previously defined in General Options.

2. **Area & processes:** each device can have a series of areas and processes visible, to limit these permissions, write in the field the areas and processes that you want to be displayed, separated by commas (e.g: Area: L1,L2,L3. Process:

B902,A101.)



3. **Log out:** to prevent a user from using another user's session, you can set an inactivity time, in minutes, after which the session will automatically close.

4. **Resize:** press and indicate the default size of the Terminal window in [**Resize**]. Normally we will use Max for the plant terminals.

NOTE: this is the common configuration, the rest of fields are generally defined by default, for more information, please go to: Configuration devices.

Step 2. Save changes

5. Press to insert the configuration of the device in the table. Repeat this action for **each** available device or terminal.

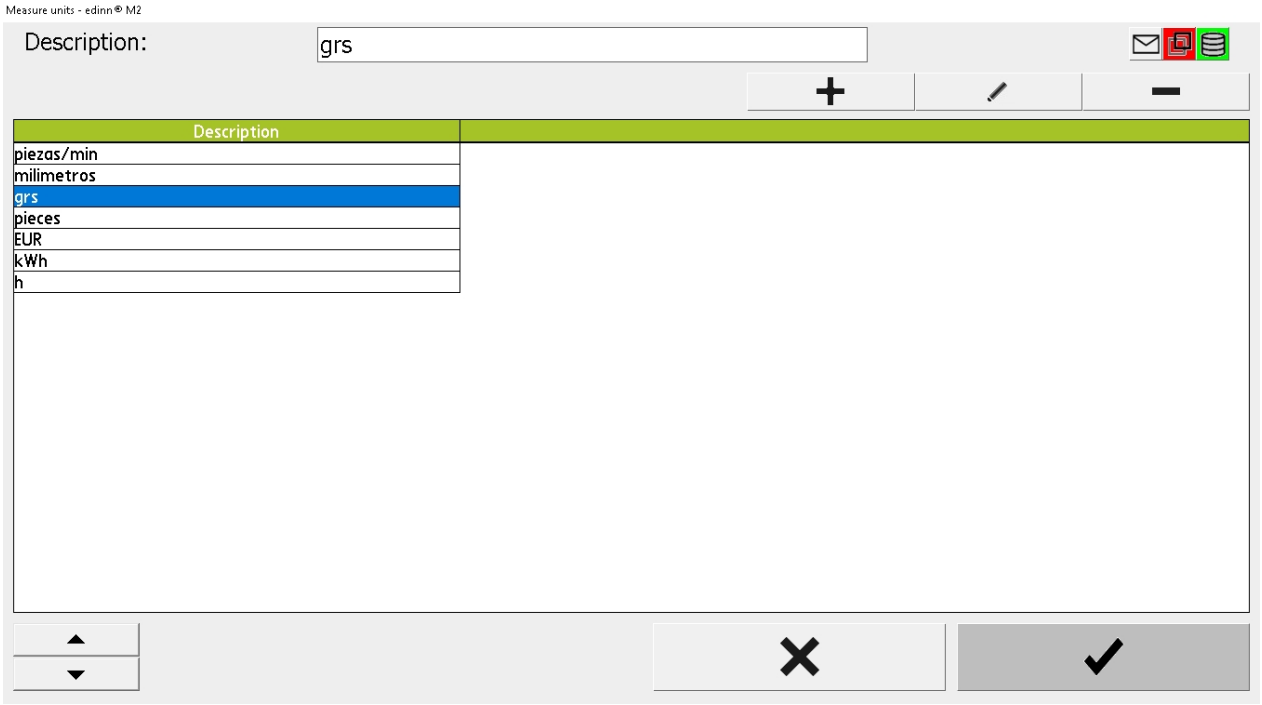
6. Press to make all the changes effective.

Measure Units

Click to configure > Measure units

Measure Units are those units in which production, consumption, etc. can be measured.

You will see a grid with the measurement units already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

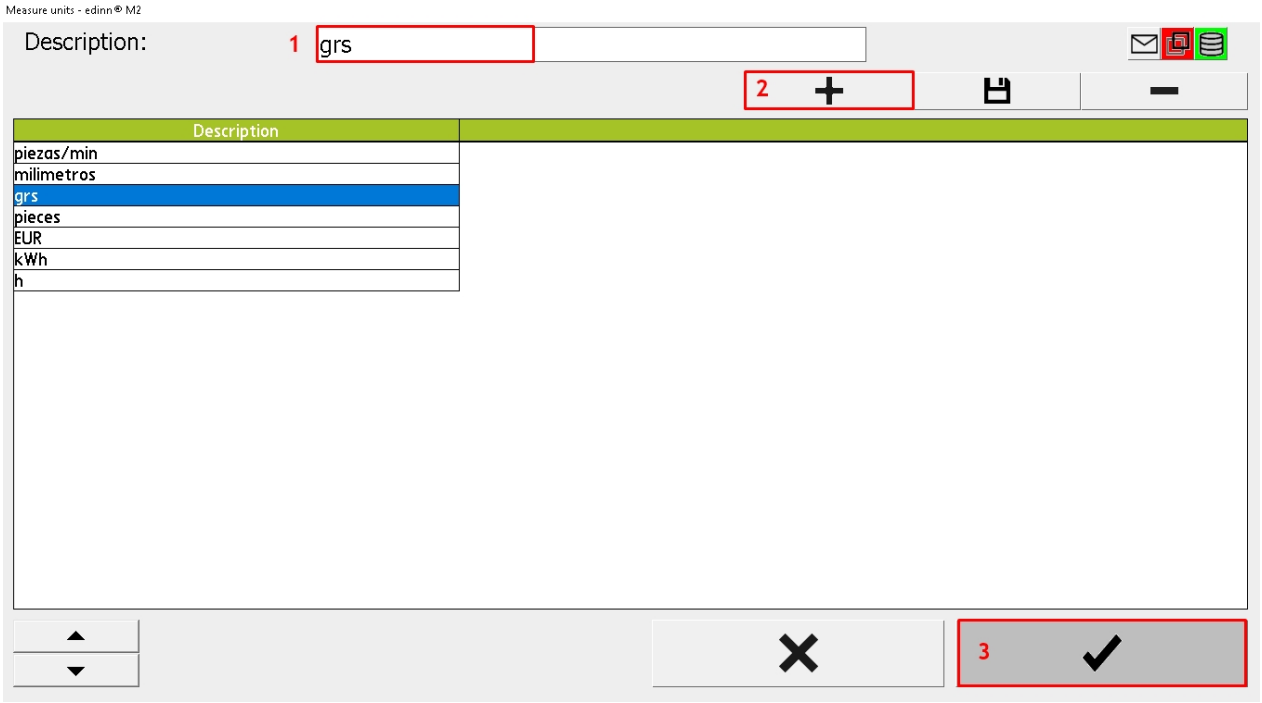


As in the other windows, the **up and down arrows**, allow to change the order of appearance of the selected element.

How to configure a measure unit step by step

Next, a list of units is configured, which will be used as reference in other configurations. In [**Click to configure**] select [**Measure units**].

1. **Description:** enter the name of the unit of measure to configure (e.g: *kWh*, *Wh*, *Euros*...)
2. Press ✚ to insert the unit of measurement in the table. Repeat this action for **each** unit of measure to be created.
3. Press ✓ to make all the changes effective.

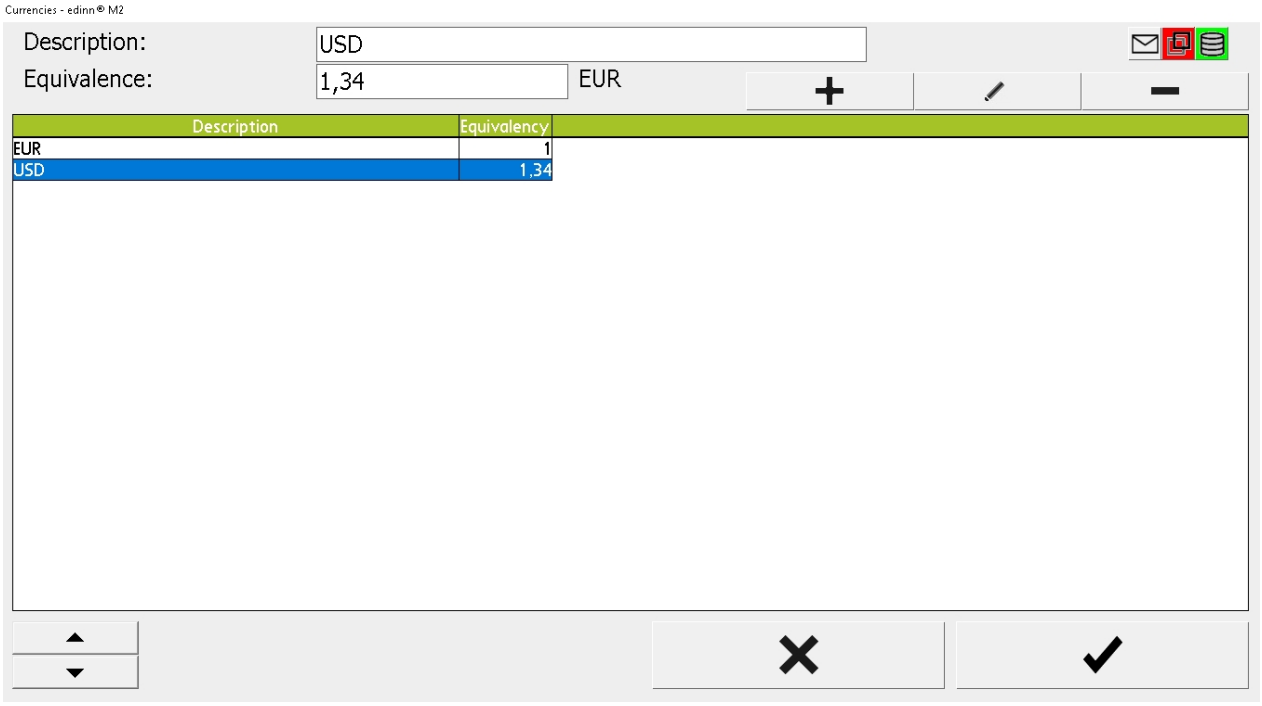


Currencies

Click to configure > Currencies

This window allows configuration of the different **currencies** that need to be configured in *consumption types, costs, etc.* and its equivalence in euros (or other base currency).

You will see a grid with the currencies already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

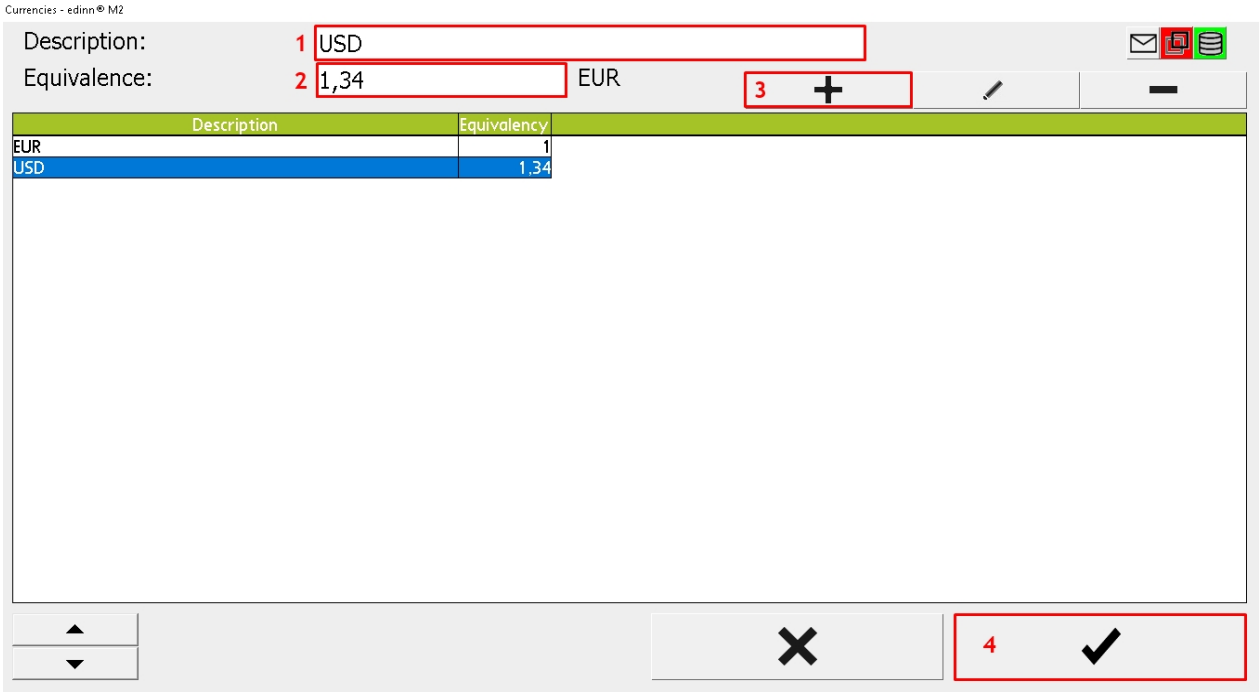


As in the other windows, the up and down arrows, allow to change the order of appearance of the selected element.

How to configure a currency step by step

Configure a list of currencies, that will be used in the **Cost module**, and establish an equivalence between them. In [**Click to configure**] select [**Currencies**].

1. **Description:** name of the currency.
2. **Equivalence:** equivalence between a currency and the currency defined as a reference (equivalence = 1)
3. Press ✚ to insert the unit of measurement in the table. Repeat this action for **each** unit of measure to be created.
4. Press ✓ to make all the changes effective.



Consumption types

Click to configure > Consumption types

This window allows configuration of all the consumption types that the system will consider. Please notice the difference between consumptions and inputs.

In the lower part: you will see a grid with the consumption types already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

In the upper part: you will see the configuration of each record, where you can configure the parameters of a new insertion or edition.

CONFIGURATION PARAMETERS:

- **Code**: code of the consumption type. For example: ELEC1, COST2...
- **Description**: description of the consumption type.
- **Equivalence**: numerical factor that will multiply the quantity read from monitoring sensors. For example: if the power meters provides Watts and we need to analyze KWs, we will multiply the number obtained by the monitorization sensor by 1/1000. We will put 0,001 in this field.
- **Units**: the unit of the consumption type. The options shown will be the measurement units already configured.
- **Cost check**: activate it if the consumption type is an economical cost. For examples: wages, salaries, etc.

- **Incentive check:** activate it if the consumption type is used as an incentive. Generally used with employees, where an incentive will be calculated based on the activity ratio and the activity objective.
- **Auto check:** consumptions can be associated with employees, processes and results. If you associate a specific consumption to each one of them in the button **[Theor. Cons]** of the windows: "Resources", "Process-Status-Result", "Process" and "Results", a consumption will be automatically calculated based on the time spent. For example: if an employee has associated consumption of x€ per hour, each hour spent working on a process will automatically calculate the consumption of that employee.

NOTE: for more information, please see: Theoretical Consumption

- **Currency**: the currency that will be associated with this consumption. The options shown will be the currencies already configured.

Resources

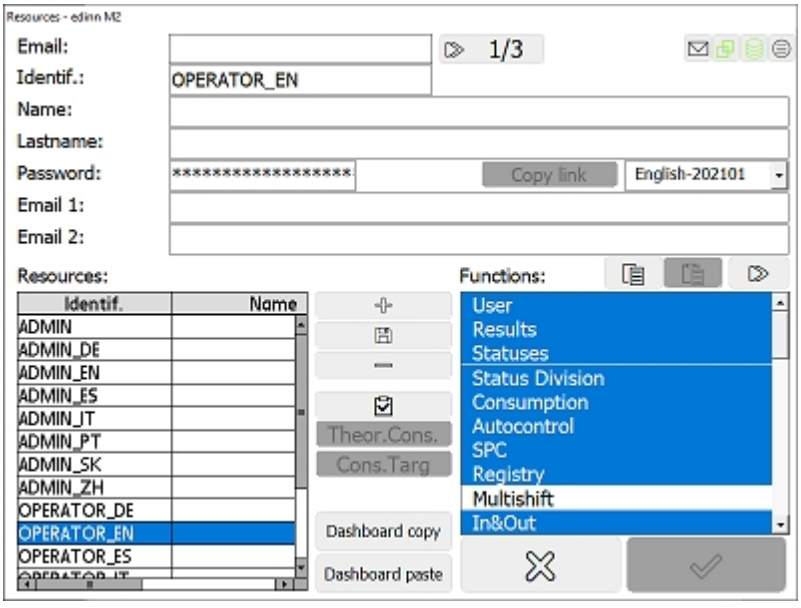
Click to configure > Resources

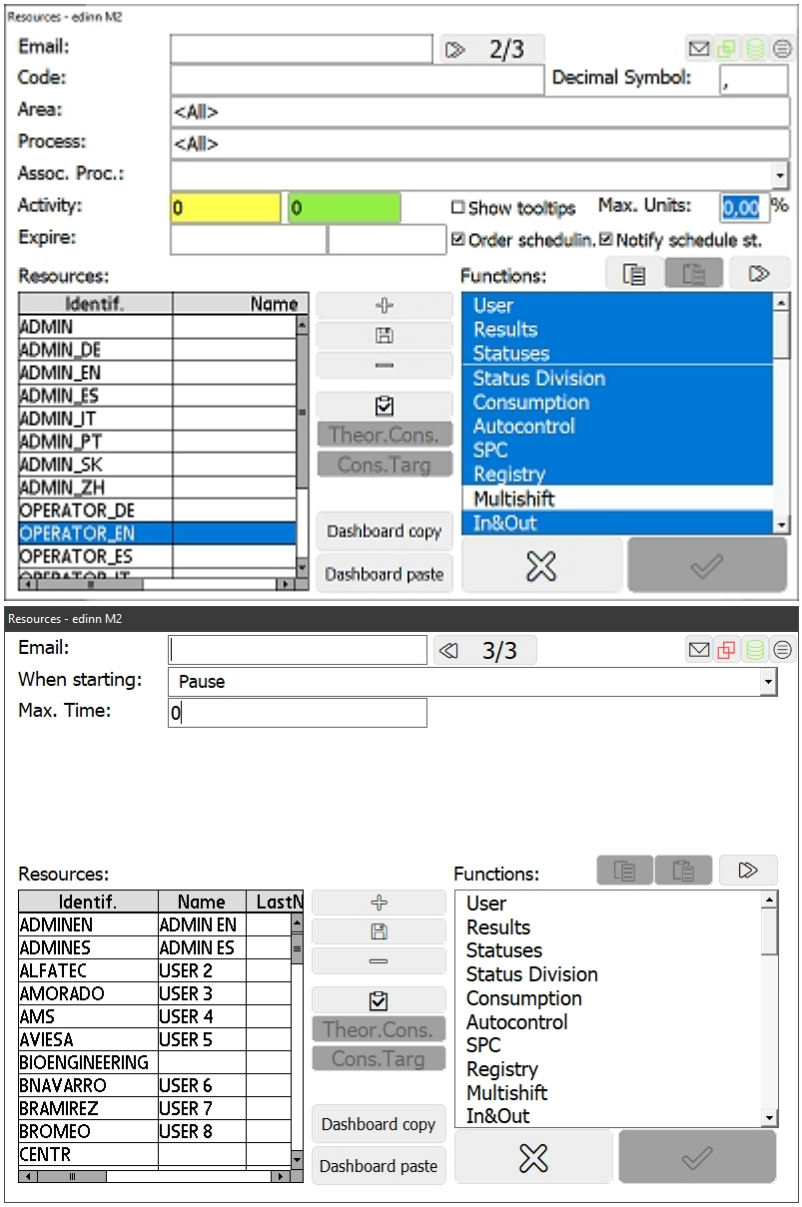
A **resource** is any person who wants to use the system, it must be registered at the Resources table shown hereafter and to have the operator role.

On the left: you will see a grid with the resources already configured

In the middle part: buttons to perform operations on the grid such as **Insert**, **Save** or **Delete**.

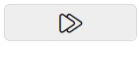
On the right: you will see the available roles for each resource.





CONFIGURATION PARAMETERS:

- **Email**: personal email of the user. It is not mandatory if the user is going to access only this actual organization, but it is mandatory if the user needs to access other companies.
- **Identif.**: identifier of the person for the system. If remote applications have been enabled, then it must be less than 10 characters long.
- **Name**: name of the person.
- **Last name**: last name of the person.
- **Password**: secret password to access the system. Take into account that if the device to which the user connects is configure with Easy Login, the user will login faster if the password is a numeric number. If remote applications have been enabled, then it must be of at least 7 characters and must have a lower case letter, an upper case and a number.

- **Edit check**: this must be marked in order to update a user password.
- **Language:** to configure the system when the user connects to the system.
- **Email1** and **email2**:
  - **Email1** is used to receive:
    - notifications or comments that are sent through the comments system.
    - instructions on how to recover the user's password.
    - the list of the autocontrol tasks that are about to expire.
    - what messages, in the communication flow between edinn and other systems (ERP, etc.), have failed and require manual intervention.
  - **Email2** is used to receive:
    - notifications of not done autocontrol tasks
    - notifications of processes that are stopped more than a certain time.
-  shows more options:
  - **Code**: code for quick access from the login window. This is useful for card readers, so that the user just passes the card and automatically logs in the system.
  - **Area**: if the user is restricted to access only the process of certain areas. Indicate them separated by commas: AREA1,AREA2. The user will only be able to see the areas indicated here. Leave blank to see all areas.
  - **Process**: if the user is restricted to access only these processes. Indicate them separated by commas: PROC1,PROC2. The user will only be able to see the processes indicated here. Leave blank to see al processes.

WARNING: once areas or processes are restricted to a user, even that the user is an administrator, only a Superadministrator user will be able to remove the restriction. This is useful to assign different administrator to different areas, or, for example, for different working facilities which have different administrators, and these facilities are configured as areas.

WARNING: when from the windows to configure areas or processes new areas or processes will be generated, for security reasons, they will not be automatically assigned to those users who have their fields of Areas and Processes in blank. This assignment must be performed manually.

  - **Assoc. Proc.**: associated process. This is useful when a process is a person, and has the following effects:

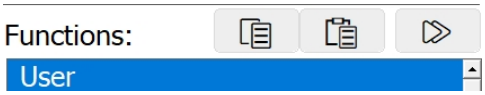
- In the results window, statuses and consumptions, if the user did not select any process, the terminal will propose the associated process.
- In the main window, if the user click on Schedule and there is no selected process and no alert to attend, he will be brought to the associated process.
- When an order is scheduled or imported, if it is done on a process which has a person associated, or the user doing it has an associated process, then the default status of the order will be the one marked as the default for the type production.
- In the view schedule window, this resource will only be able to manage orders assigned to his process, except if he has the scheduler or supervisor roles.
- When the user is assigning a work order to a process, if done to the 100%, it will not refresh the grid with the orders of the destination process.
- If the resource is not scheduler (see below), then cannot start, prefinish, cancel nor pause an order which is not scheduled for the associated process.
- This resource will only see resources which are associated to processes that this resource can see, unless this resource has the Administrator role. This is to prevent that users, which belong to areas in companies shared by multiple users in different areas, could see users without an associated process or associated to processes which they cannot see. Therefore, resources without an associated process, will be visible for any other resource.
- The estimated dates to finish a task or work order will be the indicated limit (due) date.
- More than one resource can be associated to the same process.
- If the user wants to user a status that requires results, they will be required when the user uses that status in his/her associated process.

- **Activity**: percentage of activity that is required for the working user. This is a requirement configuration for the activity report.
- **Show tooltips check**: if tooltips help must be shown next time the user logs in. The user will be able to disactivate this feature.
- **Expire**: after this date, the person will not be able to access the system, will not be shown in the resources list at the login page nor will be able to access the reports tool.

- **Max. Units:** the maximun number of resources that can be used at the same time.
- **Order Scheduling**: if marked, indicates that the resource can sort the data table of the Schedule window. Otherwise, the table will be automatically sorted by the WBS column.
- **Notify schedule status changes**: if marked, indicates that the resource will recive schedule status change notifications, as long as the resource is responsible
- **When starting**: what should happen when a new work order is started, to pause, to prefinish or to finish the work order which is being processed in that moment.
- **Maximum Time (Max. Time):** Maximum uninterrupted time in seconds that the resource can be IN in a process. After this time, the system will automatically OUT. If left at zero, it will not be applied.

BUTTONS:

- **Copy Link**: copies to the clipboard the http link to open the dashboards of the user. This function is only available for resources with the Reports role and when the password is updated. This function is not available in the Community version.
- **Check:** lets you know which users have the previously selected role. For example: the admin user has the multishift role.
- **Theor. cons.**: to configure the theoretical consumptions for the person selected. Please see: Theoretical Consumption.
- **Cons. Target**: to configure the targets of consumptions for the person selected. Please see: Consumption Targets.
- **Dashboard copy / paste:** each user has his own customized dashboard. To apply the #user1 configuration to another one, press [Dashboard copy] on the #user1, and then press [Dashboard paste] on the #user2.
-  : switches between the selection lists of **functions, scheduling and production** fields. For the **functions**, please see below. For the **scheduling** fields, these are the columns of the grid in Schedule window and the columns that each user can see can be configured here. For **production**, these are the fields that will be shown as columns when the *I/O window* is grouped.
- **Copy functions between recourses**: To do this, select the recourse you want to copy functions from, click on copy button  , select the target recourse and click on paste button 



- **Functions:** permissions according to the functions of the user.
  - **User**: allows to enter the system.
  - **Results**: allows to enter the results window.
  - **Statuses**: allows to use the status window.
  - **Consumption**: allows to use the consumption window.
  - **Autocontrol**: allows to use the autocontrol window.
  - **SPC**: allows to use the quality (SPC) window.
  - **Registry**: allows to access registry window.
  - **Multishift**: allows to view any shift and not only the actual shift.
  - **In&Out**: allows to indicate that in a process are working specific persons.
  - **Autoctrl.super.**: allows to approve the autocontrol tasks.
  - **Manual Infor.**: allows to manually inform of production quantities that have exceeded the target quantity.
  - **Supervisor**: allows to modify and delete any data of the system. It also allows
    - To see all the processes he can see, from any device, and not only the interesection between the processes he can see and those the device that he is using can see.
    - In the case that the user has an associated process (see above), the user will be able to start work orders in processes which are not his/her associated processs.
  - **Administrator**: allows to configure the system. It also allows to see all favorite reports created. If you activate this role, you will need to update the password. Please keep it secure. By using the right mouse button from the main window, it allows to assign this role to a resource associated with a process and only for the selected area, which will allow the associated resource to:
    - **Rename area**: change the description of the area of which the user is administrator.
    - **Assign/disassign processes** to the area of which the user is administrator.
    - **Assign/disassign roles**: only the **Scheduler** and **Administrator** roles are available and only applying to the selected area.
  - **SPC Multishift**: allows to view, regarding the SPC data, any period and not

only the actual shift.

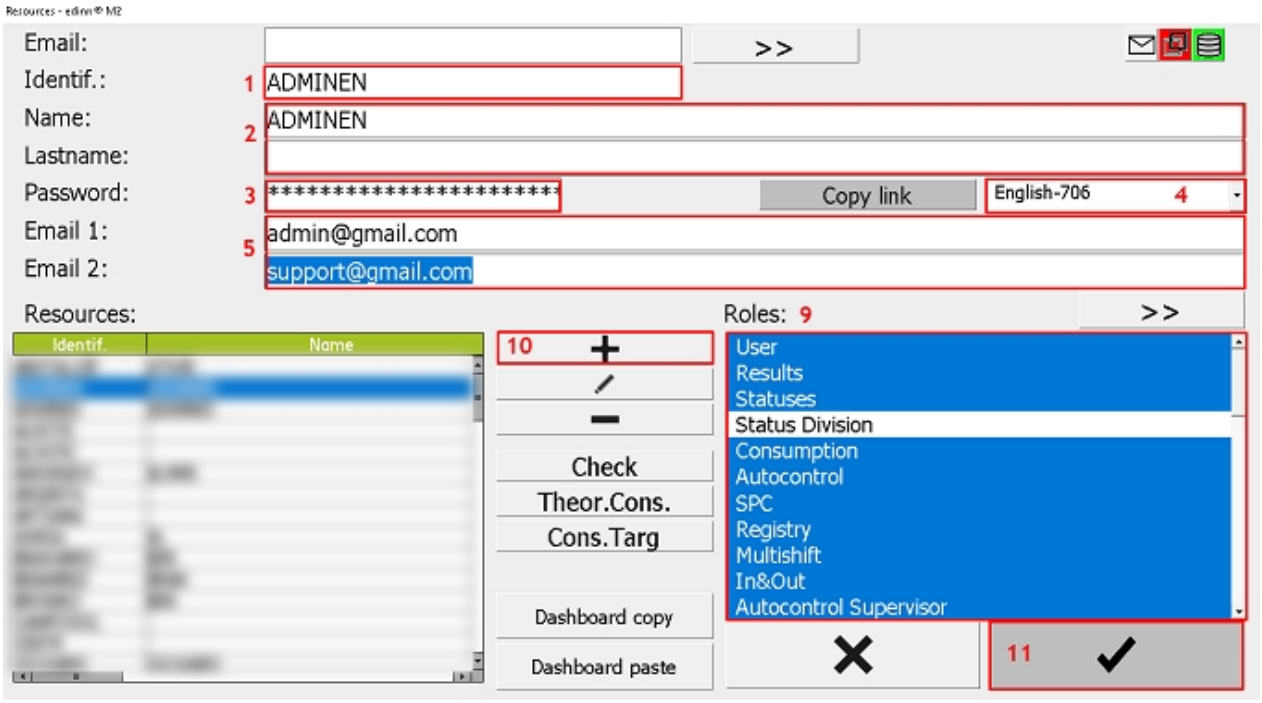
- **SPC Supervisor**: allows to modify and delete, regarding the SPC data, any working data of the system.
- **SPC Administrator**: allows to configure the system, only regarding SPC issues.
- **Scheduler**: allows to access the edit schedule window.
- **Quality checker**: allows to introduce scrap or rewored products manually from the results window.
- **Results changer**: allows to change results using the Modify button at the results window. This is usefull for the supervisor to delegate the change of the order, result or fields in records of results, but without allowing to:
    - Change the quantity.
    - Change the time.
    - And if the process is configured as Scheduled, cannot neither:
        - Change the order, unless the new order was/is active at the time of the change.
        - Change the result, unless the new result was/is inside the order which was/is active at the time of the change.
- **Reports**: allows to access to the on-line reports.
- **Assign**: allows to transfer a work order to a process, from the View Schedule window.
- **Multiprocess**: allows the user to indicate that he/she is working at various processes at the same time.
- **Remove Op.**: if a process only allows one working person, this role allows the new person that starts to work with the IN button, to expel the previous person that did not sign off with the OUT button.
- **Finish sche.**: allows to mark a order as finished at the View Schedule window.
- **Cancel sche.**: allows to mark a order as canceled at the View Schedule window.
- **Pause sche.**: allows to mark a order as paused at the View Schedule window.
- **New result**: allows to use the button New result at the results window.
- **Edit result**: allows to use the advanced editing functions of the results

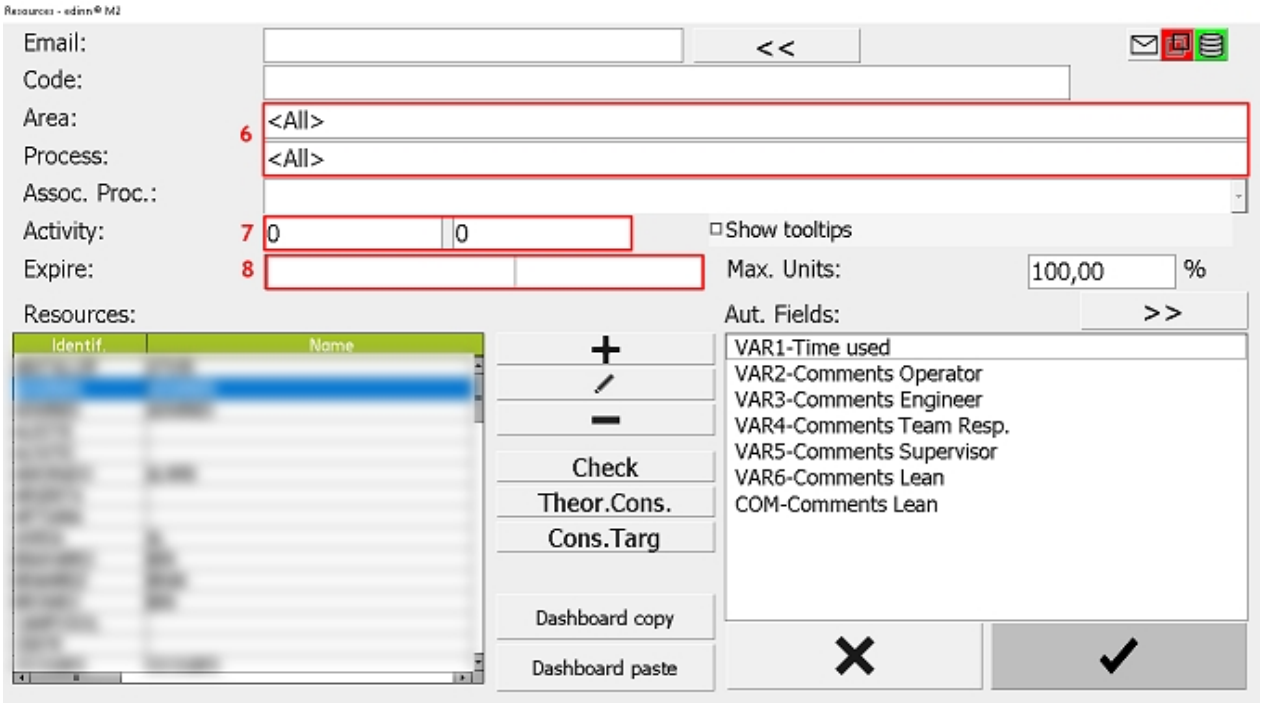
window.

- **Consumption admin**: allows configuration of the consumption aspects of the system.
- **Cost**: allows to see economical information.
- **Multiorganization**: allows, among other multi organization functions, to update password in all companies that the user has access to.
- **Common Favorites**: allows the user to create favorites reports common to all the users in the reports tool.
- **Superadministrator**: sees all areas and processes and therefore can assign them to any device or other resource. This rol is convenient to assign resources and devices to multilevel structures of areas and processes with the right mouse button from the main window.
- **Prefinish sche.**: allows to mark a order as prefinished at the View Schedule window.
- **View schedule**: allows to view the schedule at the View Schedule window.
- **View inputs**: allows to view *inputs*.
- **View outputs**: allows to view *outputs*.
- **Edit inputs**: allows to edit *inputs*.
- **Edit outputs**: allows to edit *outputs*.
- **Divide by two**: allows to divide by 2 a status even when it has not been justified by the user.
- **Manual Prod. Informer:** it allows to manually insert (but not justify) a production (working) status, as long as it is for the current time.

How to configure a resource step by step

The resources of a process are usually the users and operators. In [**Click to configure**] select [**Resources**].





Step 1. Configuration for each resource

Follow the steps below to create a new edinn user:

1. **Identif.:** define an identifier (nickname) for the user.
2. **Name & lastname:** write the name and last name of the user.
3. **Password:** define an initial password for the user, it can be modified later by the user.

4. **Language:** select the language of the terminal for the user.
5. **Email 1 & 2:** write 1 or 2 emails to notify the user of different events or incidents.
6. **Area & processes:** press  and in [**Areas**] and [**Processes**], like devices, people also have permissions to see certain processes and areas. To limit this visibility, write in both fields the areas and processes that you want to be displayed, separated by commas (e.g: Area: L1, L2, L3, Process: B902, A101.)
7. **Activity:** define a percentage or goal of activity for the user, eg: if you want to incorporate incentives based on the activity of the employee, you can set a goal of 100% when a resource (person) gets a process with an OEE ratio greater than 80% , in this way, a resource (person) can reach an activity ratio of more than 100%, and therefore, receive incentives.
8. **Expire (optional):** set an expiration date for the user's permissions. Usefulness: if the company has temporary workers, the permissions can be limited to a specific season, to avoid that other employee mistakenly uses the identifier of an operator that is no longer working or, to avoid configuring again a resource, if it is reincorporated the the following season. In this case, only the expiration date would have to be deleted.
9. **Resources:** select from the table the functions or roles that the configured user will have (eg: *give the user permission to justify stops, view reports, start self-control tasks...*). To see the available roles, please go to Resources configuration.

> **NOTE:** this is the common configuration, the rest of the fields are usually defined by default, for more information, please go to: Resources configuration.

Step 2. Save changes

10. Press  to insert the user's configuration in the table. Repeat this action for **each** resource or user.
11. Press  to make all the changes effective.

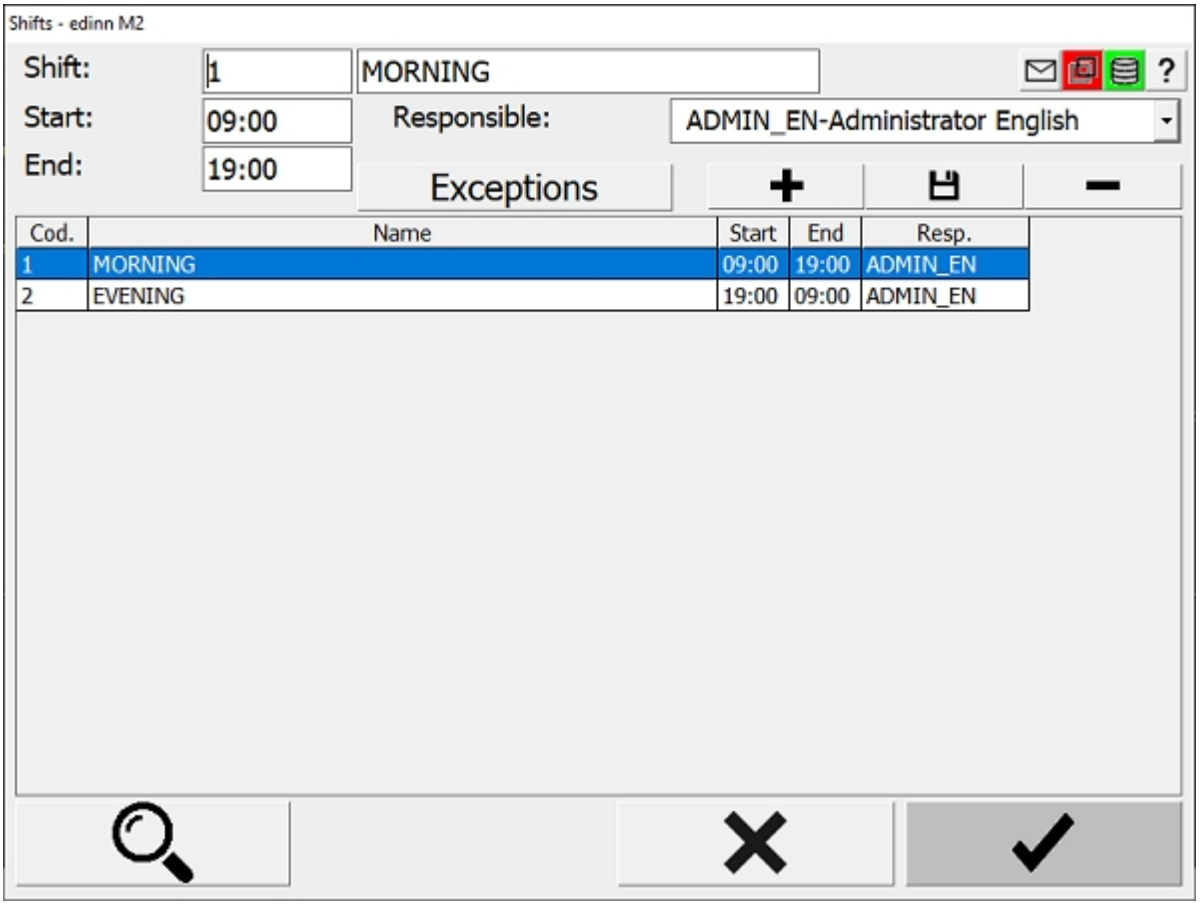
Shifts

Click to configure > Shifts

The **shifts** configuration window allows the management of the organization working shifts.

In the lower part: you will see a grid with the shifts already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

In the upper part: you will see the configuration of each record, where you can configure the parameters of a new insertion or edition, and the exceptions.



CONFIGURATION PARAMETERS:

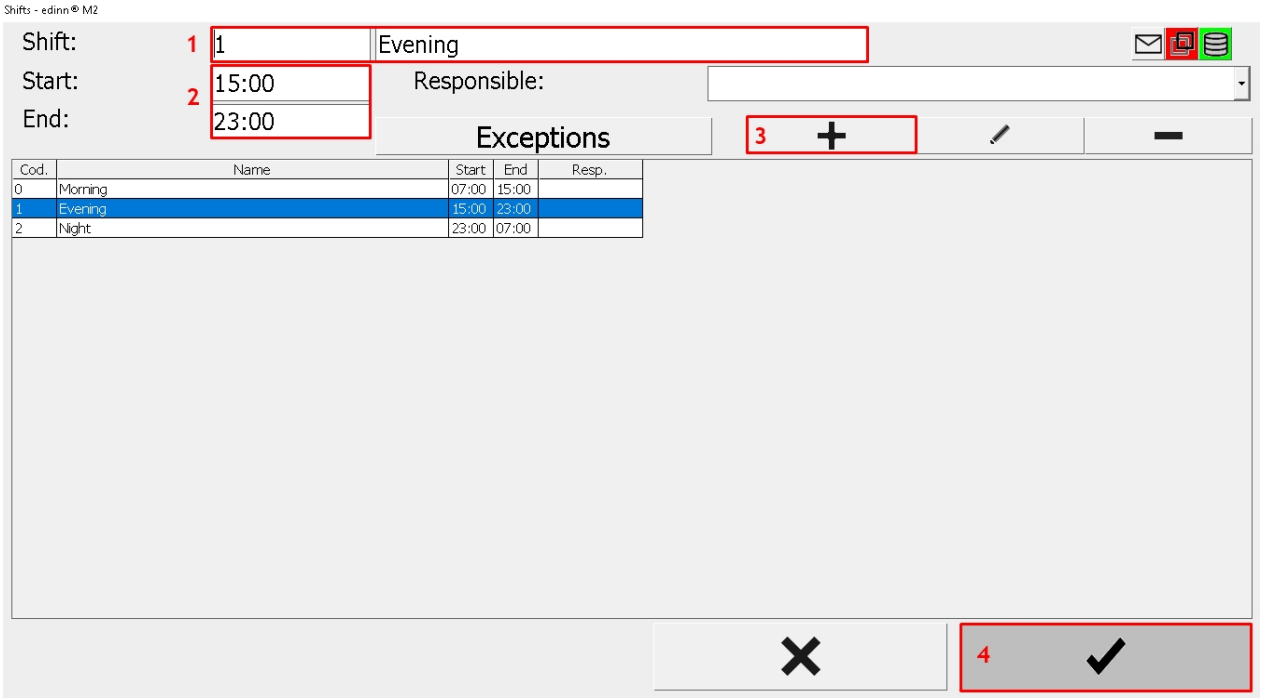
- **Shift**: code and description of the shift.
- **Start / End**: time of start/end of the shift in the hh:mm format.
- **Responsible**: select the person responsible for a specific shift. If it is configured, the person would receive certain email notifications.
- **Exceptions button:** please see exceptions.

WARNING: it is very important that always for every process, exists an associated shift. In other words, we should always cover the 24 hours for every process. If we do not configure this correctly, we could cause functioning issues, because, for example, one process could have an exception and for a certain moment would not have any associated shift. If you have doubts, you can use the configuration checking feature.

How to configure a shift step by step

The shifts in which the day is divided must be configured. In [**Click to configure**] select [**Shifts**].

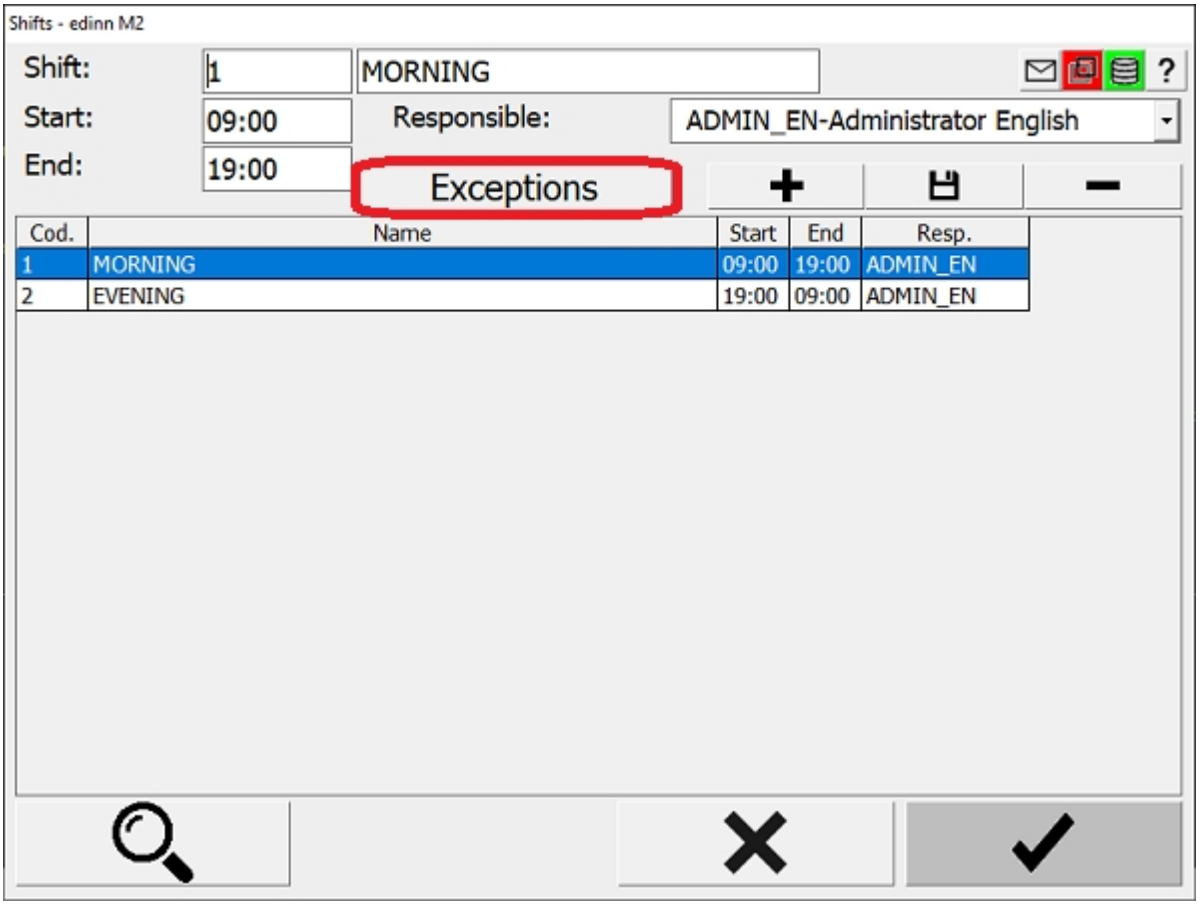
1. **Shift:** enter the identifier (e.g: *1*) and the name (e.g: *Evening*) of the shift.
2. **Start & End:** enter the time period.
3. Press ✚ to insert the shift in the table. Repeat this action for **each** shift to be created.
4. Press ✔ to save all the changes.



Exceptions

Sometimes, you may need to configure different overlapped shifts if different processes start and/or end at different times in the morning, evening or night. In order to do so, firstly you will need to create the shifts, assigning them the corresponding processes using exceptions. If you try to create overlapped shifts, a warning message will show, you will have to confirm the action to procede. If you need to know how to create or modify shifts, please check: Shifts.

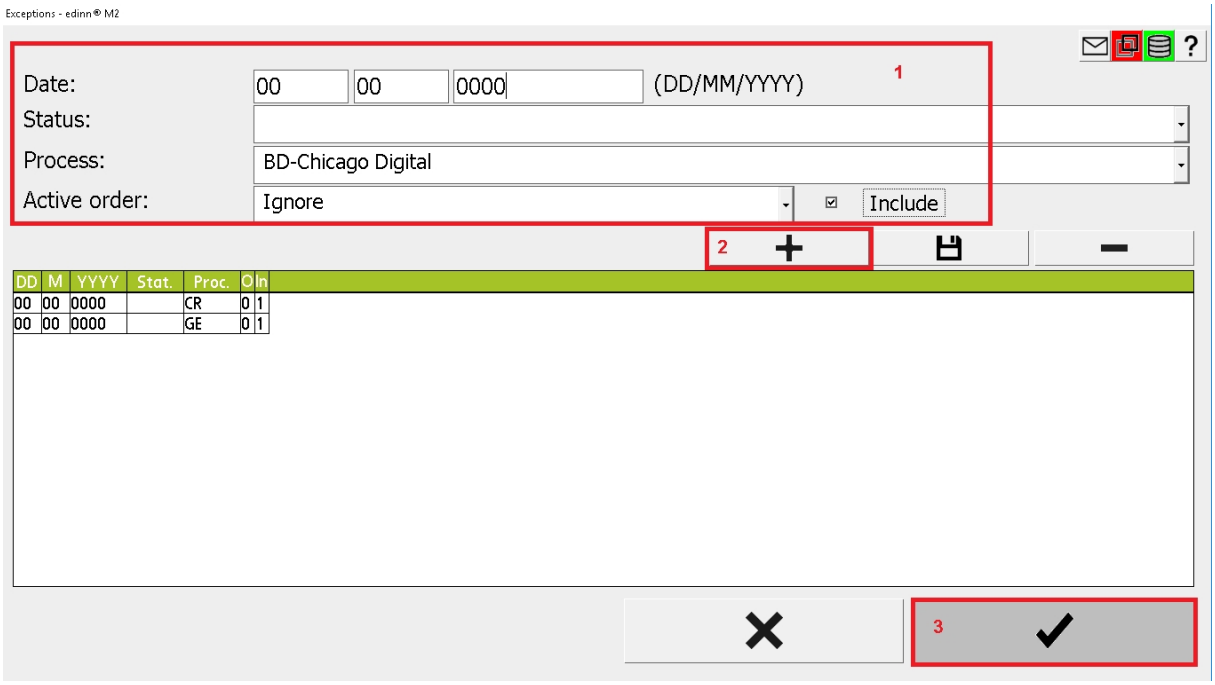
1. Once shifts have been created, select one of them from the table and click the **[Exceptions]** button:



2. For each process you need to include, fill the fields in the upper part of the window (1) and click ✚ to assign the process to this shift (2):

- **Date**: Exception date. We can use zeros as a wildcard. Examples:
  - Indicating **00/00/000** means everyday. You probably want to use this option.
  - Indicating 00/08/0000 means every august of every year.
  - Indicating 01/05/0000 means every May 1st of every year.
- Status: useful to indicate an exception on certain statuses. It won't be necessary in this case.
- **Process**: Select the process that you want to include in this shift.
- **Active order**: useful to indicate exceptions when there is no active order. Select Ignore.
- **Include**: Indicates if the exception is inclusive or exclusive. If not marked, it is an exception. **If it is marked: it is an inclusion**. Therefore, you need to mark it.

3. Press ✔ to confirm the changes(3):



4. Follow the steps again to include the corresponding processes to the remaining shifts.

Team

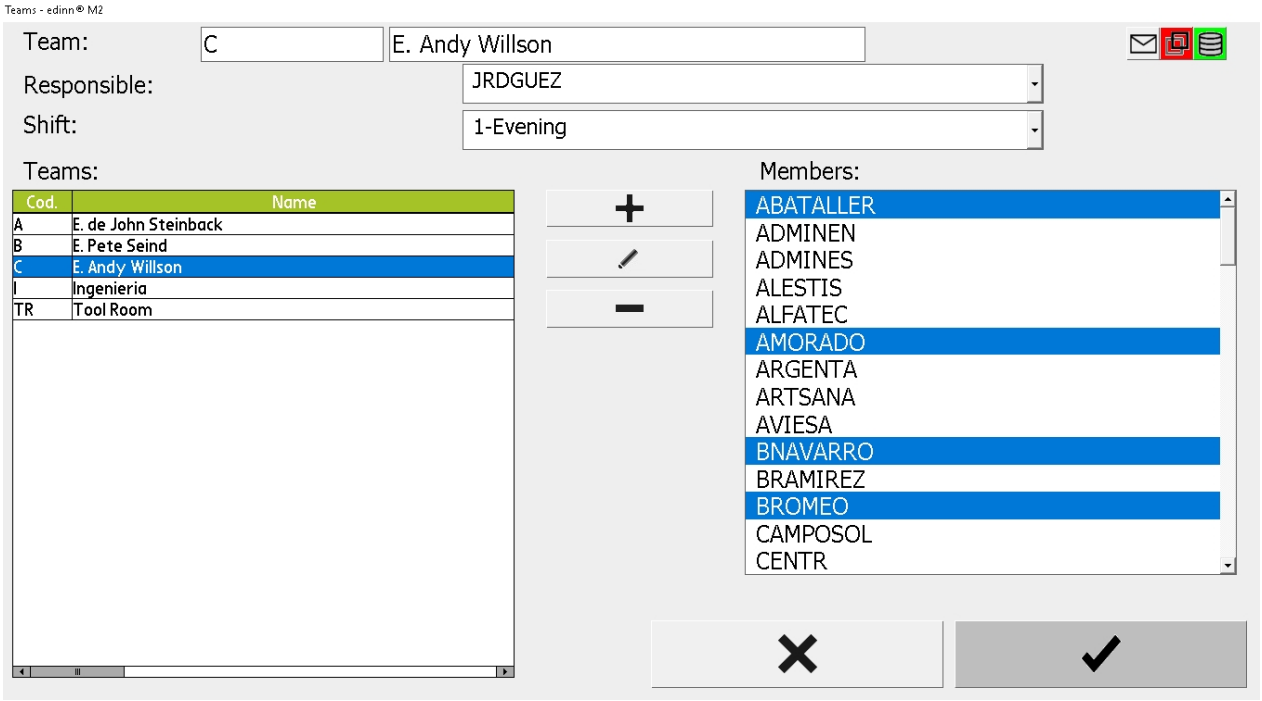
Click to configure > Teams

A **team** is a group of persons. It is useful to obtain reports filtered by teams. It should not be confused with Shifts.

On the left: you will see a grid with the teams already configured

In the middle: buttons to perform operations on the grid such as **Insert**, **Save** or **Delete**.

On the right: you will see the members of each team.



CONFIGURATION PARAMETERS:

- **Team and description:** code and name or description of the team.
- **Responsible**: select the person who is in charge of the team. If it is configured, the person would receive certain email notifications.
- **Shift**: the shift to which the team is assigned. If the system is correctly configured (at a server level), teams will start their shifts automatically and this team will automatically start to work on all processes, when the corresponding shift starts.

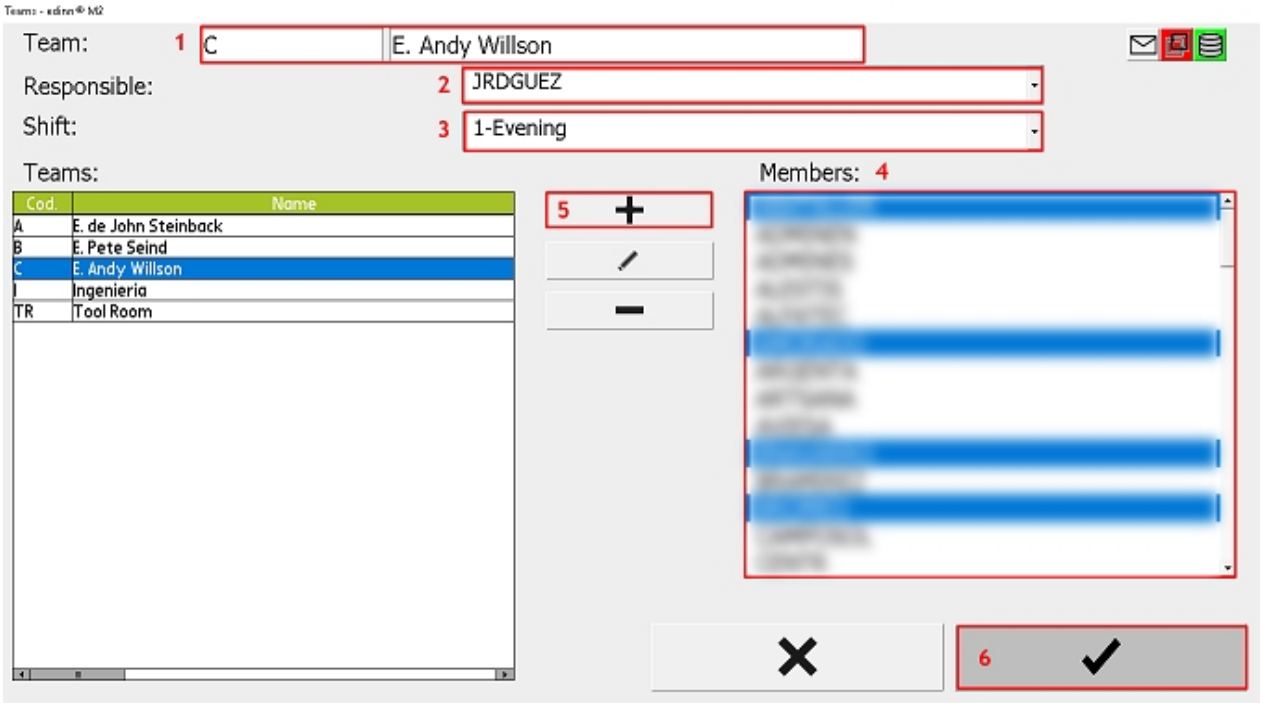
How to configure a team step by step

The next step is to configure the teams, which are groups of users or workers. In [**Click to configure**] select [**Teams**].

1. **Team:** type the name (e.g: Team 1) and identifier (e.g: T1) of the team.
2. **Responsible:** enter a team responsible (from the available resources, configured in Resources).
3. **Shift:** enter the shift in which the equipment works.
4. **Members:** select the members that make up the team (from the available resources, configured in Resources).
5. Press ✚ to insert the equipment into the table. Repeat this action for **each**

team to create.

6. Press ✓ to make all the changes effective.



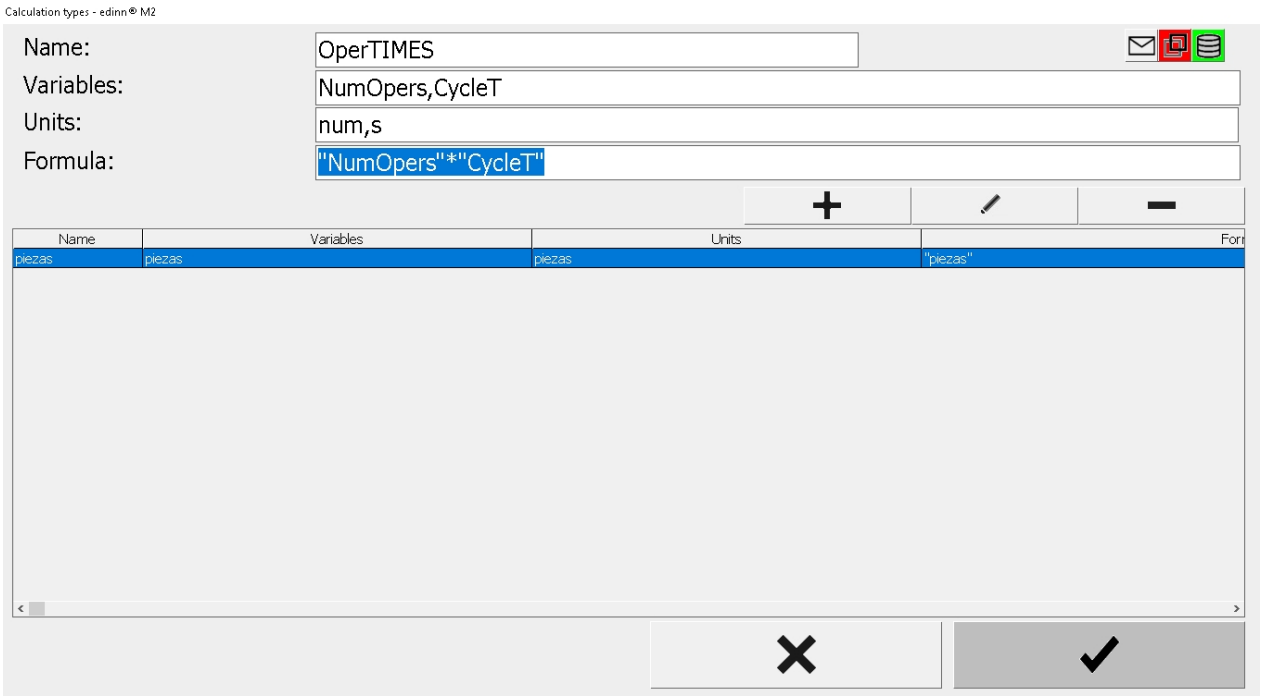
Calculation types

Click to configure > Calc. types

The **calculation types** are different formulas to perform calculations. For example: *to calculate production or scheduled times to generate results.*

In the lower part: you will see a grid with the calculation types already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

In the upper part: you will see the configuration of each record, where you can configure the parameters of a new insertion or edition.



CONFIGURATION PARAMETERS:

- **Name**: name of the calculation type (and how it will appear to the user)
- **Variables**: variables that will be requested in order to calculate the quantity. There are predefined variables that can be used:
  - **CycleOT**: Operator cycle time.
  - **CycleT**: Cycle time.
  - **CycleQ**: Quantity per cycle time.
  - **Created**: Date and time of creation of the field.
  - **DateAsRef**: The date and time of the last result, or when a change in the result id or order id happened.
  - **NumOpers**: Number of operators working on the process.
- **Units:** measurement units that will be used to express the result of the calculation.
- **Formula**: mathematic function used to calculate the result. Variables must be enclosed with quotation marks, for example: "Width".

EXAMPLES:

- Suppose that our process is producing square meters of material (you should indicate exactly the following):
  - Name: m2

- Variables: width, height
- Formula: "width" * "height"

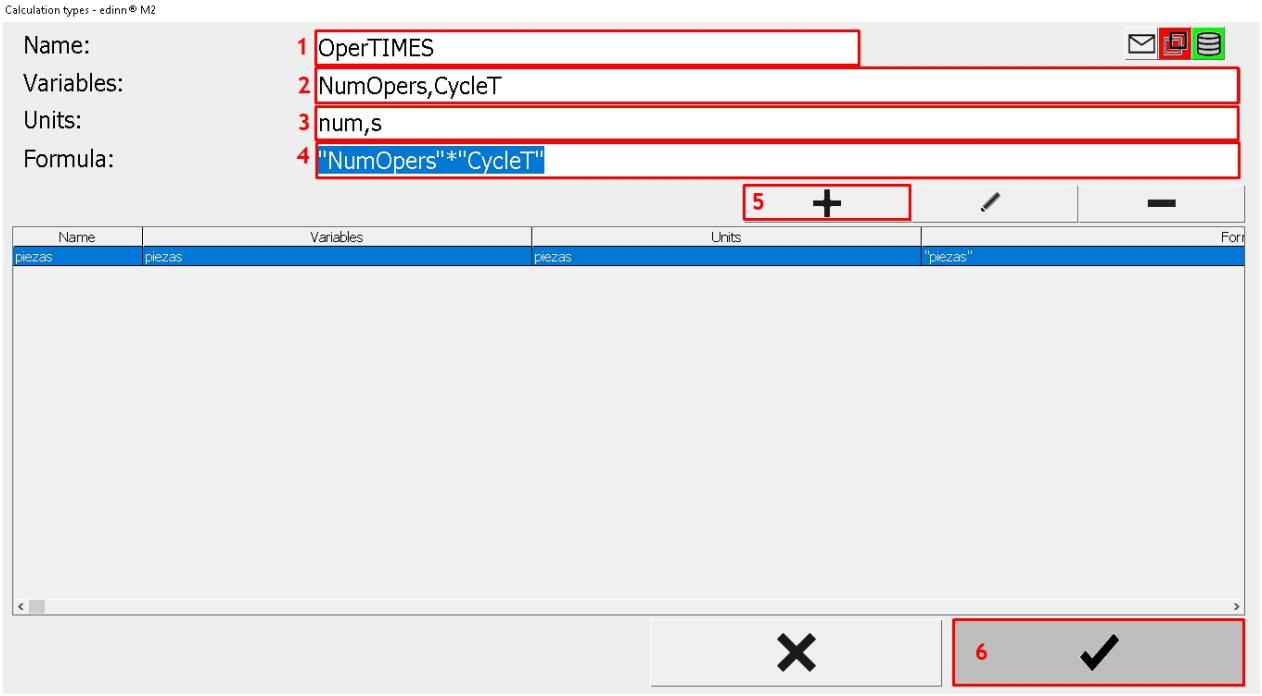
- Suppose that our process is producing bags:
  - Name: bags
  - Variables: bags
  - Formula: "bags"
- Suppose we need a type of calculation to let the user introduce production cycle time in seconds (although the user will introduce time in minutes)
  - Name: EstTime
  - Variables: esti
  - Formula: 1/("esti" * 60)
- Suppose we need a type of calculation for the user to introduce the date and time when the result will be finished:
  - Name: EstiDateTi
  - Variables: Created,EstiDateTime
    - Donde EstiDateTime debe ser un campo de fecha hora.
  - Formula: 1/DateDiffSec("DateAsRef";"EstiDateTime")

How to configure a calculation type step by step

Configure the types of calculation, which serve to calculate the equivalence between what edinn monitors with respect to the units that the factory is actually producing. In [**Click to configure**] select [**Calculation types**].

1. **Name:** enter the name (e.g: *Bags*) of the calculation type.
2. **Variables:** enter the variables that will intervene in the calculation.
3. **Units:** introduzca las unidades de dichas variables.
4. **Formula:** Write the formula to calculate the type of calculation named in the first step.
5. Press ✚ to insert the type of calculation. Repeat this action for **each** type of calculation to create.

6. Press ✓ to make all the changes effective.

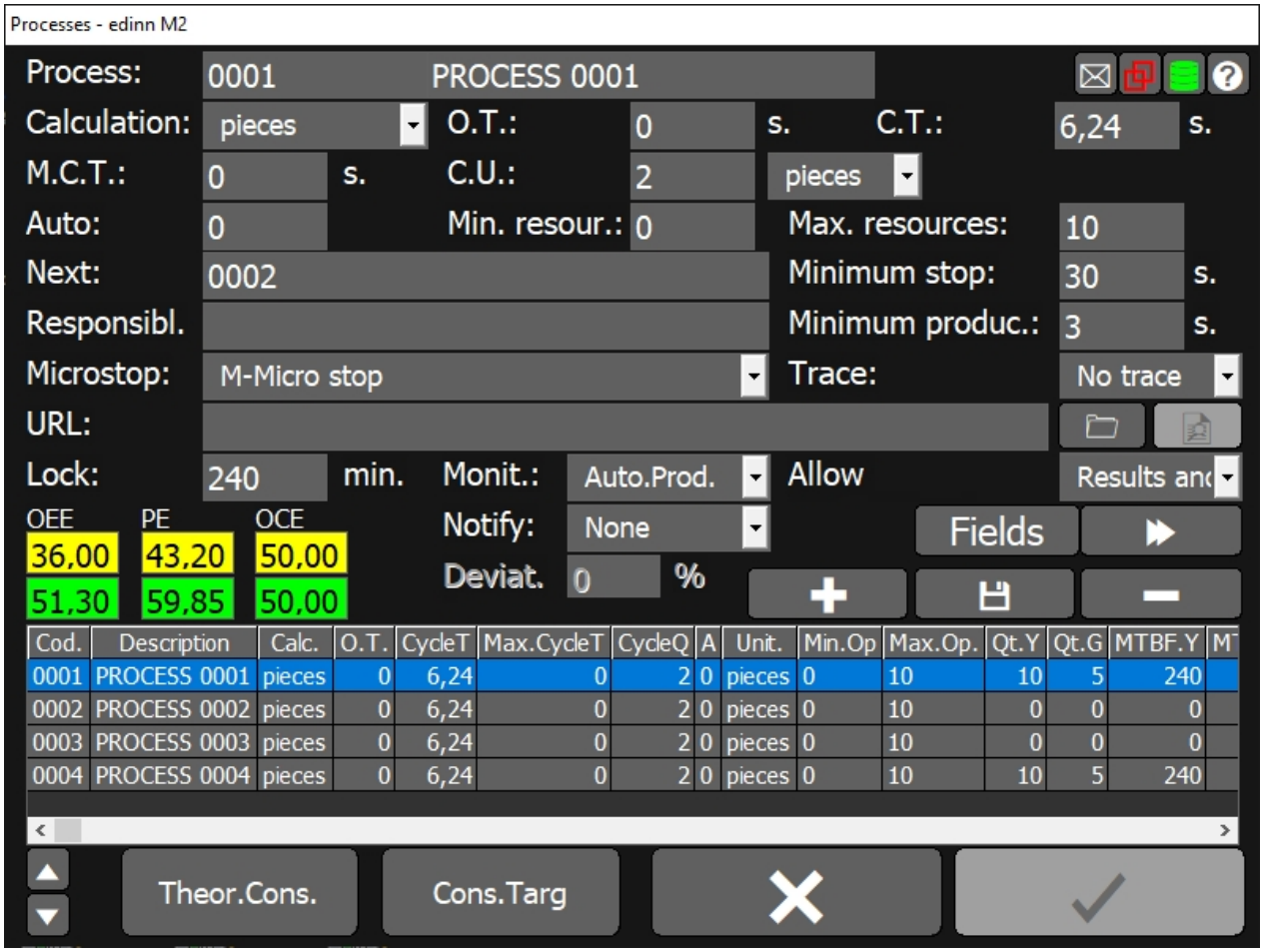


Processes

Click to configure > Processes

In the lower part: you will see a grid with the processes already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

In the upper part: you will see the configuration of each record, where you can configure the parameters of a new insertion or edition.



CONFIGURATION PARAMETERS:

- **Process**: code and description of the process. For example: *GE, Electrogen group*.
- **Calculation**: how to calculate the units produced by the process. The available options can be configured at the Calculation types configuration.
- **O.T.** (*Operator Time*): time in seconds of work of the resource (person) per cycle time.
- **C.T.** (*Cycle Time*): time in seconds per cycle of production. Please see Maximum speed.
- **M.C.T.** (*Maximum Cycle Time*): if different than 0,
  - It will be the default cycle time for orders created with the Integration Module (ISA-95).
  - There will be a special behaviour for statuses marked as Req. result.
  - The monitor will adjust Cycle Time and Cycle Units based on this parameter.
  - See later as it will generate certain special behaviours in combination with

operation time.

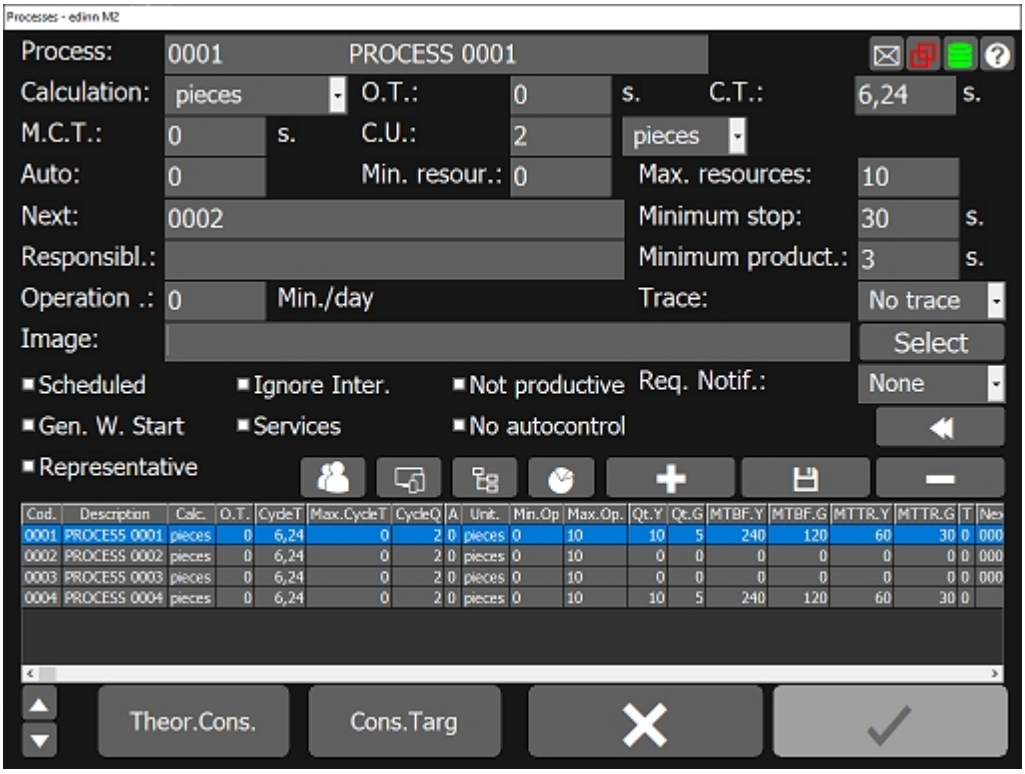
- **C.U.** (*Cycle Units*): number of units expected per cycle of production. Please see Maximum speed.
- **Unit selector**: the unit identificator of what the process produces.
- **Auto**: indicates the number of units of results with a maintained speed for the system to automatically update the maximum speed of the process. Automatic update of maximum speed is disconnected if this value is left to 0. This options is used when the cycle time is unknown.
- **Min. Opers / Max. Opers**: minimum and maximum number of working persons that can work at the process at a certain moment.
- **Minimum stop**: minimum stop time in seconds in order to be registered as failure (FAI). Stops with less time will be registered as microstops if the field **microstops** has been informed below.
- **Next**: processes that follow the actual process. Please see difference between production lines and areas.
- **Responsible**: person in charge of the machine in case of notifications.
- **Production minimum:** minimum time in seconds that a process must be working to be shown as working. This parameter is also used to determine the time that a piece needs to go from a counter of totals to a later counter of goods.
- **Microstops**: status that will be automatically used to automatically justify microstops.

NOTE: detecting microstops requires that the server checks status of counters more frequently than the "minimum stop time". In general, the more frequently the server check the status of counters, the more precision you have in detecting microstops. Please contact your technical support to validate this.

- **Trace**: to indicate, if signals are monitored, and the information is sent to the terminals (Devices) or is left alone in the server (Central).
- **URL**: document or application that will be opened when the user clicks on the  that appears after having clicked on the process, on the main window. Please see how to configure URLs. URLs can be selected from  and its syntax can be cheched from the button  .
- **Lock**: indicates after how many minutes status and result registers will be blocked for users without supervisor permission. This allows to force justifications before a certain time. This parameter will also allow to send notifications when statuses are not justified.
- **Monitor:** select the type of monitoring:

- **Manual**: the process is not monitored automatically and therefore buttons are shown for the working user to indicate manually when a good or defective piece is produced.
- **Auto. Prod.**: production is automatically monitored. The rest (rework and scrap) should be indicated manually.
- **Auto. Scra.**: scrap is automatically monitored. The rest (production and rework) should be indicated manually.
- **Auto. Prod. Scr.**: production and scrap are automatically monitored. The rest (rework) should be indicated manually.
- **Auto. Rewo.**: rework is automatically monitored. The rest (production and scrap) should be indicated manually.
- **Auto. Prod. Rewo.**: production and rework are automatically monitored. The rest (scrap) should be indicated manually.
- **Auto. Scra. Rewo.**: scrap and rework are automatically monitored. The rest (production) should be indicated manually.
- **Full auto**: all types of results are automatically monitored.

- **Allow conf.:** allows the user to configure the process and to add more statuses and more results.
- **Notify:** if selected, the system will notify by email (1) when the selected ratio (OEE, Availability, Speed or Quality) is below the yellow target, in any of the defined periods. Please see Notification priorities. You can also select **Scheduling** and then the notification will be sent when finishing or pre-finishing an order and having produced a certain % more or less than the target quantity. You can indicate this excess or default % in the field **Deviation**, which will be activated only in this case.
- **Fields:** to configure **default values** for the fields of this process. Fields allow the user to adapt the platform to a specific industry. For more information, please see: Fields Configuration.
- **Ratios configuration (OEE, PE, OCE)**: green and yellow targets for the OEE and PE (is the same as the OEE but without considering idle statuses) and OCE ratios for the processes. By clicking any of these fields, more targets will appear: please see Ratios configuration.



-  shows more options:
  - **Operation time**: for Work Management, type the seconds per day that the process should work in labour days. It is useful to calculate availability as the percentage of time that the process was working comparing with this number, independently of the time schedule that the process followed. It also affects the speed ratio, which turns into the percentage of the time that the process met the due dates of the orders it performed. Therefore, if different than 0, it will affect how the availability and speed ratios are calculated and:
    - **Availability**: you should schedule statuses to define, for this process and every moment, which days are of classes holiday, vacation and weekend. All the rest of the time will be considered as labour. For each labour day, the system will expect the process to have been working the time indicated here. The percentage of availability will be calculated based on the real production time versus what is indicated here.
    - **Speed**: you should use the scheduling module. It will be calculated according to the orders performed by the process in the period, comparing the production time with the delayed time with respect to the due date and time limit of the order. If there is no delay, then will compare the actual dedicated time to the task or work order versus the estimated time for it. Therefore, for a process which is a person to maintain high the speed ratio, he/she should assure that the limit date of the tasks or work orders is on the

future before starting them, that he/she finishes them before the limit date and that he/she works in them less time than their expected duration.

- If, in addition, the Maximum Cycle Time is different than 0, and the Scheduling Module is activated, then the monitor will calculate the time when the process should generate results as what happens first between the due date of the current working order and the date resulting by adding the Maximum Cycle Time to the last date when results where generated, considering only labour periods.
- The process will automatically pass to a stopped status if it is without an active work order for more than the time of Stop Minimum and the moment of the next result is not calculated based on fields (for example with DateAsRef).

- **Image**: select the image for the process.

> **NOTE:** If you upload a new version of a file with the same name, and it does not refresh to the new version, it might be because this function uses the web server and some web servers provide images from cache. In that case, please upload the same image with a different name of file.

- **Scheduled**: if marked the process is restricted to follow the orders scheduled for it, and that can be viewed at the view schedule window.
- **Ignore inter.**: marks the process as to be ignored when transferring from and to information to another system, like ERPs. Please see *integration with other systems*.
- **Req. Notif.**: this selector forces the user to indicate quantities when closing an order in the selected items.
- **Gen W. Start**: this mark indicates to the system to automatically generate a new status to indicate that a new work has started, every time a new result with 0 quantity is inserted, if at the status configuration the option Work start has been marked for one, and no more than one, status.
- **Services:** indicates that the process is dedicated to attend services. This mark will, for example, show the working time of these processes as services in the FTE report.
- **Not productive:** indicates that the process is not productive and therefore:
  - The system will not calculate OEE, OCE, MTBF and other ratios for this process.
  - The system will not show it in certain views and reports.

- The monitor service will ignore all items associated with this process.
- Its calendar will not be updated.
- Alerts about this process will not be notified.
- No failure predictions will be made for this process.
- It will not be shown in the drop down lists to manage work orders.
- It will be ignored by the automatic production scheduler.

- **No autocontrol**: disactivates all automatic generation of autocontrol tasks.
-  : to select the resources to authorize in the selected process.
-  : to select the devices with access to the selected process.
-  : to select the areas to assign to the selected process.
-  : to create a dashboard for the selected process, using a template. These dashboards, when associated to devices, can be used as a screen saver in the configuration of devices.

- **Theor. cons.**: to configure the theoretical consumptions for the selected process. Please see: Theoretical Consumption.
- **Cons. Target**: to configure the targets of consumptions for the selected process. Please see: Consumption Targets.

WARNING: Some process configuration parameters influence the way in which the Monitor Service captures the information automatically from the process and also the behaviour of Central Services. Therefore, if we change certain parameters, it is recommended to advise the system technical support to optionally adjust the server and restart the monitoring systems.

How to configure a process step by step

To create and configure the processes of a plant, complete the following minimum required configuration in [**Click to configure**] select [**Processes**]. To configure the rest of the fields, access Configuration processes.

Step 1. Configuration for each process

1. **Process:** write the name (e.g: *Pulisher Group A*) and identifier (e.g: *A901*) of

the process to be configured.

NOTE: If new processes or areas are created, check that they are visible to devices or users.

2. **Times & resources:** configure each field (Calculation, O.T, C.T, M.C.T, C.U, Auto, Min. Oper, Max. Oper.) fundamental for calculating the process availability ratio.

3. **Next:** for linear layouts, determine what the next process is. Utility: if a stop occurs in Machine 1, if it affects Machine 2, the Machine 2 will suffer a "Dependency" (which does not affect the OEE calculation of this) instead of a "Stop".

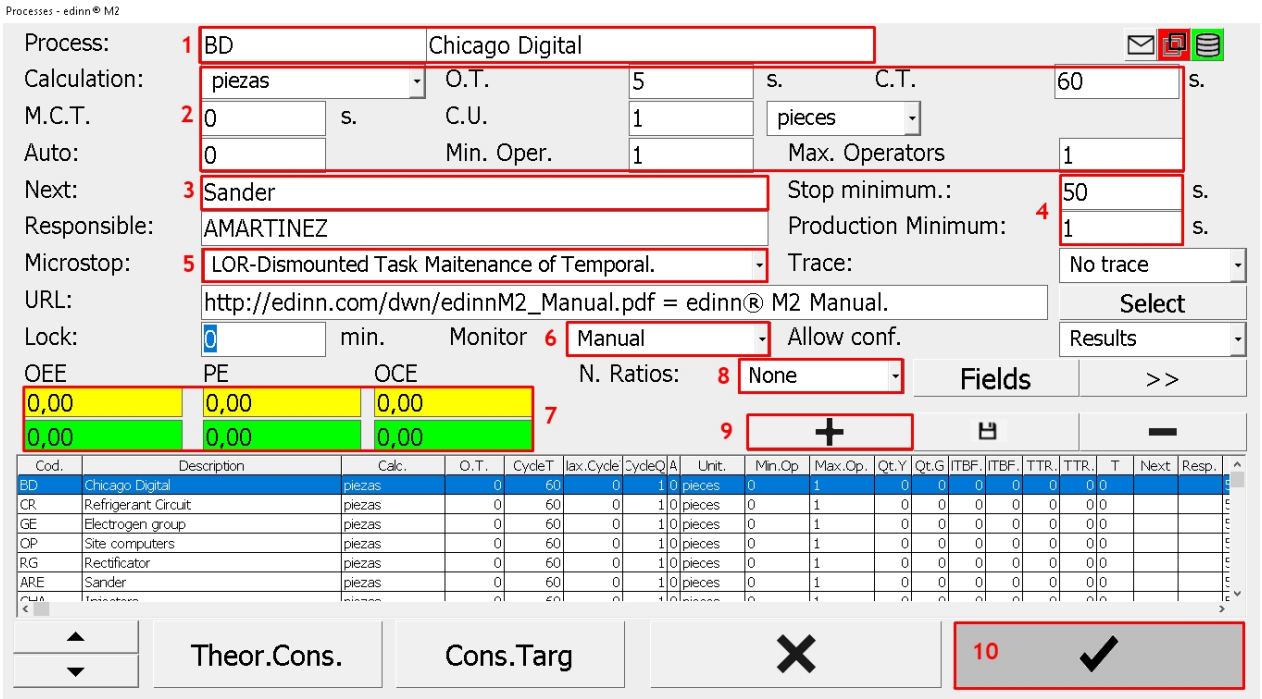
4. **Stop & production minimum:** define the minimum time for the monitor to consider a production (generally, the time used to manufacture two pieces) and the minimum time from which a stop will **not** be considered micro-stop and, therefore, it will have to be justified by the user.

5. **Microstop:** in case of detecting a micro-stop in the production, indicate with what status it will be qualified.

6. **Monitor:** indicate if the quality issues (scrap, rework...) are going to be automatically or manually monitorized.

7. **N. Ratios:** selecte if you want the system to notify by email when the selected ratio (OEE, Availability, Speed or Quality) is below the yellow target.

8. **(IMPORTANT) Ratios:** configure the green and yellow objectives for each of the process variables, for more information, see Configuration Ratios.



Step 2. Save changes

9. Press ✚ to insert the process configuration into the table. Repeat this action for **each** process of the plant.

10. Press ✓ to make all the changes effective.

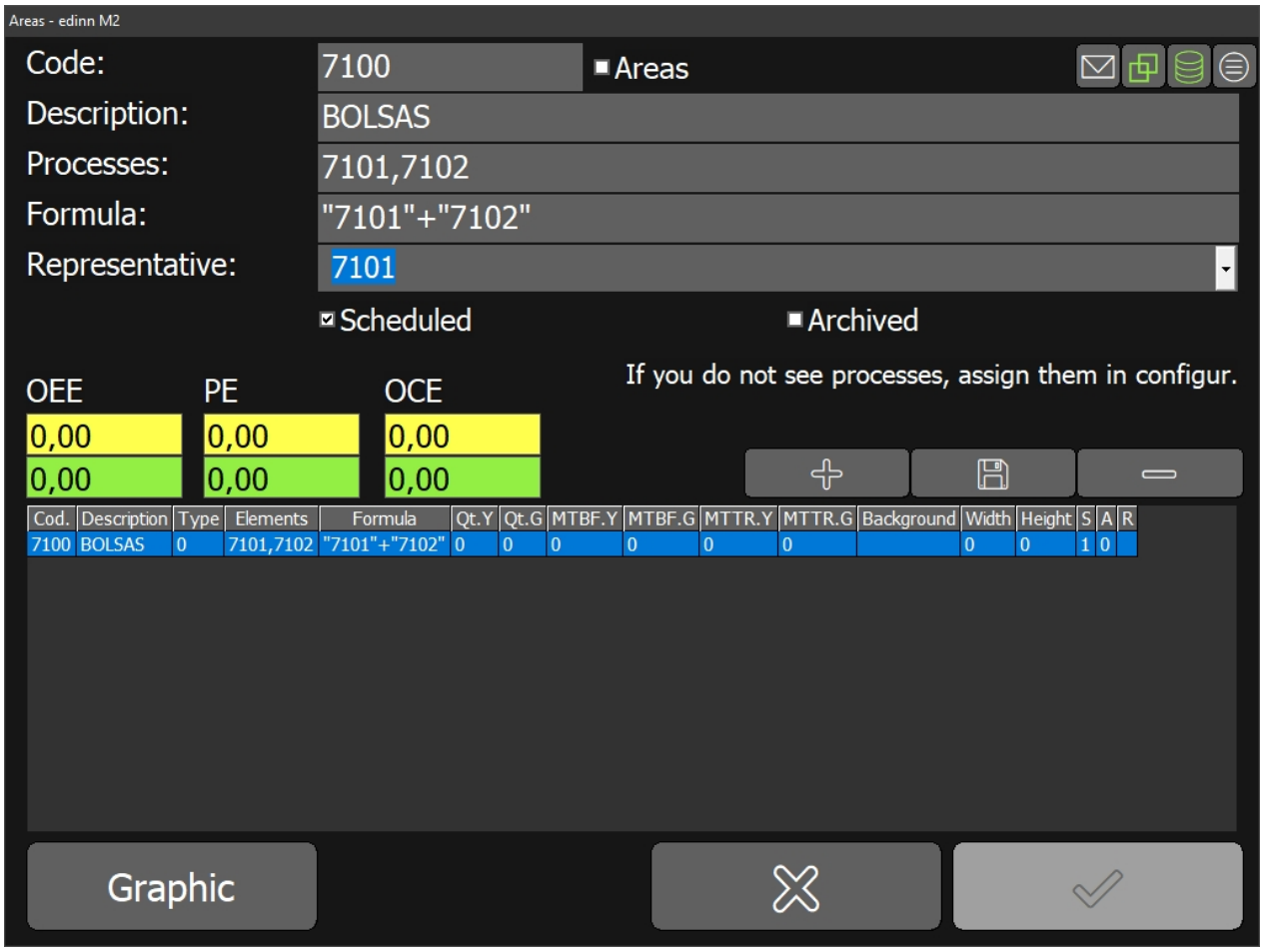
NOTE: the commonly configured fields have been presented, to know the definition of the other fields, please go to: Configuration Process.

Areas

Click to configure > Areas

In the lower part: you will see a grid with the areas already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

In the upper part: you will see the configuration of each record, where you can configure the parameters of a new insertion or edition.



CONFIGURATION PARAMETERS:

- **Area and description**: code and description or name of the area.
- **Area check:** to indicate if the selected area is an area of areas.
- **Processes**: list of processes that compose the area, separated by commas. Example: A*0L1,A0L2*.
- **Areas:** if the areas checkbox is checked, list of areas that make up this area.
- **Formula**: mathematical function to calculate the equivalent ratios of the area in function of the ratios of the processes. For example: if the area is composed of only 1 process, then the formula would be the id of the process enclosed with quotation maks. If the area is composed of 2 proceses, for example A0L1 and A0L2, the basic formula would be "A0L1" + "A0L2", although we could use any mathematical formula upon our needs. As we can see, variables, in this case the id of the processes, must be enclosed with quotation marks. If an area is composed of many processes but we want the ratios of the area to depend only on a subset of them, then the formula should only include those relevant processes.
- **Representative:** this field indicates if there is a process that is the representant of the area. This field is not available for areas of areas. The representative process has the following applications:
  - To be able to show the "path" graphic of an area.
  - In the reports analysis-ratios and analysis-evolution, when aggregated information of the area will be required, it will show this process as if it was the area.
  - If this process is in line (field Next seen above) with other processes, the statuses of those processes will be propagated to this process as they are, and not as a dependency. Therefore, this process may not generate not justified statuses
  - Blocking of results (configured in statuses) will not affect this process.
  - The OEE of this process will be updated with the one of the area it belongs to.
- **Scheduled**: if marked, when viewing the current status of the area, it will only show colours and alerts based on the processes of the area which have active working orders of this area. This is useful for a "project" perspective, where areas are not just a group of processes, but a list of orders (tasks), or in other words, a project. If not marked, when viewing the current status, the colour of the area will be calculated upon the statuses of its processes, which is useful when areas are not projects, but groups of processes.
- **Archived**: if checked, the area will not be available for production order

management, but it will be visible in the main window and in the reports.

- **Ratios configuration (OEE, PE, OCE)**: green and yellow targets for the OEE and PE (is the same as the OEE but without considering idle statuses) and OCE ratios for the areas. By clicking any of these fields, more targets will appear: please see Ratios configuration.
- **Graphic button:** allows to introduce an image customizable with the ratios. For instance: to introduce an image of the plant and relate each process image with their ratios. For more information please see: graphic configuration.

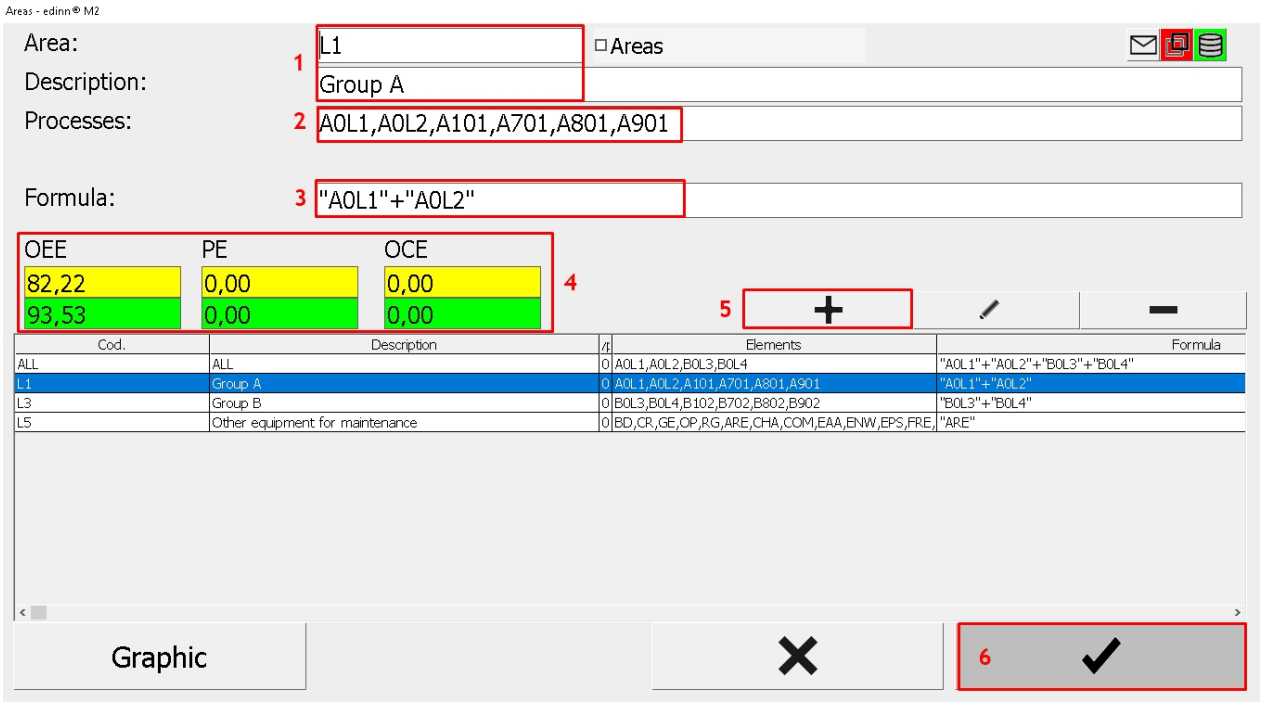
How to configure an area step by step

The areas encompass processes. To create and configure the areas of a working center, complete the following configuration. In [**Click to configure**] select [**Areas**].

Step 1. Configuration for each area

1. **Area & Description:** write a name (e.g: Group A) and an identifier (e.g: L1) of the area being configured.
2. **Processes:** determine which processes are included in the configured area, separated by commas, e.g: *40L1,40L2,40L3*.The areas are **not** exclusive, two areas can share the same processes.
3. Define the **mathematical function** to calculate the area ratios according to the processes. Variables must be enclosed with quotation marks or enclosed by parentheses and separated by ";", e.g: with MIN (1CON; 2ROT; 3DRI; 4CON; 5PRE) the ratio of the area is calculated using the lower ratio of the processes included in it.
4. Configure the OEE for the area.

NOTE: if new processes or areas are created, check that they are visible to devices o users.



Step 2. Save changes

5. Press ✚ to insert in the table the configuration of the area. Repeat this action for **each** area of the plant.

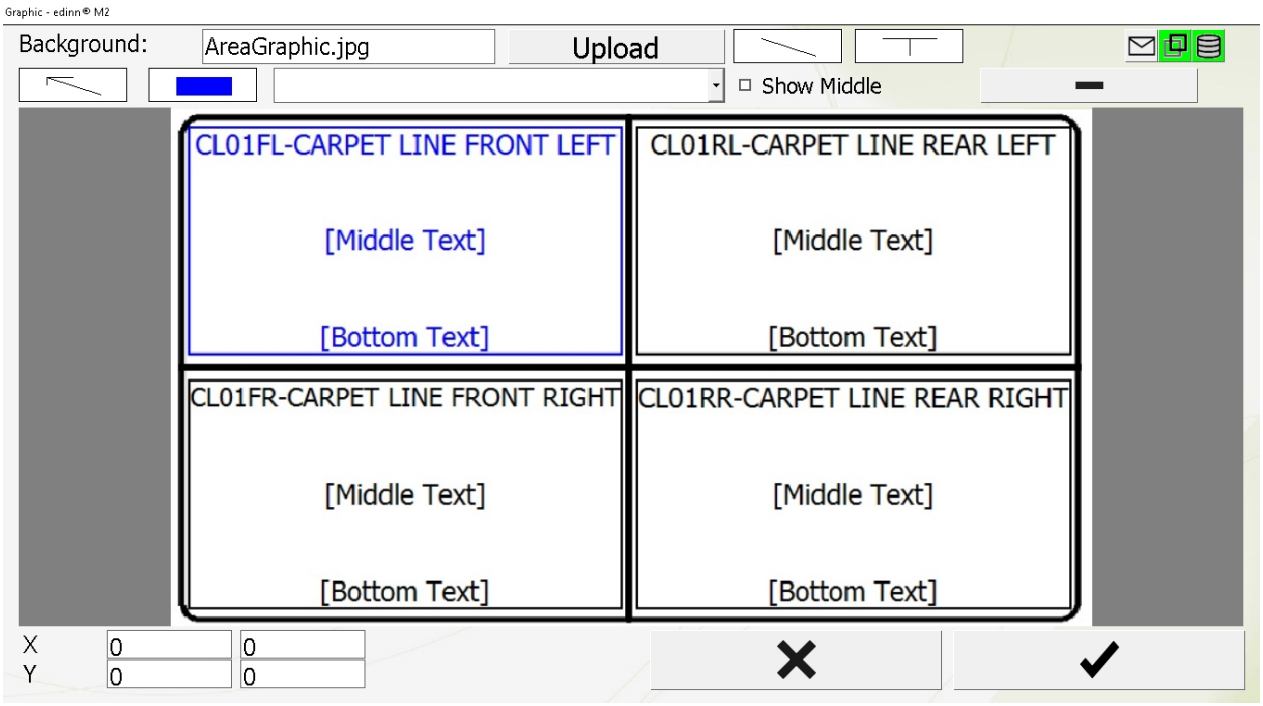
6. Press ✓ to make all the changes effective.

NOTE: the commonly configured fields have been presented, to know the definition of the other fields, please go to: Configuration Areas.

Insert a Graphic

The **graphic button** allow to introduce an image customizable with the ratios.

For instance: *to introduce an image of the plant an relate each process image with their ratios.*



- **Background**: indicates the name of the background image.

NOTE: If you upload a new version of a file with the same name, and it does not refresh to the new version, it might be because this function uses the web server and some web servers provide images from cache. In that case, please upload the same image with a different name of file.

- : to introduce a **line** in the graphic, useful to establish visual connections between objects.
- T : to insert **text** in the graphic.
- : to **select** elements within the graphic. We mark begin and end in order to select all elements within the defined square.
- : creates a rectangle that **identifies a process** in the graphic. First select a process in the following drop down combo and then drawn the rectangle.
- : allows to **delete** objects of the graphic.
- **X and Y**: allows to **modify the coordinates and the size** (width and height) of any object of the graphic.

The **correct resolutions** for the background image are:

- For Min (240x320): 159x228 (name_file2)
- For 640x480: 414x317 (name_file3)
- For 800x600: 517x396 (name_file4)
- For 1024x768: 662x507 (name_file5)

If we need a graphic for each resolution, files should be named: name_file2.jpg, name_file3.jpg, etc.

Consumption issues

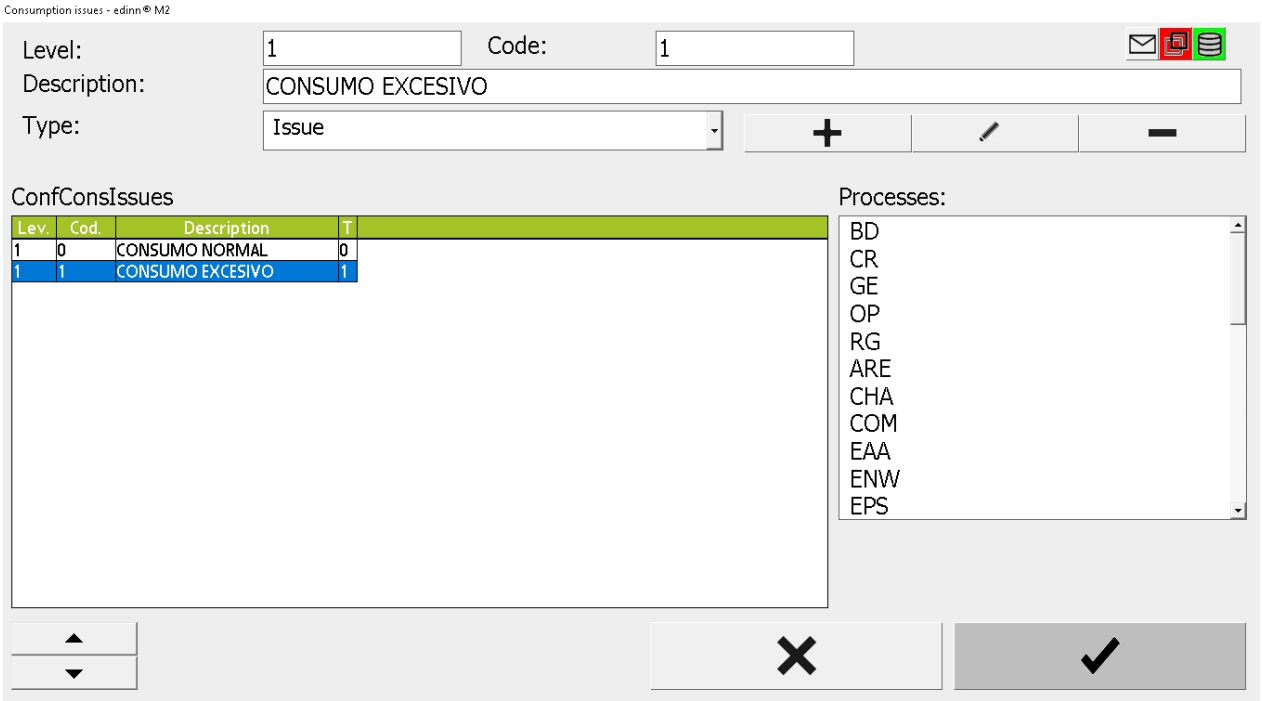
Click to configure > Consumption issues

Consumption issues are explanations for excessive consumptions. When correctly configured, the system will generate consumption issues if there is an excessive consumption. These consumption issues could need a user justification.

In the lower part: you will see a grid on the left with the **consumption issues** already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**, and a list of processes to with each consumption issue is associated on the right.

> **NOTE:** If one consumption issue is not associated to a process, it will not be shown in that particular process.

In the upper part: you will see the configuration of each record, where you can configure the parameters of a new insertion or edition.



CONFIGURATION PARAMETERS:

- **Level**: consumption issues are organized in 3 levels. These levels allow to show to the user the consumption issues in up to 3 levels. The user selects one option, and enters into the second level, selects another option, and goes to the 3rd level where he has to finally select a consumption issue.

- **Code**: code of consumption issue. It is recommended to use code 0 for the normal consumption.
- **Description**: description of consumption issue.
- **Type**: normal consumption, excessive consumption or title (which is the entrance to another level of consumption issues)

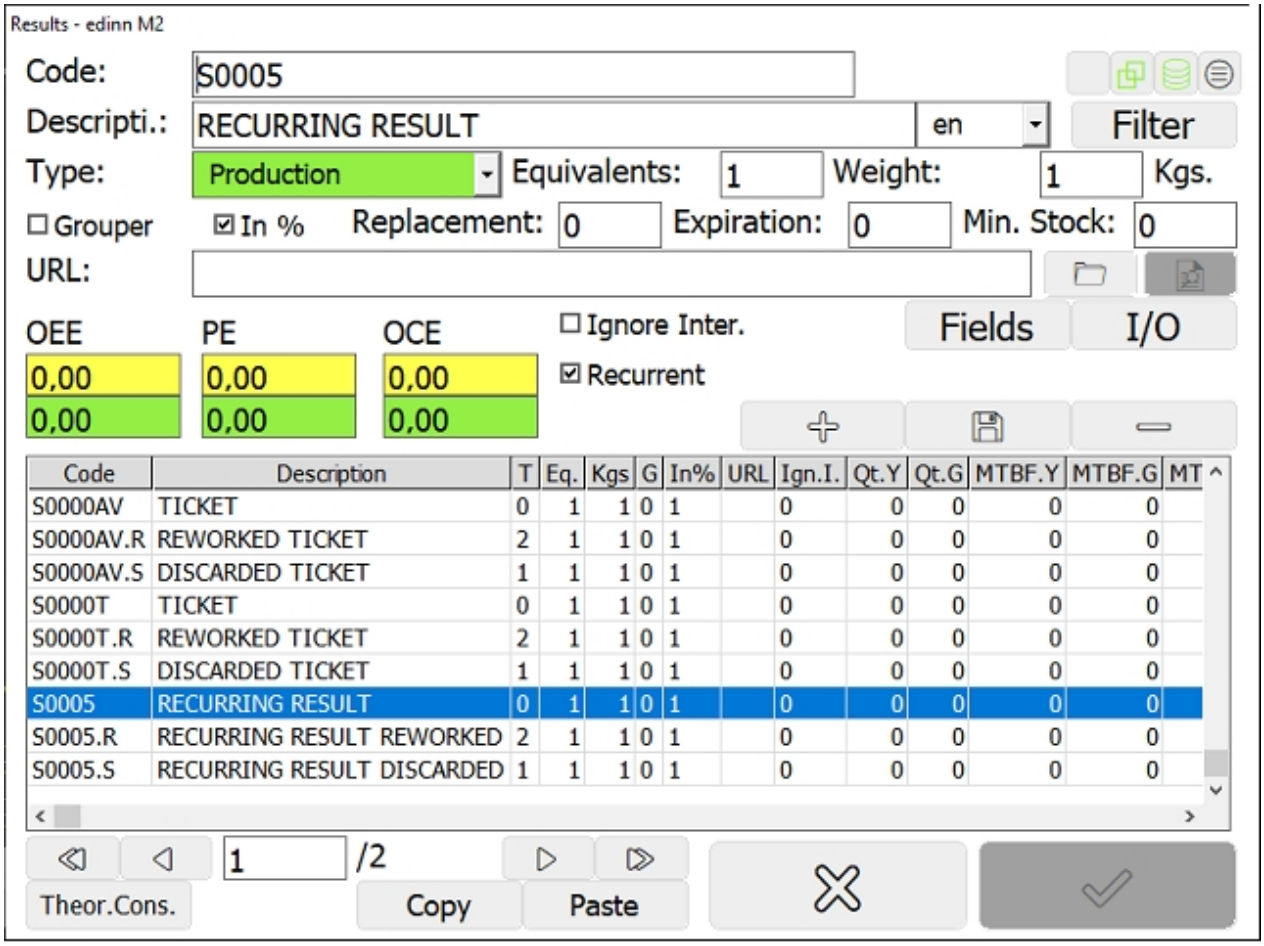
Results

Click to configure > Results

The *results* configuration window allows configuration of all possible results, including the good and the bad (rework and scrap). Also *inputs and outputs* of a process are results, so they must also be configured here. Please see the difference between consumptions and inputs.

In the lower part: you will see a grid with the results already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

In the upper part: you will see the configuration of each record, where you can configure the parameters of a new insertion or edition.



CONFIGURATION PARAMETERS:

- **Code and description**: Code and description or name of the result.
- **Language.**
- **Filter button:** to filter the results by the parameters established.
- **Type**: to indicate if the result is production, a rework or scrap.
- **Equivalence**: every production counting is equivalent to how many pieces. For example: 1 unit (a box) is equivalent to 6 pieces.
- **Weight**: weight of each piece (equiv.).
- **Grouper check**: if this result will be used to group results.
- **In % check**: if production is measured in %. Useful for results with a large cycle time (days or months) and to allow the working user to provide periodical advance of his work.
- **URL**: document or application that will be opened when the user clicks on the  that appears after having clicked on the result, on the results window. Please see how to configure URLs. URLs can be selected from  and its syntax can be cheched from the button  .
- **Ignore Inter. check**: marks the result as to be ignored when transferring from and to information to another system, like ERPs. Please see *integration with other systems*.
- **Ratios configuration (OEE, PE, OCE)**: green and yellow targets for the OEE and PE (is the same as the OEE but without considering idle statuses) and OCE ratios for the areas. By clicking any of these fields, more targets will appear: please see Ratios configuration.
- **Fields:** to configure new and customized fields, that allow the user to adapt the platform to a specific industry. For more information, please see: Fields Configuration.
- **I/O**: allows to define the pattern of inputs and outputs to generate this result.
-  : there can be so many pages of results, you can use the selector of page to navigate faster within them.
- **Theor. cons.**: to configure the theoretical consumptions for the selected process. Please see: Theoretical Consumption.
- **Replacement:** time it takes for the material to arrive since the order is placed (in days).

- **Expiration:** time in which a material expires since it is manufactured (in days).
- **Min. Stock:** minimum stock below which a new order must be placed.
- **Recurrent**: if this result will be recurring. Task or work orders which are marked as recurring can only generate recurring results.

How to configure a result step by step

In [**Click to configure**] select [**Results**]. To create and configure results, complete the following configuration.

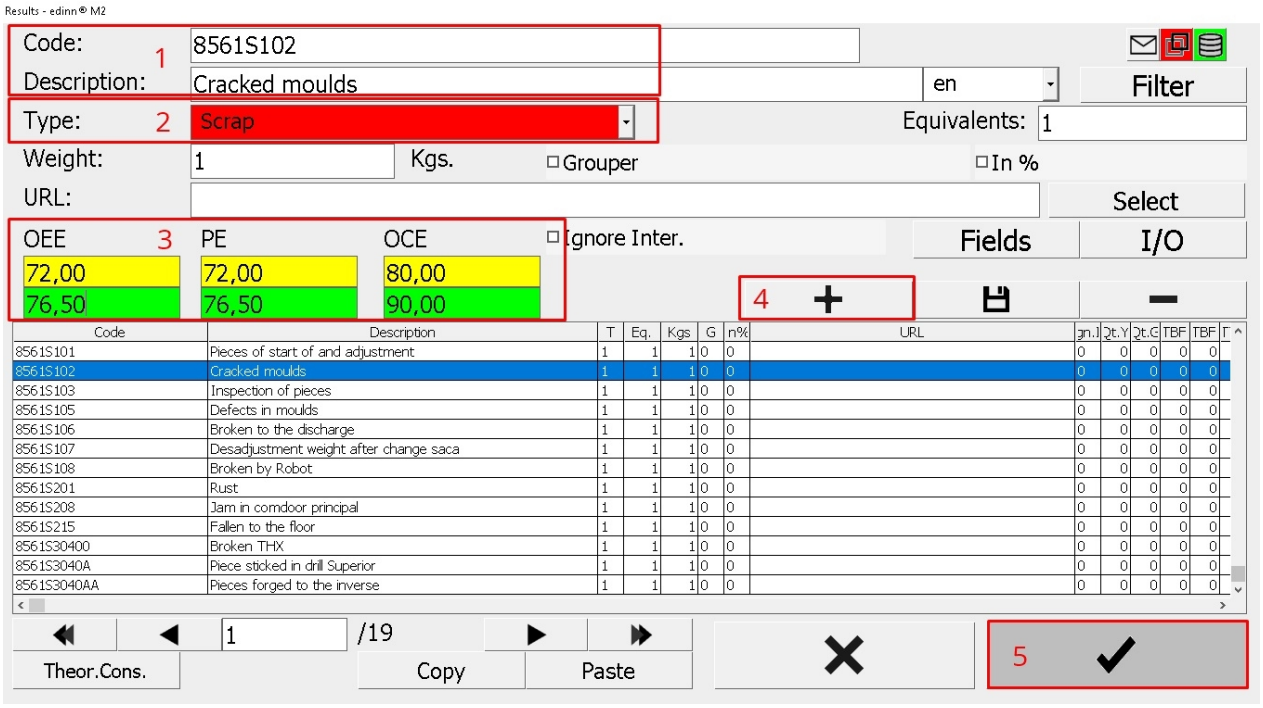
Step 1. List all the items

A same product can have different types of results: **Good** results (**Production**) & **Bad** results because of quality reasons (**Rework** (product that needs to be reworked) & **Scrap** (product discarded))

List all the products that your process produces for each type of result, e.g:

Step 2. Enter the list in edinn

1. **Code & Description:** write a result code (e.g: PROD1) and a description (e.g: Product 1). Add "SCRAP" or "REWORK" to the description or ".S" or ".R" to the result code for better differenciation.
2. **Type**: indicate if the result is a production, a rework or scrap. **Mandatory:** configure at least 1 Scrap result for each Production type.
3. Configure the OEE for the result.



Step 3. Save changes

4. Press ✚ to insert in the table the configuration of the result. Repeat this action for **each** result produced.

5. When all the changes have been done, press the ✓ button.

NOTE: Please see Results configuration for deeper configuration.

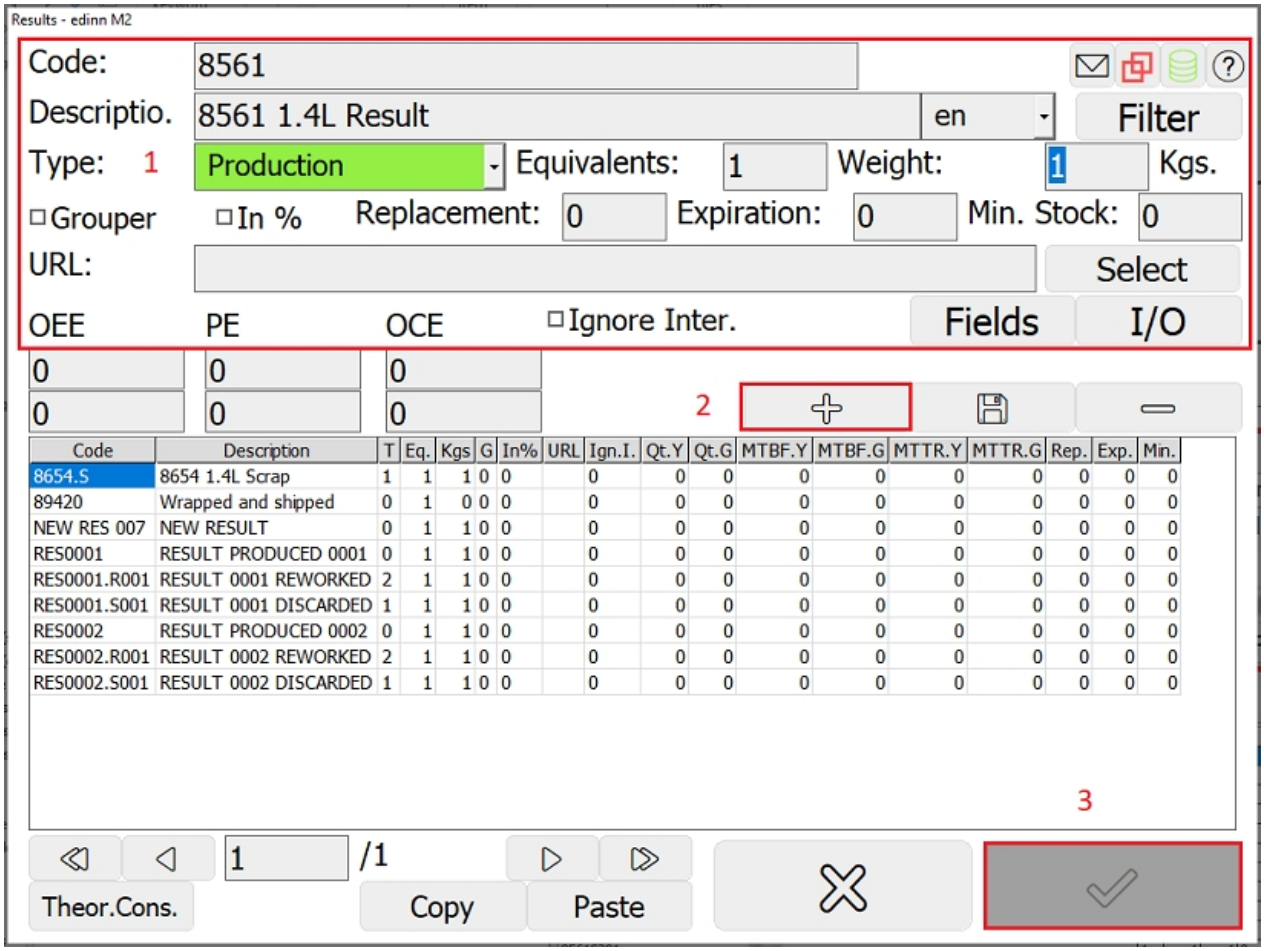
How to configure a new production result step by step

This procedure details the steps for creating a new result.

Step 1: Result Configuration

1. Click on the ⚙ button from the Main Window of the **Terminal**.
2. Select **Results** in the **CLICK TO CONFIGURE** dropdown.

Step 2: Fill-in the fields



Code	Description	T	Eq.	Kgs	G	In%	URL	Ign.I.	Qt.Y	Qt.G	MTBF.Y	MTBF.G	MTTR.Y	MTTR.G	Rep.	Exp.	Min.
8654.S	8654 1.4L Scrap	1	1	1	0	0		0	0	0	0	0	0	0	0	0	0
89420	Wrapped and shipped	0	1	0	0	0		0	0	0	0	0	0	0	0	0	0
NEW RES 007	NEW RESULT	0	1	1	0	0		0	0	0	0	0	0	0	0	0	0
RES0001	RESULT PRODUCED 0001	0	1	1	0	0		0	0	0	0	0	0	0	0	0	0
RES0001.R001	RESULT 0001 REWORKED	2	1	1	0	0		0	0	0	0	0	0	0	0	0	0
RES0001.S001	RESULT 0001 DISCARDED	1	1	1	0	0		0	0	0	0	0	0	0	0	0	0
RES0002	RESULT PRODUCED 0002	0	1	1	0	0		0	0	0	0	0	0	0	0	0	0
RES0002.R001	RESULT 0002 REWORKED	2	1	1	0	0		0	0	0	0	0	0	0	0	0	0
RES0002.S001	RESULT 0002 DISCARDED	1	1	1	0	0		0	0	0	0	0	0	0	0	0	0

1. **Basic fields to fill:**
   - **Code:** result code, generally ".S" is added to indicate a Scrap result.
   - **Description**: description or name of the result. Generally "SCRAP" is added to the description to indicate a Scrap result.
   - **Type**: to indicate if the result type is production, rework or scrap.
2. Press ✚ to insert the new result. Repeat this action for each new result to add.
3. When all the changes have been done, press the ✓ button.

NOTE: Please see Results configuration for deeper configuration.

Step 3. (OPTIONAL) PR relation

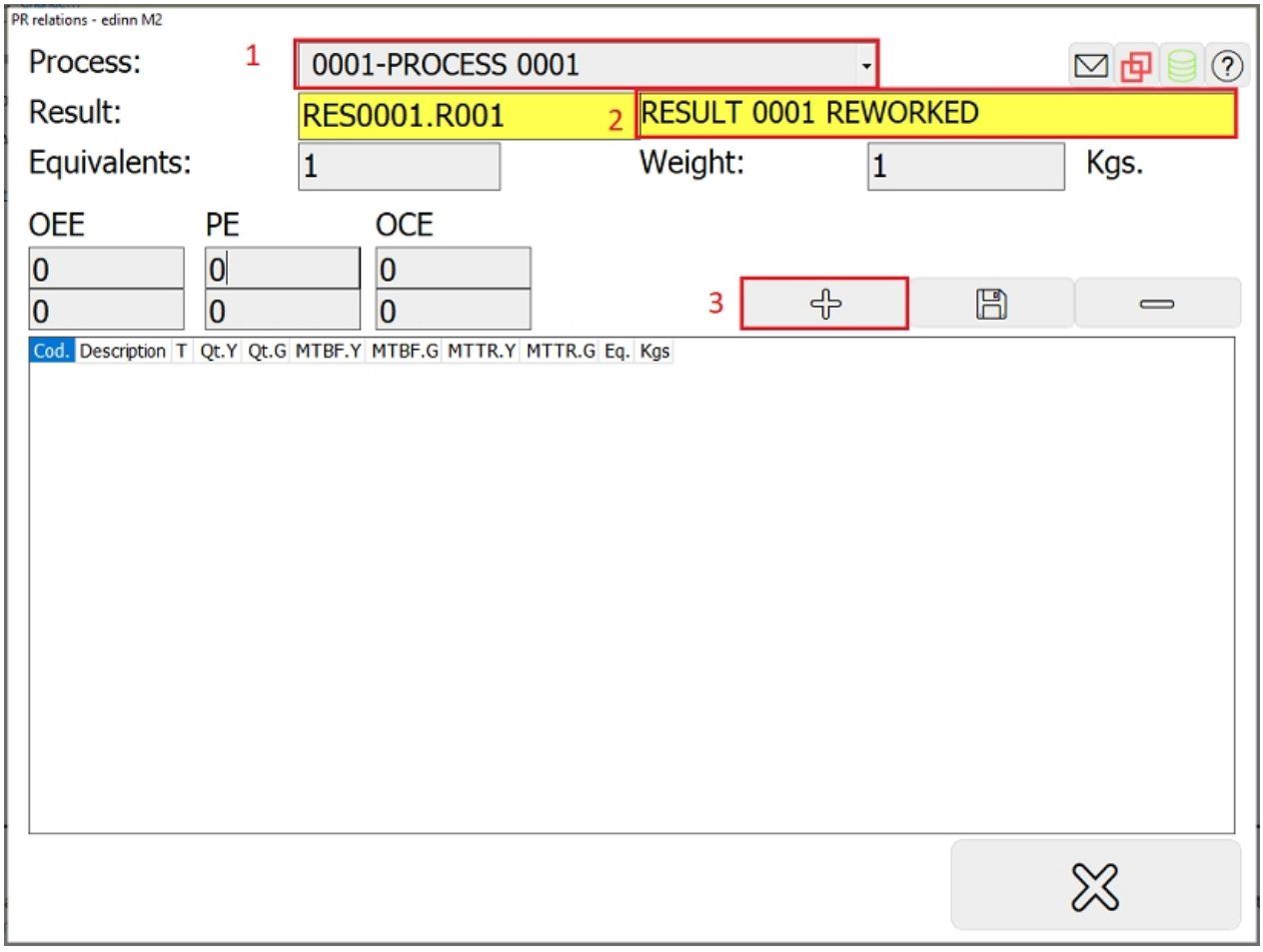
Perform this step only if you need the new result to have different equivalents, weight or objectives (in general, the parameters configurable in this window as indicated below) when produced in different processes.

1. Click on the ⚙ button from the Main Window of the **Terminal**.

2. Select **PR relation** in the **CLICK TO CONFIGURE** dropdown.

Relate the result created with a process following these steps:

1. **Process:** choose the process to which the created result belongs.
2. **Result:** press <CLICK HERE> and choose a result among the availables.
3. Press ✚ to insert the new result. Repeat this action for each new result to add.



Step 4. PSR Relation

Until this moment, you have created the result in the general directory of results and, optionally (in step 3) you have establised some particular characteristics for this result in some specific processes. However, the result cannot yet be used by the users. For the users to be able to use this result, it must be related with a process and a status.

For more information, please see particularization through relations.

To perform this step, please see the document PSR relations configuration.

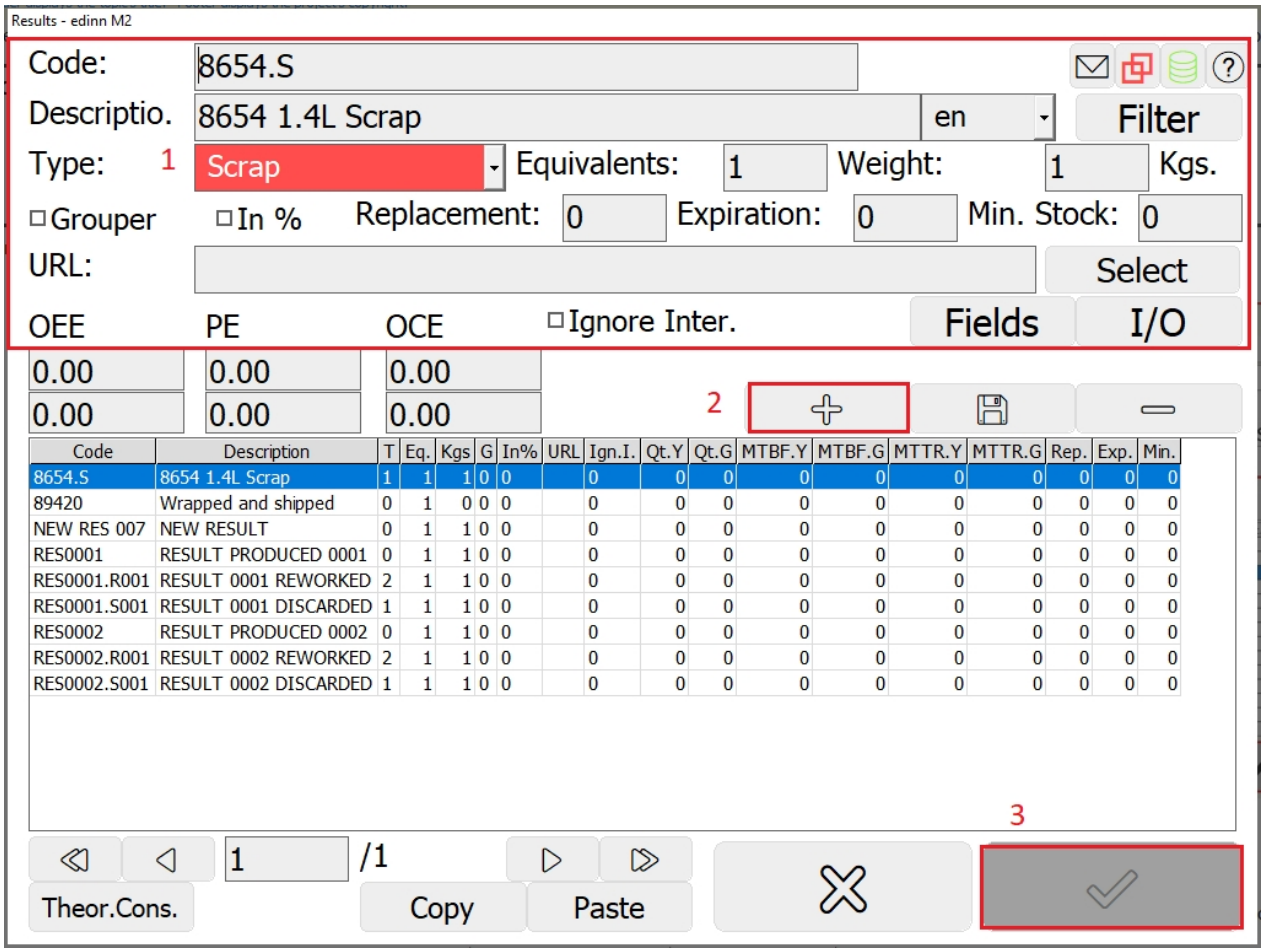
How to configure a new scrap result step by step

This procedure details the steps for creating a new scrap result. Creation of rework results is identical only changing the type of result to rework.

Step 1: Result Configuration

1. Click on the  button from the Main Window of the **Terminal**.
2. Select **Results** in the list of **CLICK TO CONFIGURE** dropdown.

Step 2: Fill-in the fields



1. **Basic fields to fill:**
   - **Code:** result code, generally ".S" is added to indicate a Scrap result.
   - **Description**: description or name of the result. Generally "SCRAP" is added to the description to indicate a Scrap result.
   - **Type**: to indicate if the result is production, a rework or scrap. It is needed to configure at least one Scrap result for each Production type.

2. Press  to insert the new result. Repeat this action for each new result to add.
3. When all the changes have been done, press the  button.

NOTE: Please see Results configuration for more information.

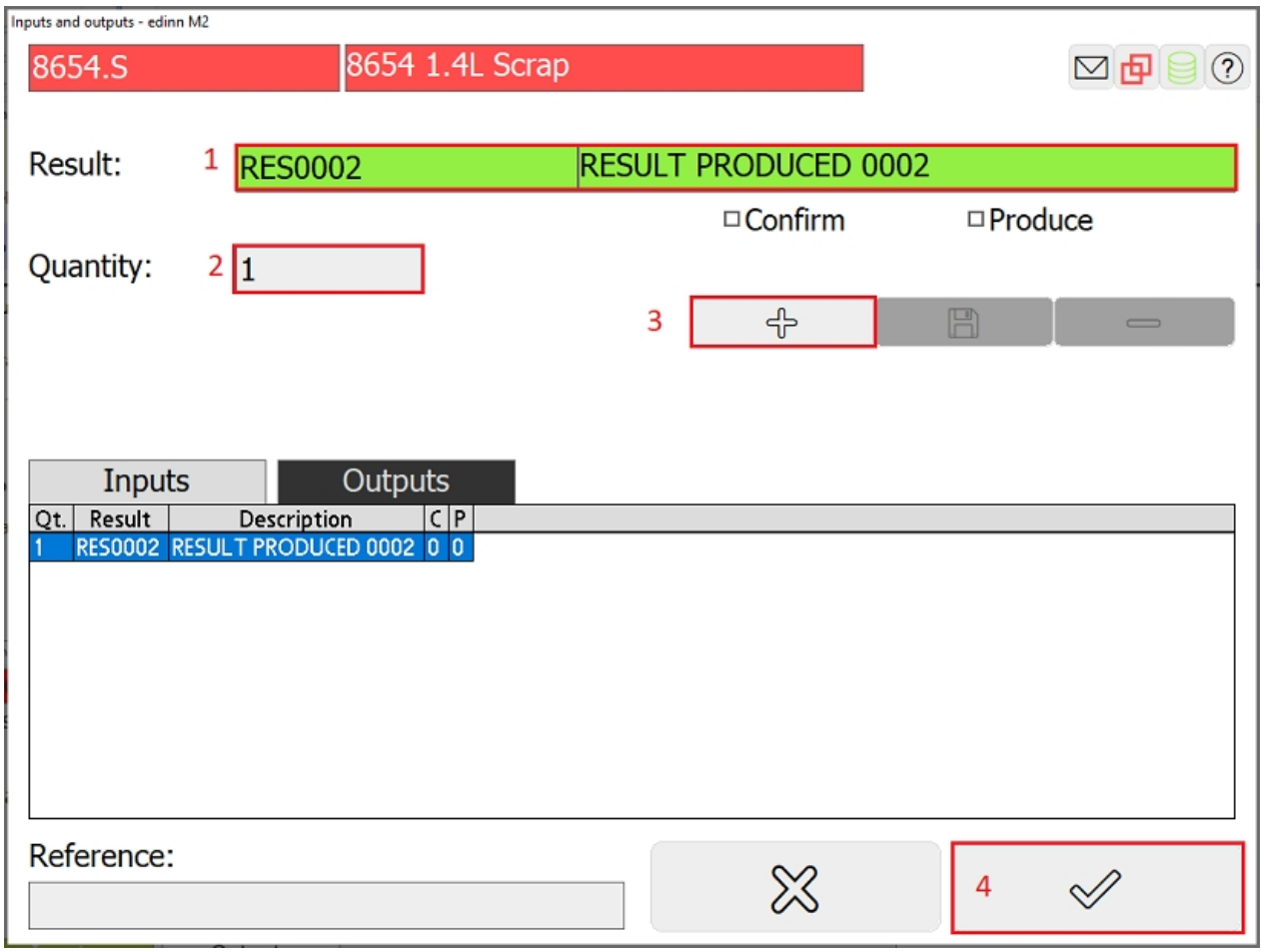
Paso 3. I/O Relation between a bad and a production result

Every scrap or rework result must be related, through inputs and outputs, with its production equivalent for the system to be able to correctly calculate the good quantity produced.

For this, **once you have create the result of Production type, the bad result (scrap or rework), select the bad result that you need to relate and click on the Inputs / Outputs button (I/O)**.

From that moment:

1. **Select** the "father" result that you need to relate the bad result with.
2. **Type** the quantity that is consumed of "father" results every time a bad result is produced.
3. **Insert** the relation in the Inputs list.
4. **Save** the change.



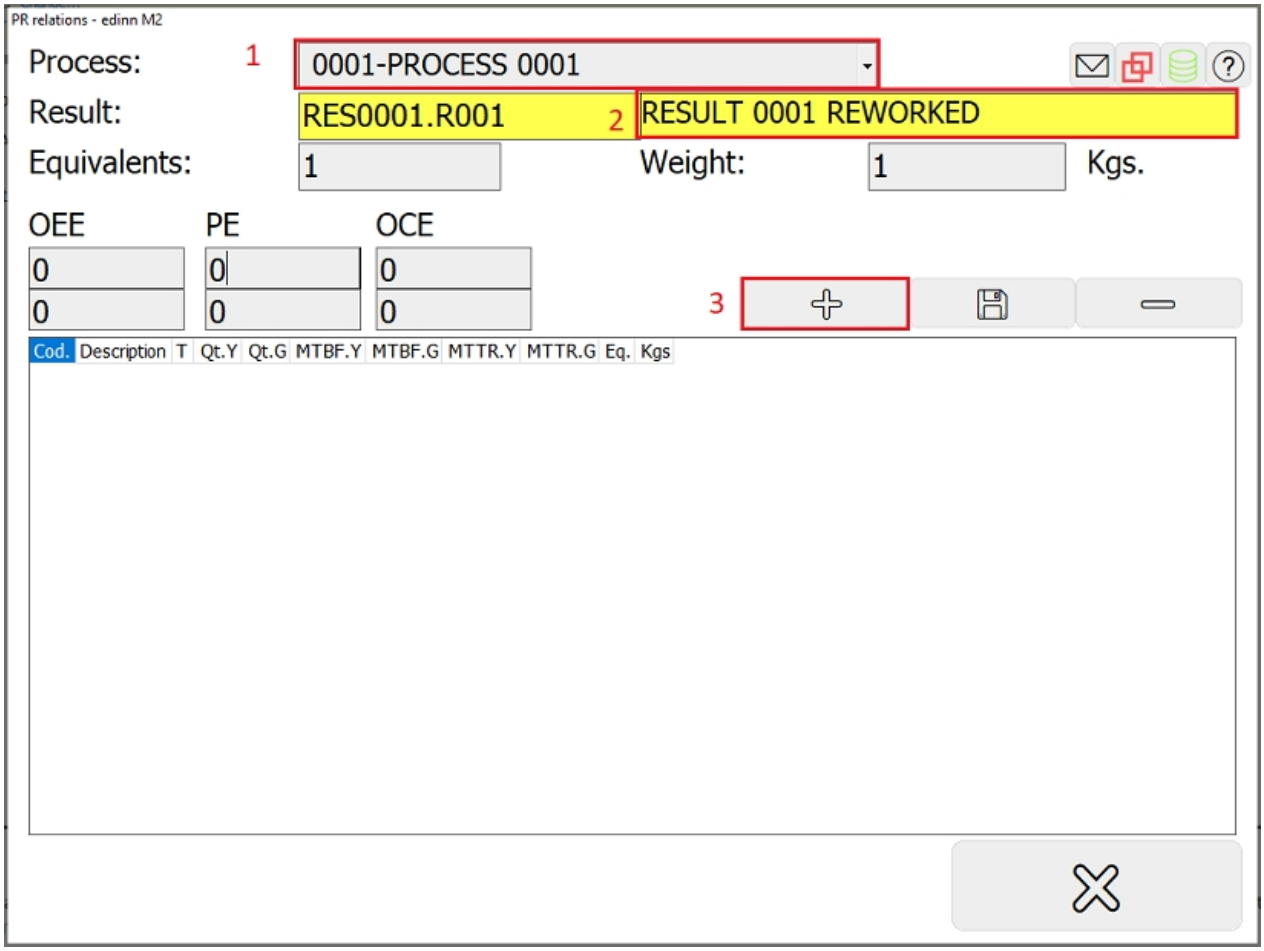
Step 4. (OPTIONAL) PR relation

Perform this step only if you need the new result to have different equivalents, weight or objectives (in general, the parameters configurable in this window as indicated below) when produced in different processes.

1. Click on the ⚙ button from the Main Window of the **Terminal**.
2. Select **PR relation** in the **CLICK TO CONFIGURE** dropdown.

Relate the result created with a process following these steps:

1. **Process:** choose the process to which the created result belongs.
2. **Result:** press <CLICK HERE> and choose a result among the availables.
3. Press ✚ to insert the new result. Repeat this action for each new result to add.



Step 5. PSR Relation

Until this moment, you have created the result in the general directory of results and, optionally (in step 3) you have establised some particular characteristics for this result in some specific processes. However, the result cannot yet be used by the users. For the users to be able to use this result, it must be related with a process and a status.

For more information, please see particularization through relations.

To perform this step, please see the document PSR relations configuration.

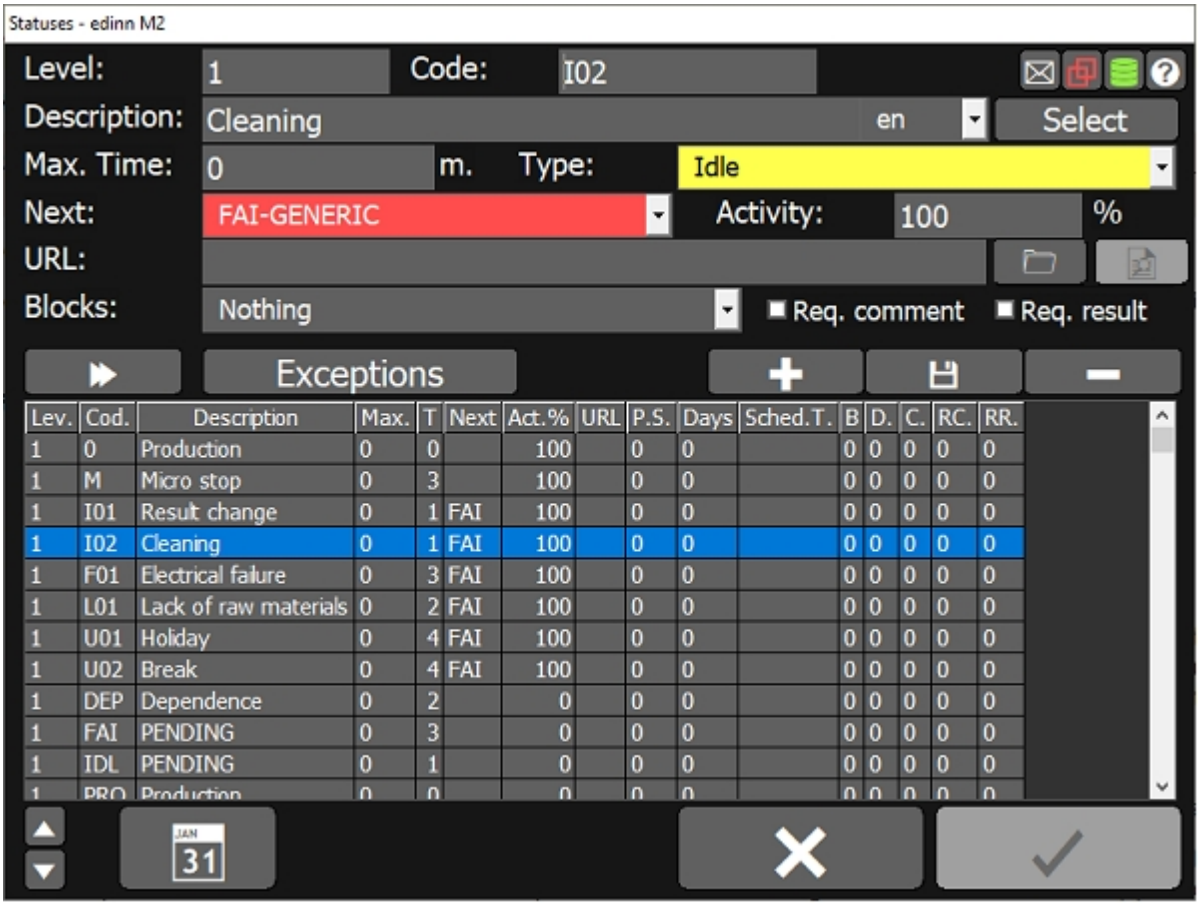
Status

Click to configure > Statuses

WARNING: Please remember that for a status to be available for a user to use it in a process, it needs to be later related in the Process-Status-Result (PSR) configuration, otherwise, by configuring statuses here we just create the status, but it cannot be used manually by a user until it is related with a specific process.

In the lower part: you will see a grid with the status already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

In the upper part: you will see the configuration of each record, where you can configure the parameters of a new insertion or edition.



CONFIGURATION PARAMETERS:

- **Level**: statuses are organized in 3 levels. These levels allow to show to the user the status in up to 3 levels. The user selects one option, and enters into the second level, selects another option, and goes to the 3rd level where he has to finally select a status.
- **Code and description** of status. Code 0 is reserved for the normal production status.
- **Language:** select the language for the description.
- **Max. Time**: maximum time in minutes that a process can be in this status. If any value different from 0 is specified, the system will automatically close this status when it is opened and the indicated minutes are surpassed. Additionally, if you try to justify an undefined status (FAI) to this status, the system will only accept the maximum justified minutes, the rest will be also required to be justified. If in the Process-Status-Result configuration, for this status, a value different than zero is indicated, the system will take that value for that process and not what is

indicated here.

- **Type**: production, failure (the working user is waiting for the process), idle time (the process is waiting for the working user), dependence (the process is on hold because it depends on another process) and not scheduled (no work is scheduled because of legal stops, holidays, etc.). Please see "What is a status?" for more information.

- **Next**: the next status to automatically pass when the maximum minutes are surpassed. If PREVIOUS is indicated, when the maximum time is reached, the process will go back to the previous status to the actual status. If the previous status is also configured as PREVIOUS in this field, previous status will be also retrieved, and so forth. This is useful to when a process is justified at a certain status on late friday, then the week-end enters automatically, and the process comes back on monday, when it reaches maximum time of the status, to the previous status (the last of friday).

  In this field, it is also possible to indicate a generic status, for example FAI, so that the user will be forced to justify it.

- **Activity**: percentage of activity that is required from a working user when a process is in this status. This is a requirement configuration for the activity report.

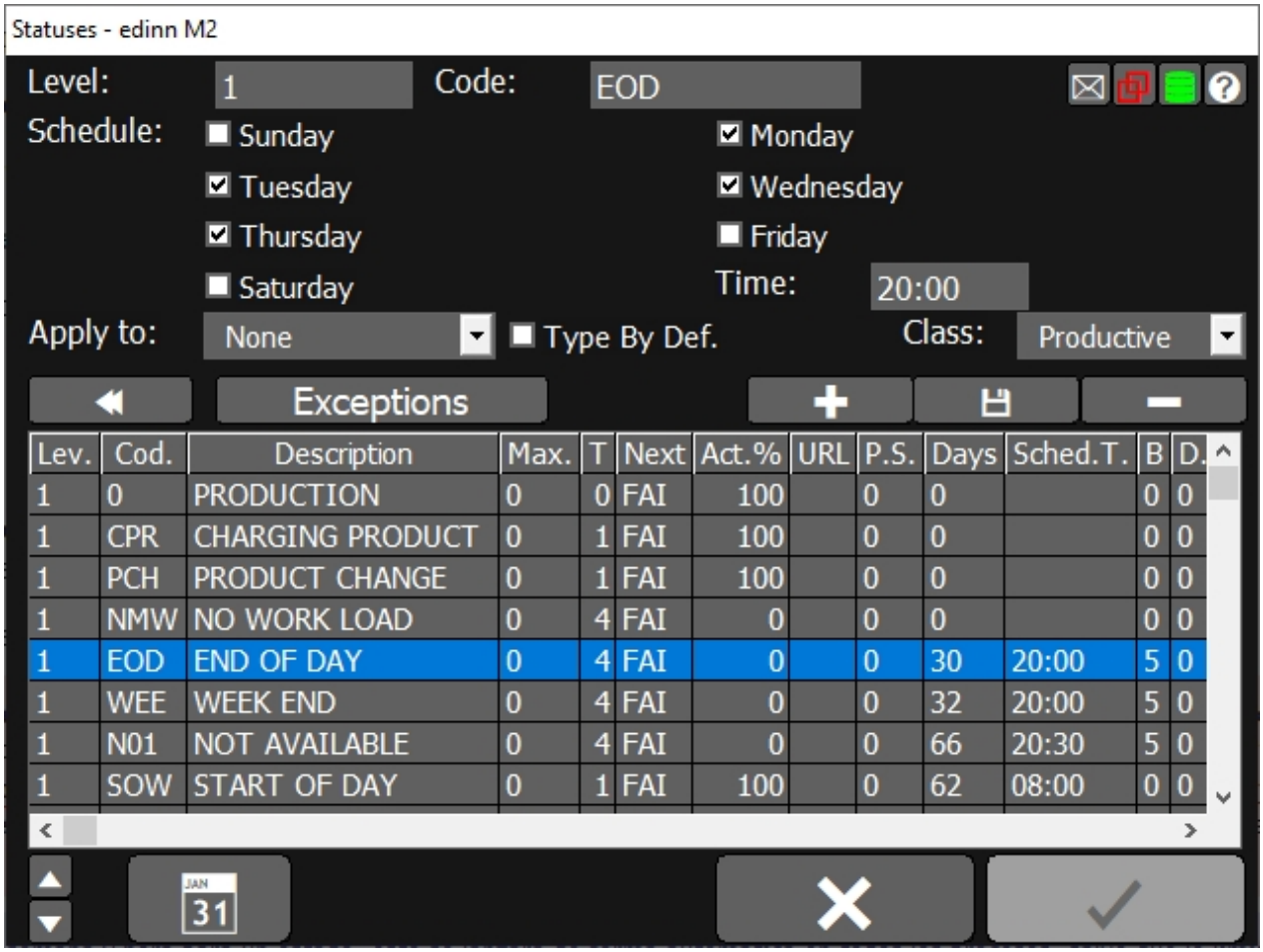
- **URL**: document or application that will be opened when the user clicks on the  that appears after having clicked on the status, on the status window. Please see how to configure URLs. URLs can be selected from  and its syntax can be cheched from the button  .

- **Block**: if the status should block any other statuses, **as long as they have not been generated yet**, and optionally results. Options are:

  - Nothing: no blocking.
  - Stops: this status will block all not productive statuses coming later.
  - Production: this status will block all productive statuses coming later.
  - Statuses: this status will block all statuses coming later.
  - Results: this status will block all results coming later.
  - Results & Stops: this status will block all results and not productive statuses coming later.
  - Results & Production: this status will block all results and productive statuses coming later.
  - Results & Statuses: this status will block all results and statuses coming

later.

- **Req. comment:** to use the status a mandatory comment will be required.
- **Req. result**: to use the status it will check that no more time than Max Cycle Time of the process has passed since the last result of advance or since the current working order was started. This is useful to guarantee that users inform of results advance before, for example, indicating that they have finished the day. Only applies if the user does not have any associated process or if, having it, the user is operating on his/her associated process.
- **Exceptions button:** please see exceptions.



-  shows more options.
  - **Schedule**: status can be automatically scheduled by days of the week and time. This allows, for example, to close days or weeks with END OF PRODUCTION and start them up when needed.
  - **Time**: time at which the status will automatically be inserted.

WARNING: If you schedule an unscheduled or dependency status, for example for the weekend, and you do not schedule an status for availability to be automatically launched after the previous, then you will be telling to the automatic scheduler that your processes are never available to work, and the

automatic orders scheduler will schedule the orders as late as it can, which normally will be the date of today plus the days indicated in the Real time oldest record in the Console. Therefore, if you want to use the automatic scheduler of orders, following with the previous example where you scheduled statuses, of type unscheduled or dependency, to indicate that the weekend is unscheduled for work, then you must schedule a status of any type different to unscheduled or dependency on monday at the start of the shift.

NOTE: Please take into account that if you program days of the week and time for a status, it will be automatically launched in all processes, except if you limit that through exceptions or with a different configuration for this status and a certain process in the configuration of PSR Relations, if that relation exist as it is not mandatory if this status is not going to be used manually by any user.

Therefore:

1) If you want a status to be available for the use of the users: you must create the PSR Relation with that process. In this case, if additionally that status must be automatically launched by a schedule, then you could configure that in the PSR relation.
2) If you do not want a status to be available for the use of the users, you must not create a PSR relation with any process; in this case, if additionally that status must be automatically launched in certain processes by a schedule, then you must use the exceptions explained here.

PSR relations and exceptions can be combined.

- **Apply to:** options are "Work start" and "Work end" [under development]. This mark tells the system to automatically generate a new status to indicate that a new work has started or finished, every time a new result with 0 quantity is inserted (for Work start), if at the process configuration the option Gen W. Start has been marked for the process. Only one status should be marked with this option.
- **Type by def.**: indicates to the system that this status is the default for the type. Please see "What is a status?" for more information.
- **Class**: allows to identify the type or class of the status, according to the edinn classification for statuses. This is useful to compare statuses from different plants with different configurations of statuses. Possible classes, with an explanation where necessary, are as follows:
  - **Audit**: the process cannot work as it is being audited.
  - **Corrective Maintenance**: the process cannot work as it is in corrective maintenance.
  - **External dependency**: the process cannot work as it depends on an element or process external to the organization.
  - **Failure**: the process cannot work as it is in a failure.
  - **Holiday**: the process cannot work as the persons which operate it are in holiday.

- **Internal dependency**: the process cannot work as it depends on an element or process internal to the organization.
- **No workload**: there is not workload for the process at the moment.
- **Ordering and Cleaning**: the process cannot work as it is being ordered and cleaned.
- **Predictive Maintenance**: the process cannot work as it is in predictive maintenance.
- **Preventive Maintenance**: the process cannot work as it is in preventive maintenance.
- **Productive**: the process is working.
- **Quality test**: the process cannot work as it waiting for a quality test.
- **Result change**: the process cannot work as it is in result change.
- **Setup**: the process cannot work as it is being prepared (setup) for a new order.
- **Teardown**: the process cannot work as the order is being finished.
- **Tool change**: the process cannot work because any of its tools is being changed.
- **Training**: the process cannot work as the persons which operate it are in training.
- **Vacation**: the process cannot work as the persons which operate it are in vacation.
- **Weekend**: the process cannot work as the persons which operate it are in a weekend.

-  with these buttons you can change the statuses position by the order you want them to appear at the status window. As you can see in the image above, these statues are ordered the same way they appear at the status window.
-  Shows the calendar calculated by the system based on the statuses configuration or the statuses which really occured.

How to configure a status step by step

In [**Click to configure**] select [**Statuses**]. To create and configure results, complete the following configuration.

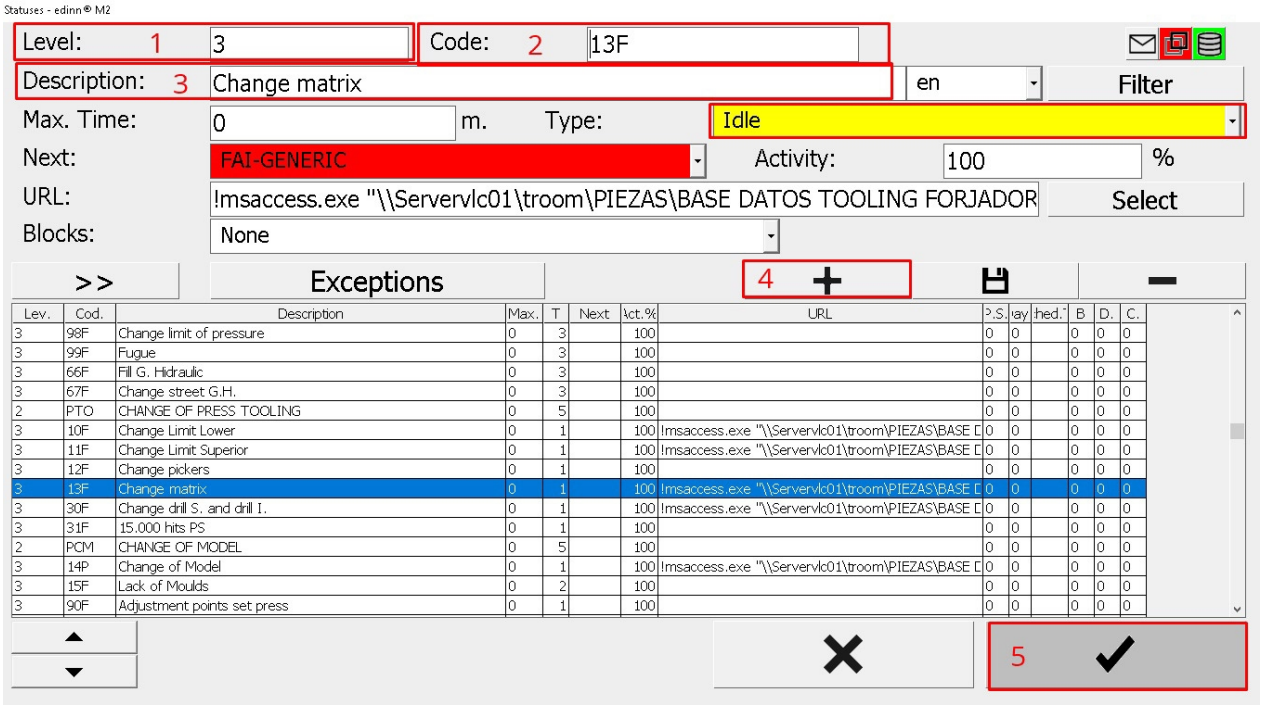
Step 1. List all the items

Lists all the **statuses** through which your machine can pass and its **types**, e.g:

- **Production.**
- **Idle**: process cleaning, product change, quality tests...
- **Stops/Failure**: motor failure, excessive pressure...
- **Dependency**: dependence on another process, lack of stock...
- **Not scheduled**: break, weekend...
- **Title**: agrupator status, e.g: to put all the type of stops toghether.

Step 2. Enter the list in edinn

1. **Level:** indicate the level of the status (see Statuses organization using levels)
2. **Code & Description:** write a status code (e.g: STAT1) and a description (e.g: Quality test).
3. **Type**: indicate the type below (step 1) defined.



Step 3. Save changes

4. Press  to insert in the table the configuration of the status. Repeat this action for **each** status to add.

5. When all the changes have been done, press the  button.

NOTE: Please see Status configuration for deeper configuration.

Statuses organization using levels

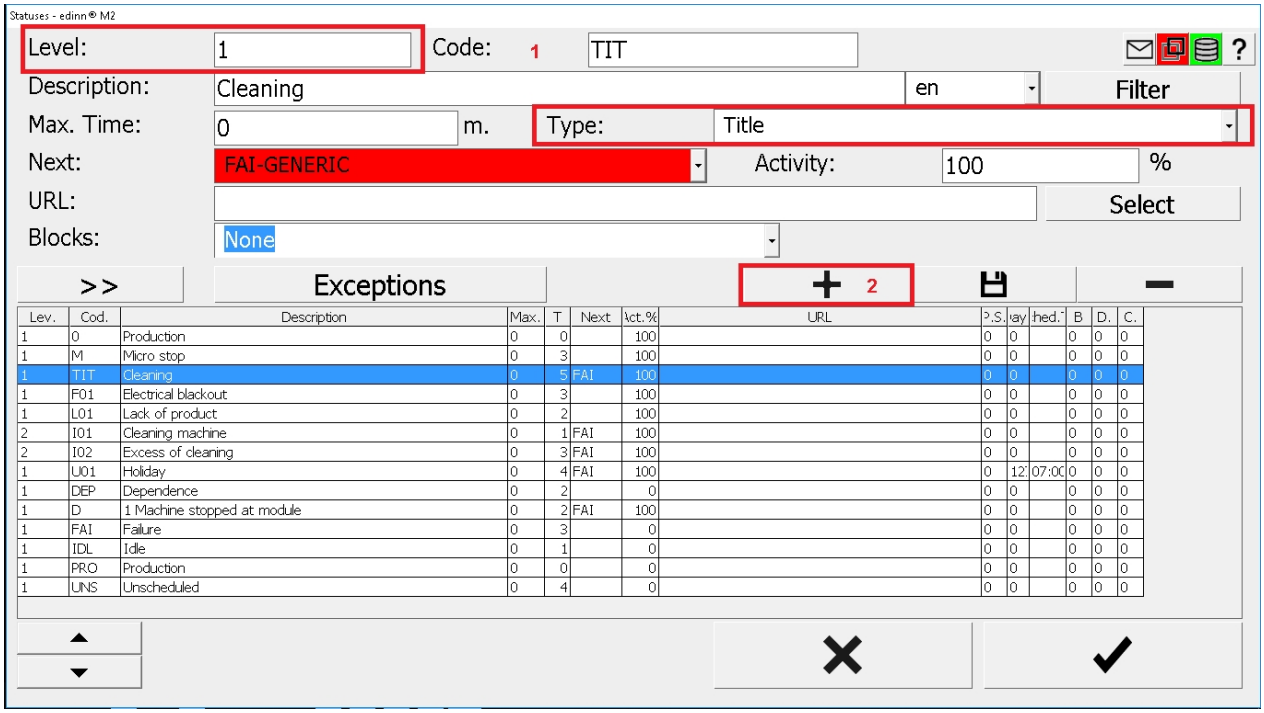
In [**Click to configure**] select [**Statuses**]. Levels allow to organize statuses grouping them by desired criteria. Thus, different statuses may belong to the same category. This has to be headed by a status of the type *Title*.

To organize statuses in different levels, please follow these steps:

Step 1. Create or select a status of the type *Title*

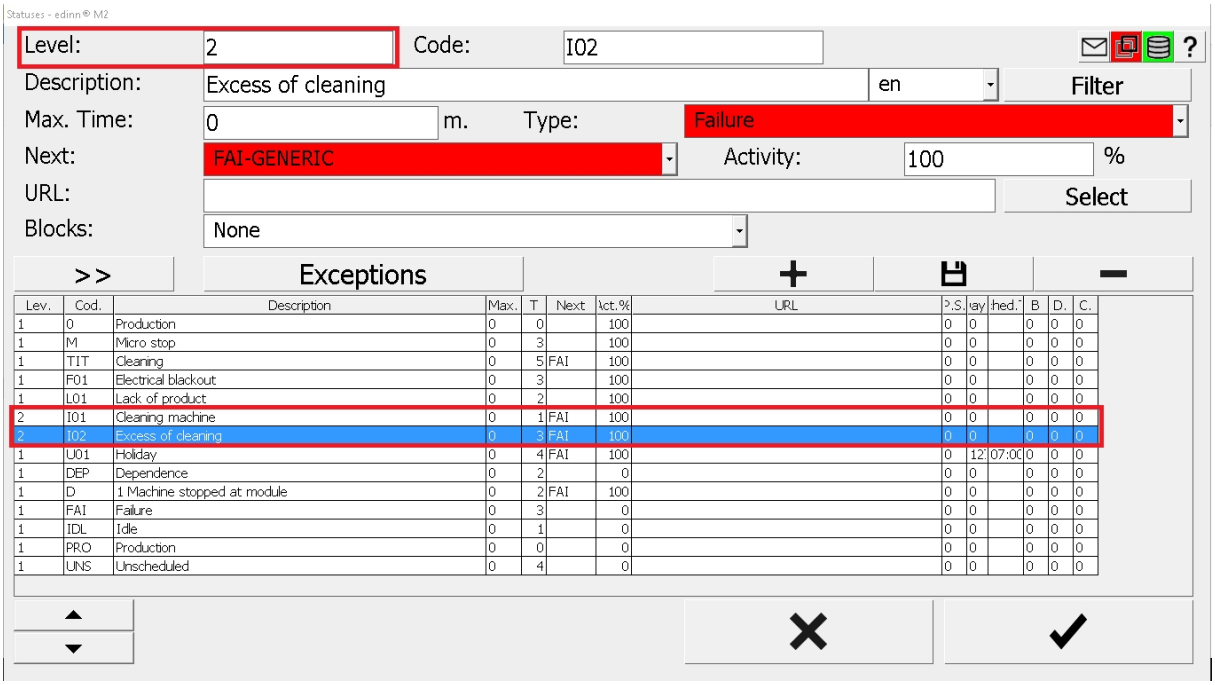
This will be the container status, that is why it has to be **level:** *1* and **Type:** *Title*. Once you fill the necessary fields, press the Add button () to insert the new status in the table.

In the following example, we created a status called *Cleaning*. This status is meant to group different statuses related to the cleaning of the machines.

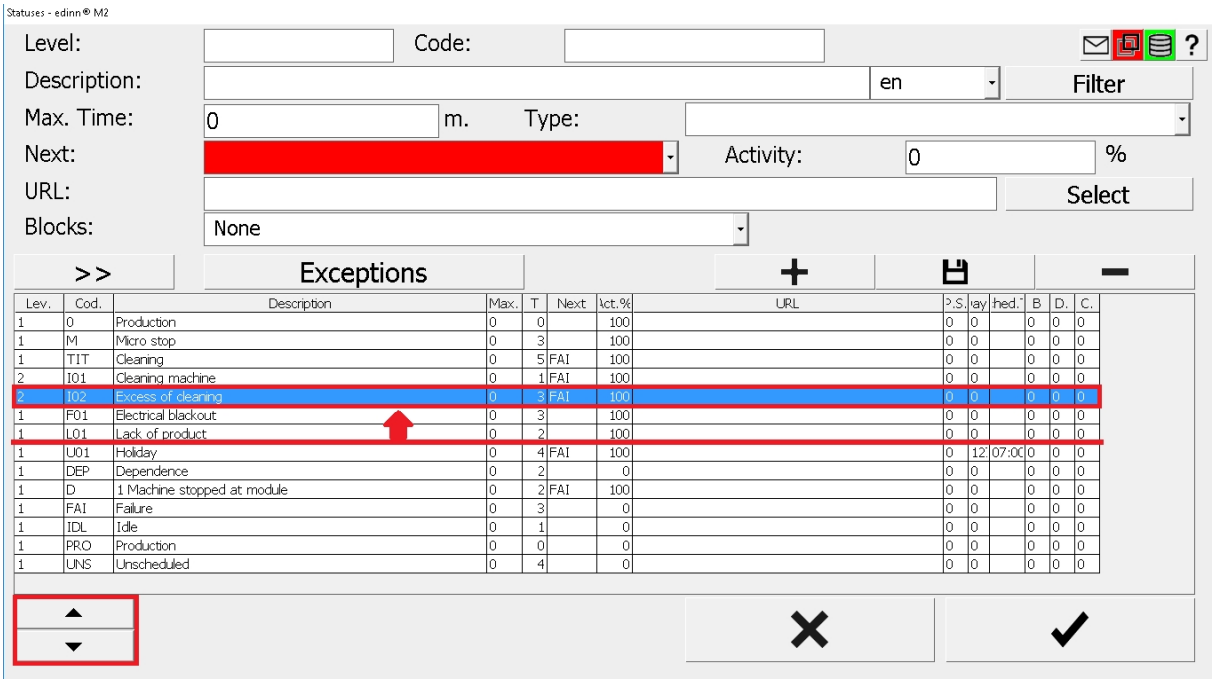


Step 2. Configure the subordinate statuses

1. **Level:** For each status, indicate its level. You can configure levels from 1 to 3. In the following example we selected level 2 for the statuses *Cleaning machine* and *Excess of cleaning*, as we want to subordinate those statuses to the status *Cleaning*.



2. Put each subordinate status on the positions below the title. To do so, select the status on the table and use the up and down arrow buttons located at the bottom left part of the screen.



How to create a new status step by step

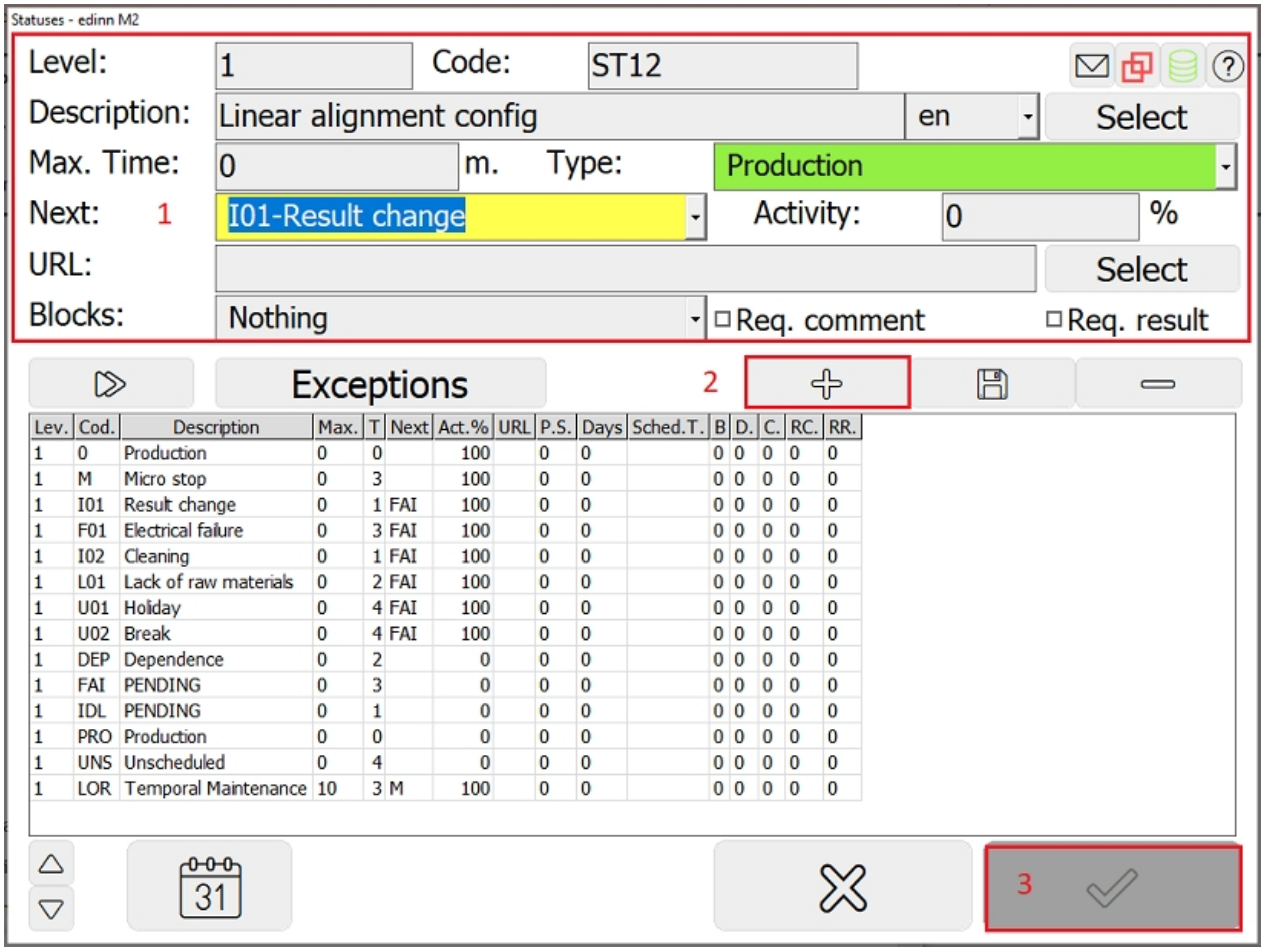
This procedure details the steps for creating a new status.

WARNING: Please remember that for a status to be available for a process, it needs to be later related in the Process-Status-Result (PSR) configuration, otherwise, by configuring statuses here we just create the status, but it cannot be used until it is related with a specific process. For the procedure, please see: relations PSR

Step 1: Status Configuration

1. Click on the ⚙ button from the Main Window of the **Terminal**.
2. Select **Statuses** in the **CLICK TO CONFIGURE** dropdown.

Step 2: Fill the fields



1. Fill the fields to configure the new status:
    - **Description:** write a name for the status, for example: Dependency1.
    - **Code:** write an abbreviation or code for the status, for example: D01.
    - **Type:** select whether the status is a stop, a dependency, a title, etc.
        - Stop and Idle do not affect the OEE.
        - No scheduled and Dependency affect the OEE.
        - Title is a status type used as a status agrupator. For example: to put all the type of stops toghether.
    - **Level:** indicate the level of the status in a tree structure. For example: if we configure DEPENDENCY as a "Title" and Level 1, the status that we want to be in this category will have the level 2.
    - **Etc.**
2. Press ✚ to insert the new status .Repeat this action for each new status to add.
3. When all the changes have been done, press the ✓ button.

How to modify a status step by step

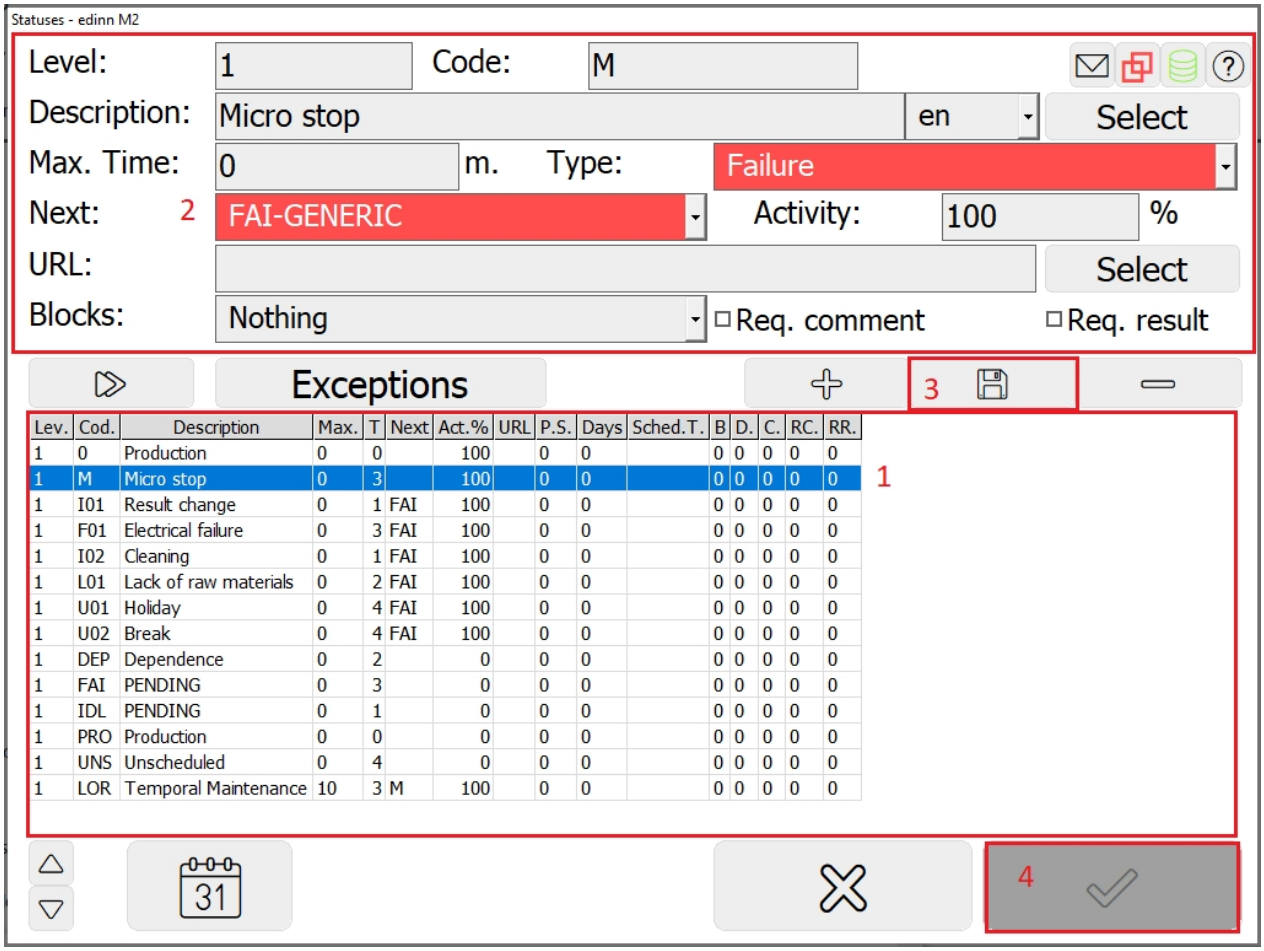
This procedure details the steps for modifying a current status.

WARNING: Please remember that for a status to be available for a process, it needs to be later related in the Process-Status-Result (PSR) configuration, otherwise, by configuring statuses here we just create the status, but it cannot be used until it is related with a specific process. For the procedure, please see: relations PSR

Step 1: Status Configuration

1. Click on the ⚙ button from the Main Window of the **Terminal**.
2. Select **Statuses** in the **CLICK TO CONFIGURE** dropdown. You will see the following window:

Step 2: Change the fields



1. Select the status that you want to modify.

2. Change any of the fields.
3. Once modified, press the  icon.
4. When all the changes have been done, press the  button.

How to schedule statuses

Introduction

The status can be planned to be launched on certain dates and/or at certain times, this can be configured from two configuration windows:

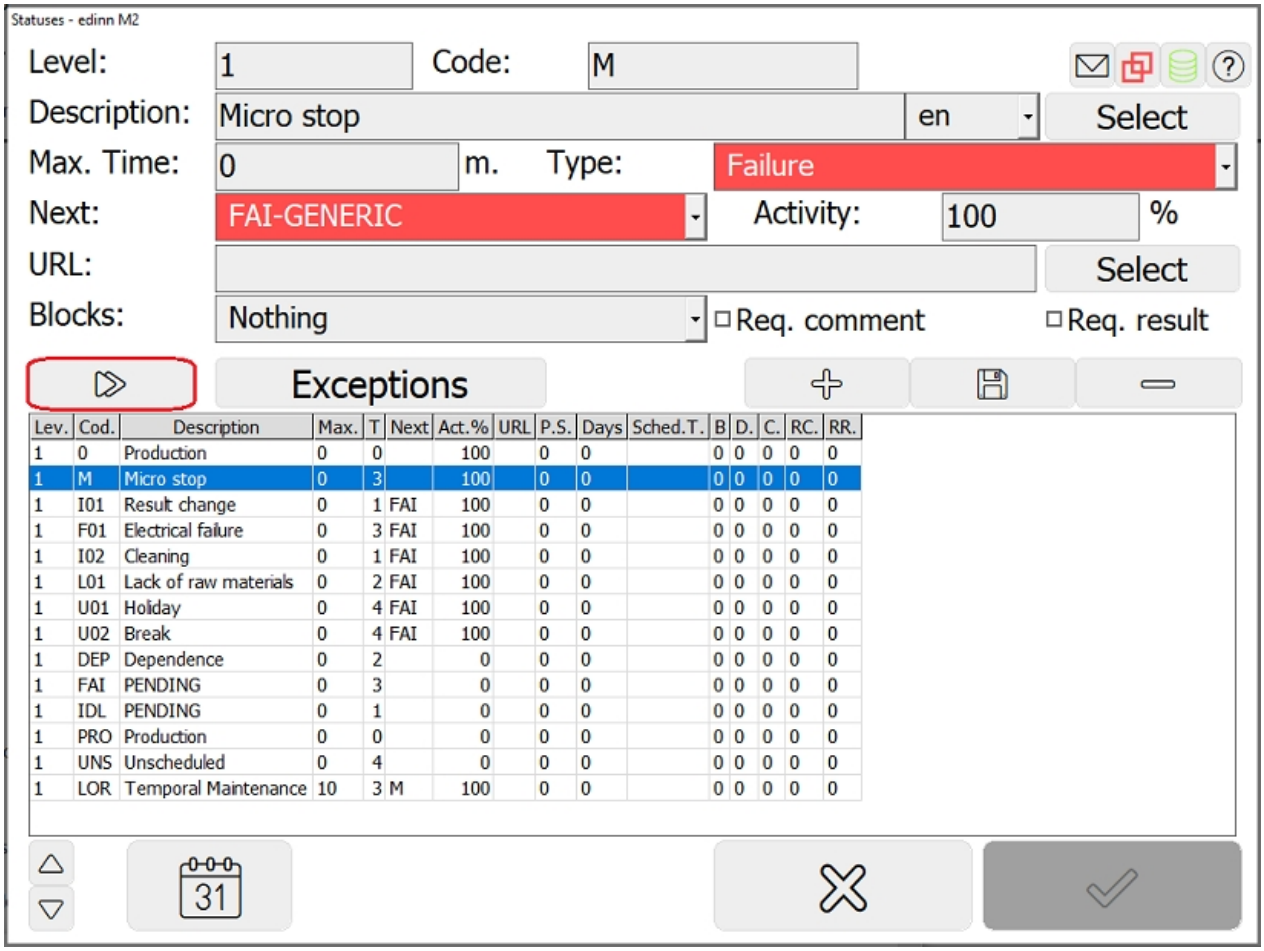
- In "**Status configuration**": to schedule the statuses for all the proceses.
- In "**PSR relations**": to schedule the statuses only for certain process or processes.

For all the processes

In this procedure, the steps to schedule statuses **for all the processes** will be detailed:

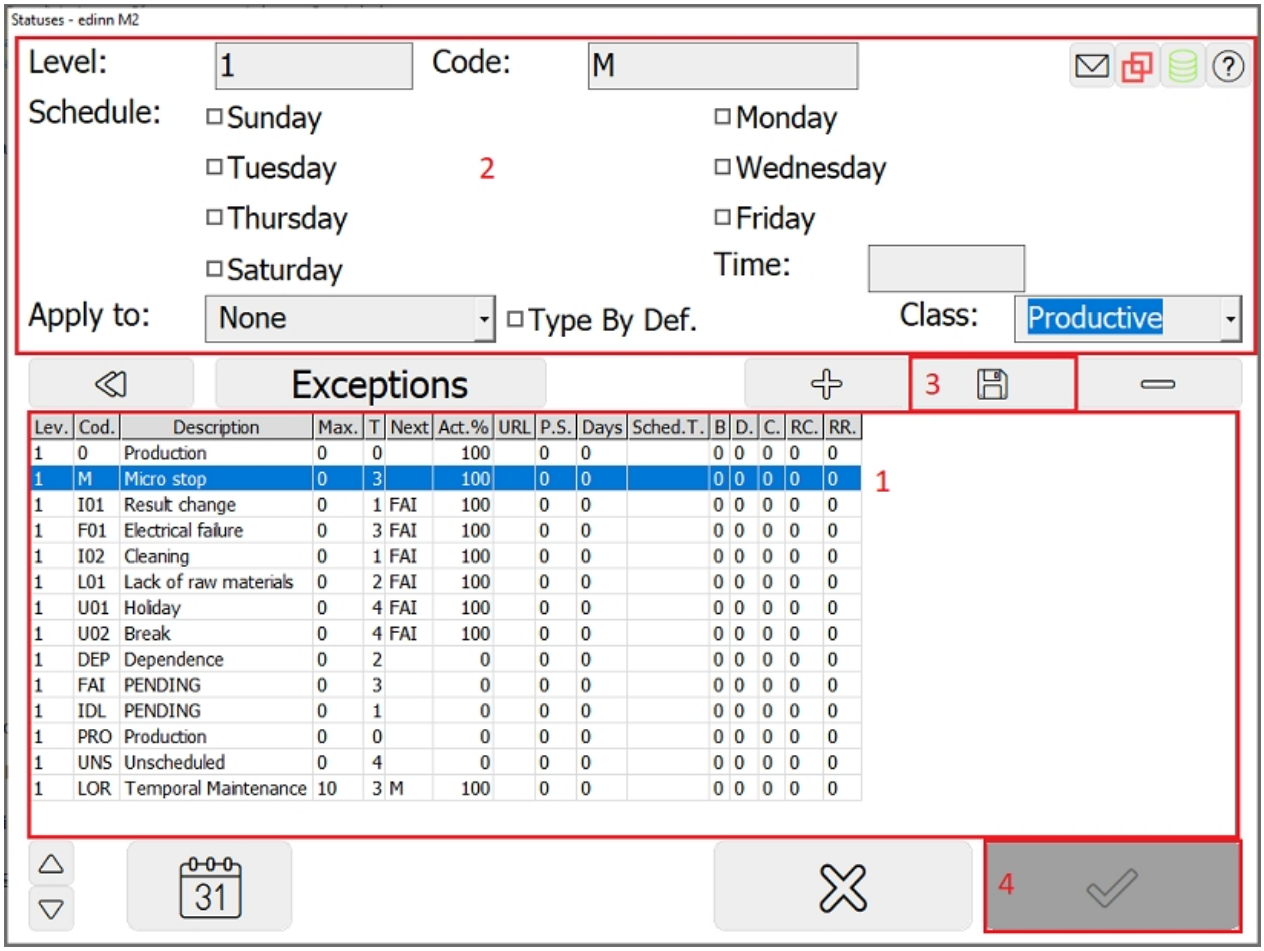
Step 1: Status Configuration

1. Click on the button from the Main Window of the **Terminal**.
2. Select **Statuses** in the **CLICK TO CONFIGURE** dropdown.
3. Press the  button to show advanced configuration:



Step 2: Fill the fields

In this window we can define the necessary parameters for the automatic launch of statuses. Follow the following steps:



1. Select the status to schedule from the table.
2. Select the days and time for the status to be launched.
3. Press the **[Mod]** (modify) button to insert the new changes.
4. Press the **[Accept]** button to save all the changes made.

Examples:

- Status “WEEKEND”: launched on Fridays at 20:00.
- Status "END OF THE DAY”: launched on Monday, Tuesday, Wednesday and Thursday at 20:00.

WARNING: take care to not plan two status for the same period.

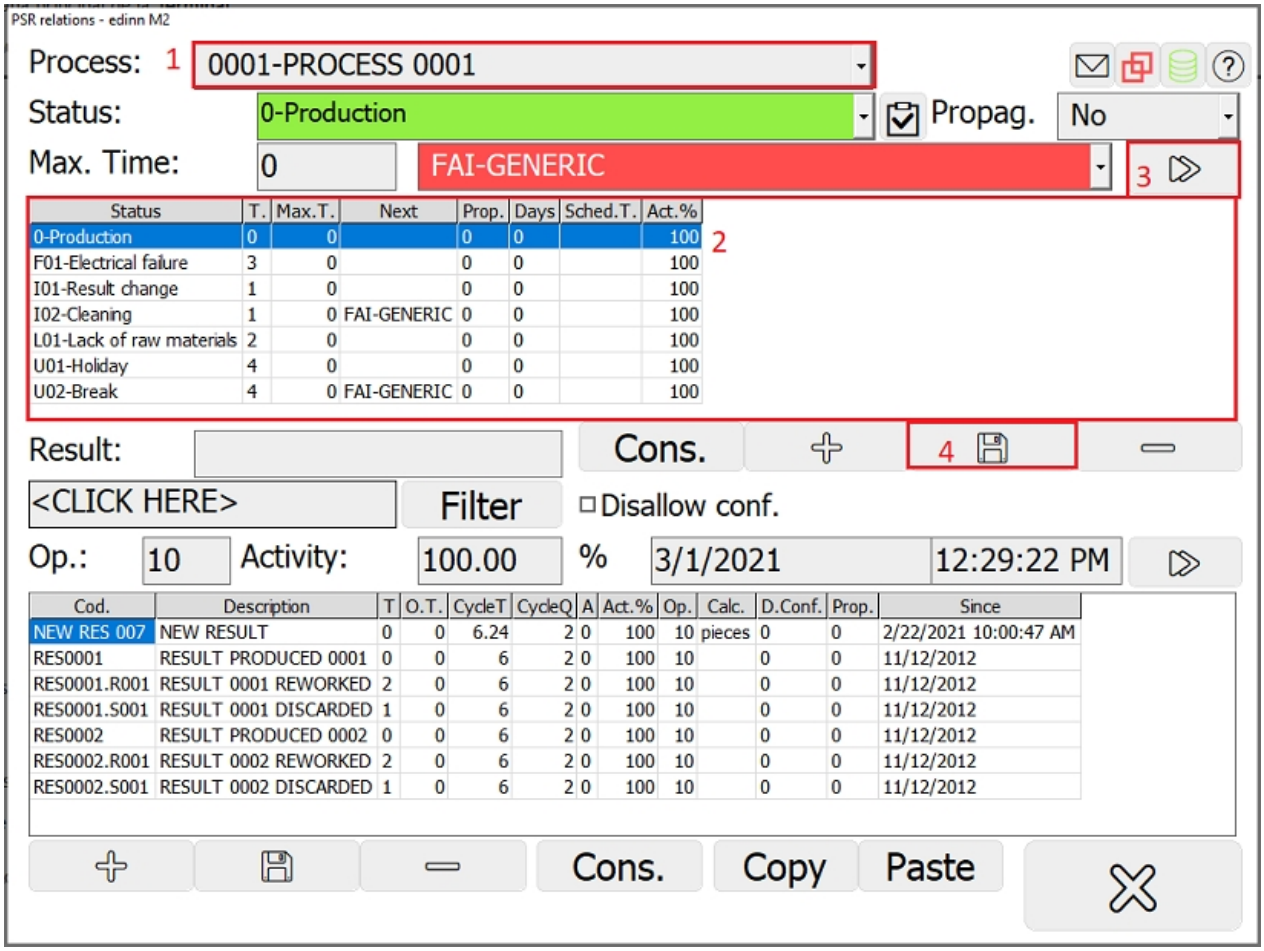
For specific processes

In this procedure, the steps to schedule statuses **for a specific process** will be detailed:

Step 1: PSR relations Configuration

1. Click on the ⚙ button from the Main Window of the **Terminal**.
2. Select **PSR relations** in the **CLICK TO CONFIGURE** dropdown.

Step 2. Fill the fields



1. Select the process for which you want to configure the status.
2. Select the status to schedule from the table.
3. Press the button to show advanced configuration:
4. Select the days and time for the status to be launched.
5. Press the 💾 button to insert the new changes.

Examples:

- Status “WEEKEND”: launched on Fridays at 20:00.
- Status "END OF THE DAY”: launched on Monday, Tuesday, Wednesday and Thursday at 20:00.

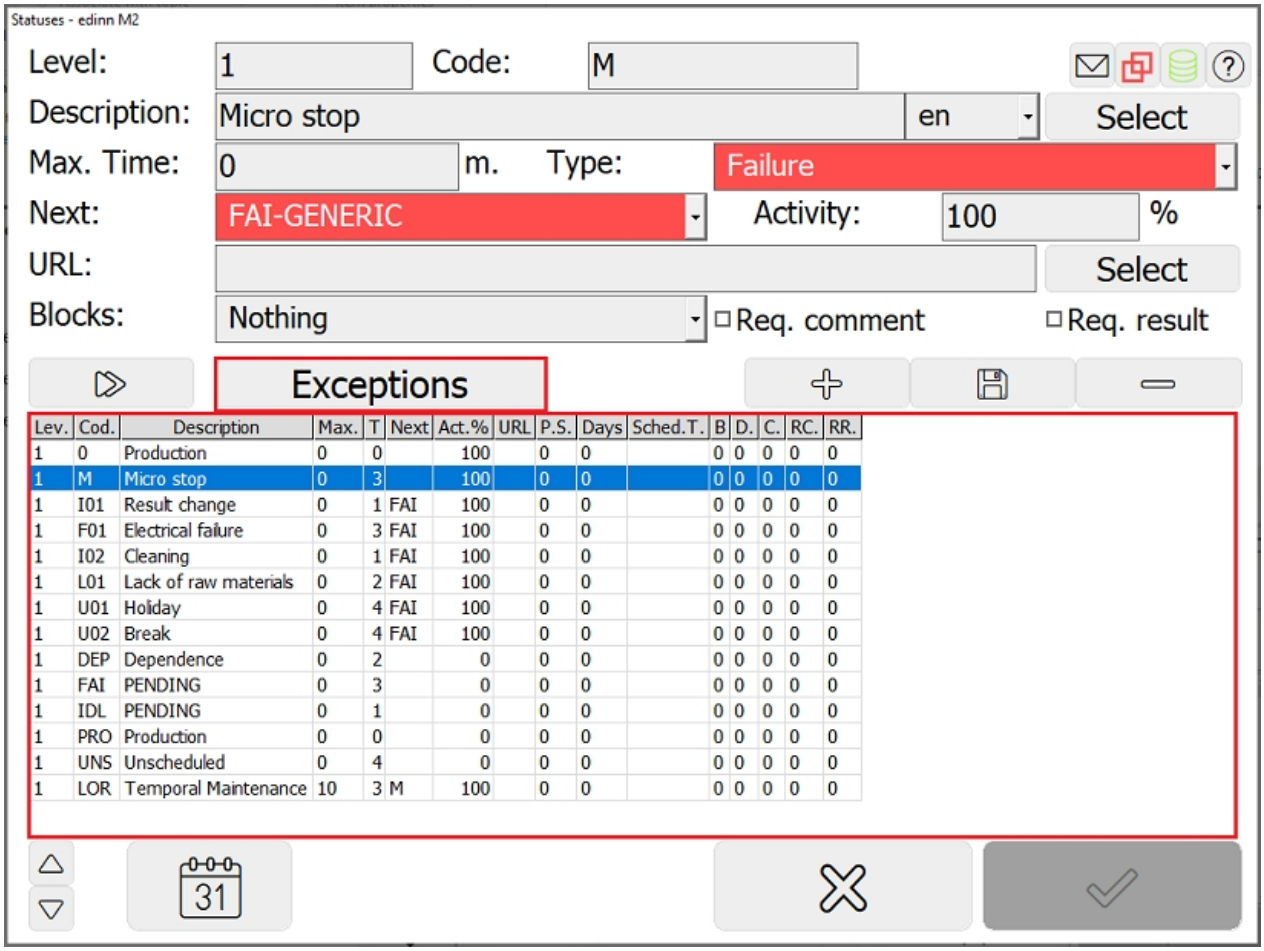
WARNING: take care to not plan two status for the same period.

How to configure exceptions

In the **Status configuration**, we can configure **Exceptions**, which allow to skip the automatic launch of scheduled status or autocontrol tasks, or to force that they are launched at a specific moment.

Step 1: Status Configuration

1. Click on the  button from the Main Window of the **Terminal**.
2. Select **Statuses** in the **CLICK TO CONFIGURE** dropdown.
3. Select a status for which you want to configure the exceptions.
4. Press the "**Exceptions**" button.



Step 2: Fill the fields

1. The exceptions can be programmed to be executed according to different parameters:
    - **Depending on the date and/or time**: the date and/or time when the exception will occur. We can use zeros as a wildcard. Examples:
        - Indicating 00/08/0000 means every august of every year.
        - Indicating 01/05/0000 means every May 1st of every year.
    - **Status:** this field is useful to avoid the launch of an autocontrol task when a process is stopped on a certain status.
    - **Process:** for indicate a process where the launching of a certain status has to be avoided.
    - **Active order**: indicate here if you want the status to be avoided or not when there is an active order.
    - **Include**:
        - Mark it: to treat the exception as inclusive = do it only for this case.
        - Do not mark it: to treat the exception as exclusive = do it for all the cases except for the one configured.
2. Press  when the fields are filled to insert the exception.
3. Press  button to save all the changes made.

Process-Results (PR)

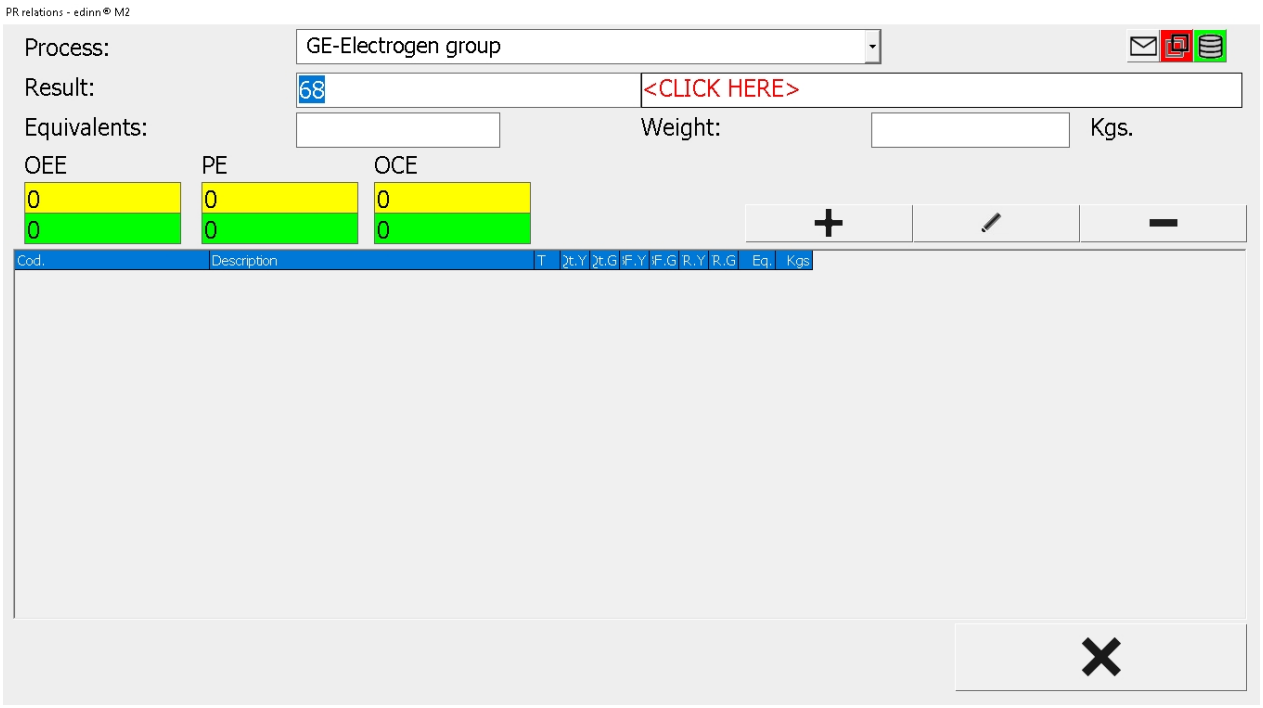
Click to configure > PR relations

Process-Results relations mean the specific information needed when a process is producing a specific result. This is useful when it is needed to have certain characteristics (the ones present on this window) specific for certain results.

> NOTE: for more information please see particularization through relations.

In the lower part: you will see a grid with the relations between Processes and Results already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

In the upper part: you will see the configuration of each record, where you can configure the parameters of a new insertion or edition.



CONFIGURATION PARAMETERS:

- **Process:** for which the **result** is related.
- **Code and description** of the related result.
- **Equivalents and Weight**: characteristics of results configuration, that can be specifically changed when the result is being generated on a certain process.
- **Ratios configuration (OEE, PE, OCE)**: green and yellow targets for the OEE and PE (is the same as the OEE but without considering idle statuses) and OCE ratios for the areas. By clicking any of these fields, more targets will appear: please see Ratios configuration.

How to configure a Process-Result (PR) step by step

In [**Click to configure**] select [**PR relations**]. To **relate** processes with specific results, complete the following configuration.

Step 1. List all the items

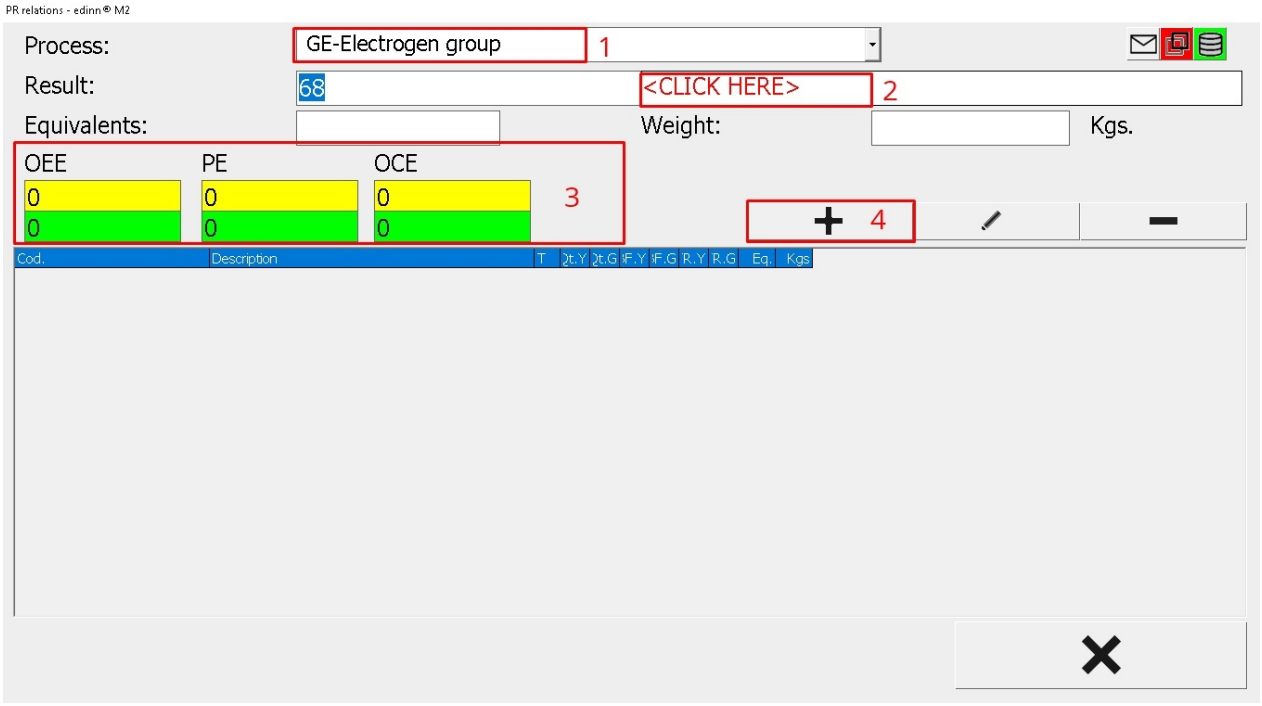
List all the **results** that belong to each **process** previously configured, e.g: Machine1 produces: Prod1, Prod2 & Prod3.

Step 2. Enter the list in edinn

1. **Process:** configure process by process, of those available in the dropdown.

2. **Result:** enter a record for each relationship between the process and a result selected in <CLICK HERE>

3. Configure the OEE for the relation if needed.



Step 3. Save changes

4. Press ✚ to insert in the table the configuration of the status. Repeat this action for **each** status to add.

> **NOTE:** Please see PR configuration for deeper configuration.

Process-Status-Result (PSR)

Click to configure > PSR relations

Process-Status-Results relations is the configuration to relate results with statuses and with processes. The results and statuses that have not been related with processes, cannot be used by users nor the system will not take them into account in certain functionalities.

In this window, for each status and result, it is possible to assign different or the same characteristics of the status configuration and the results configuration. This is useful when it is needed to have certain characteristics (the ones present on this window) specific for certain processes, statuses and results. For more information please see particularization through relations.

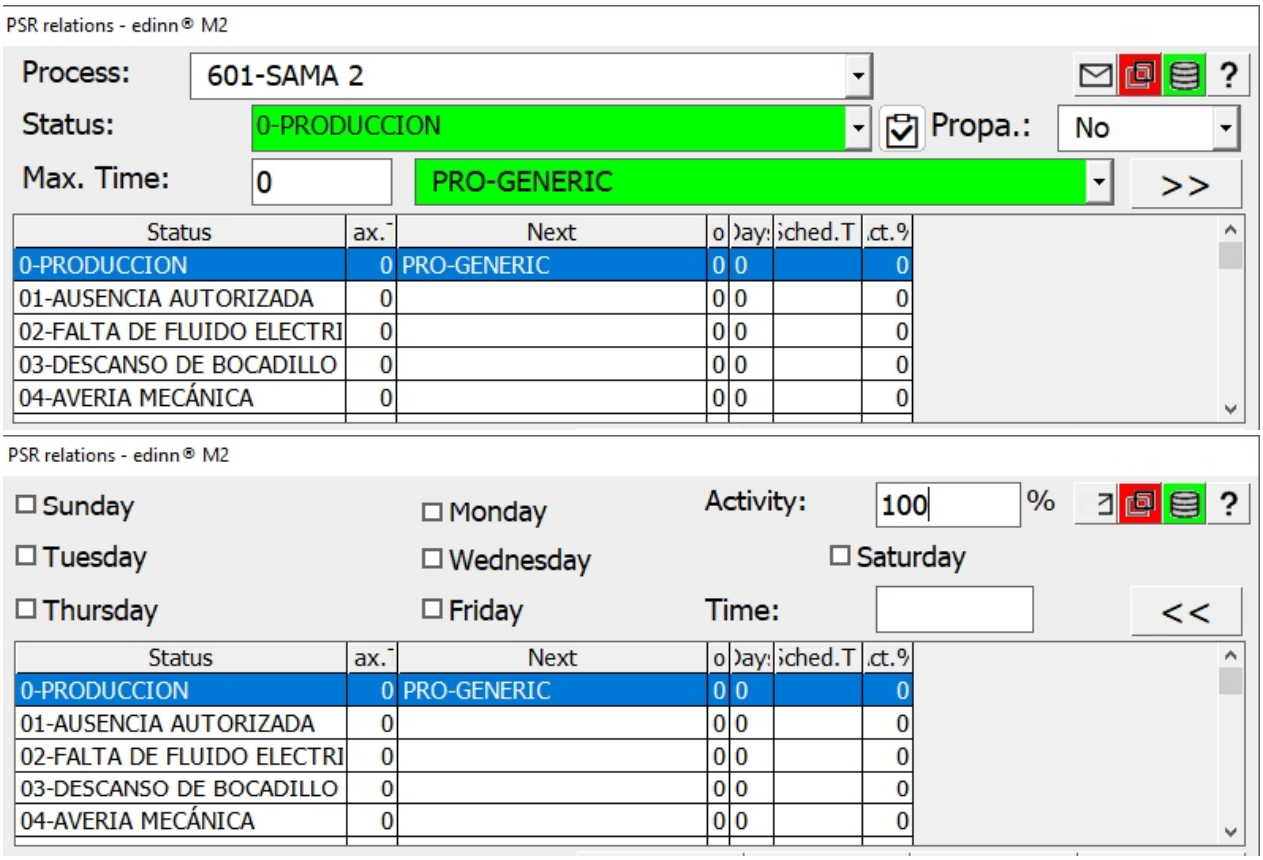
In the upper part: you will see the configuration for the **statuses**, with a grid with the relations between Processes and Statuses already configured

In the lower part: you will see the configuration for the **results**, with a grid with the relations between Processes, Statuses and Results already configured

Above each grid you will see the configuration parameters, with which you can perform operations on the database such as **Insert**, **Save** or **Delete**.

Cons. button: allows configuration of specific **consumptions** parameters for this process, status and results.

CONFIGURATION PARAMETERS FOR STATUS (and a specific process):



- **Process:** for which the relation is established.
- **Status:** for which the relation is established.
- **Check button** : to add/remove relations between one status and multiple processes.
- **Propagate**: if the status will be automatically propagated to processes that are later in the area. Please see difference between production lines and areas.
- **Max. Time**: maximum time in minutes that the selected process can be in this status. In the field on the right you can specify the next status that will begin after the closing of the initial status. Additionally, if you try to justify an

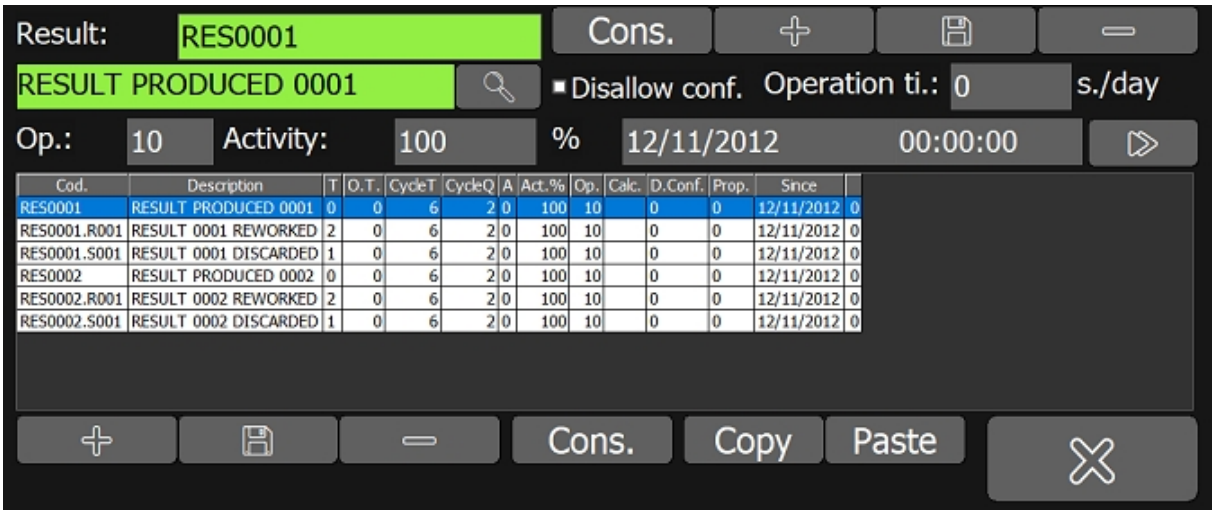
undefined status (FAI) to this status, the system will only accept the maximum justified minutes, the rest will be also required to be justified. Please see configuration of Statuses for more information about this field.

For example: *you can set the maximum time of a change of a produt to be 60 minutes, once surpassed, the system will automatically close this status.*

-  shows more options:
  - **Schedule**: status can be automatically scheduled by days of the week and time. This allows, for example, to close days or weeks with END OF PRODUCTION and start them up when needed.
  - **Time**: time at which the status will automatically be inserted.

CONFIGURATION PARAMETERS FOR RESULTS (and a specific process):

> **NOTE:** you can only view or manage results having previously **selected (in the table above) a production status** and **clicked the button with the magnifying glass icon**.



- **Result**: code and description of the result related to process and status.
- **Disallow conf.:** does not allows the user to configure the process and to add more statuses and more results.
- **Operation time**: type the seconds per day that the process should work in labour days. For more information please view the help of the processes configuration.
- **Op.**: maximum number of working users that can be working on this Process, Status and Result.
- **Activity**: percentage of activity that is required for the working user. This is a requirement configuration for the activity report. For example: if we introduce a value of 100%, will mean that the resource (person) must achieve 100% of OEE in this process, when in this status and producing this result.

- **Date and time**: date and time when this PSR relation starts applying. This is useful to change important parameters, like **Cycle Time** and **Cycle Unit**, without affecting historical data.

-  shows more options:

  - **Propagate**: if marked, the result change in this process and status will be cascade propagated to the related processes.

  - **Fields:** to configure new and customized fields for the PSR relations, This allows the system to adjust to any sector, as the organization can create its own information fields.. For more information, please see: Fields Configuration.

  - **Drop down combo of calculation types:** here you can indicate the Calculation type which will be used to calculate the nominal speed (Cycle Units / Cycle Time). This is useful for the system to automatically supervise the results and when they are generated, since the working user can be required to indicate the estimated time and quantity of results.

WARNING: Please note that this field **is not the same as the Calculation of the** Process configuration. The one in the process configuration is to calculate production, and this is to calculate nominal speed. Leave this field blank if you do not need to automatically calculate the nominal speed (Cycle Units / Cycle Time).

  - **O.T.** (*Operator Time*): time in seconds of work of the resource (person) per cycle time.

  - **C.T.** (*Cycle Time*): seconds for each cycle time. Please see Maximum speed.

WARNING: Please verify that the Cycle Time is equal or greater than the "Check counters every" of the edinn Server Console, Monitor, Common Tab.

  - **C.U.** (*Cycle Units*): number of units of results expected per cycle of production. Please see Maximum speed.

  - **Auto**: indicates the number of units of results with a maintained speed for the system to automatically update the maximum speed of the process. Automatic update of maximum speed is disconnected if this value is left to 0.

- **Operation time:** type the seconds per day that the process should work in labour days. It is useful to calculate availability as the percentage of time that the process was working comparing with this number, independently of the time schedule that the process followed. It also affects the speed ratio, which turns into the percentage of the time that the process met the due dates of the orders it performed. Therefore, if different than 0, it will affect how the availability

and speed ratios are calculated.

WARNING: This field will take by default the same value as the homonym field in process configuration and has the same implications, but it allows setting different values for different dates or results.

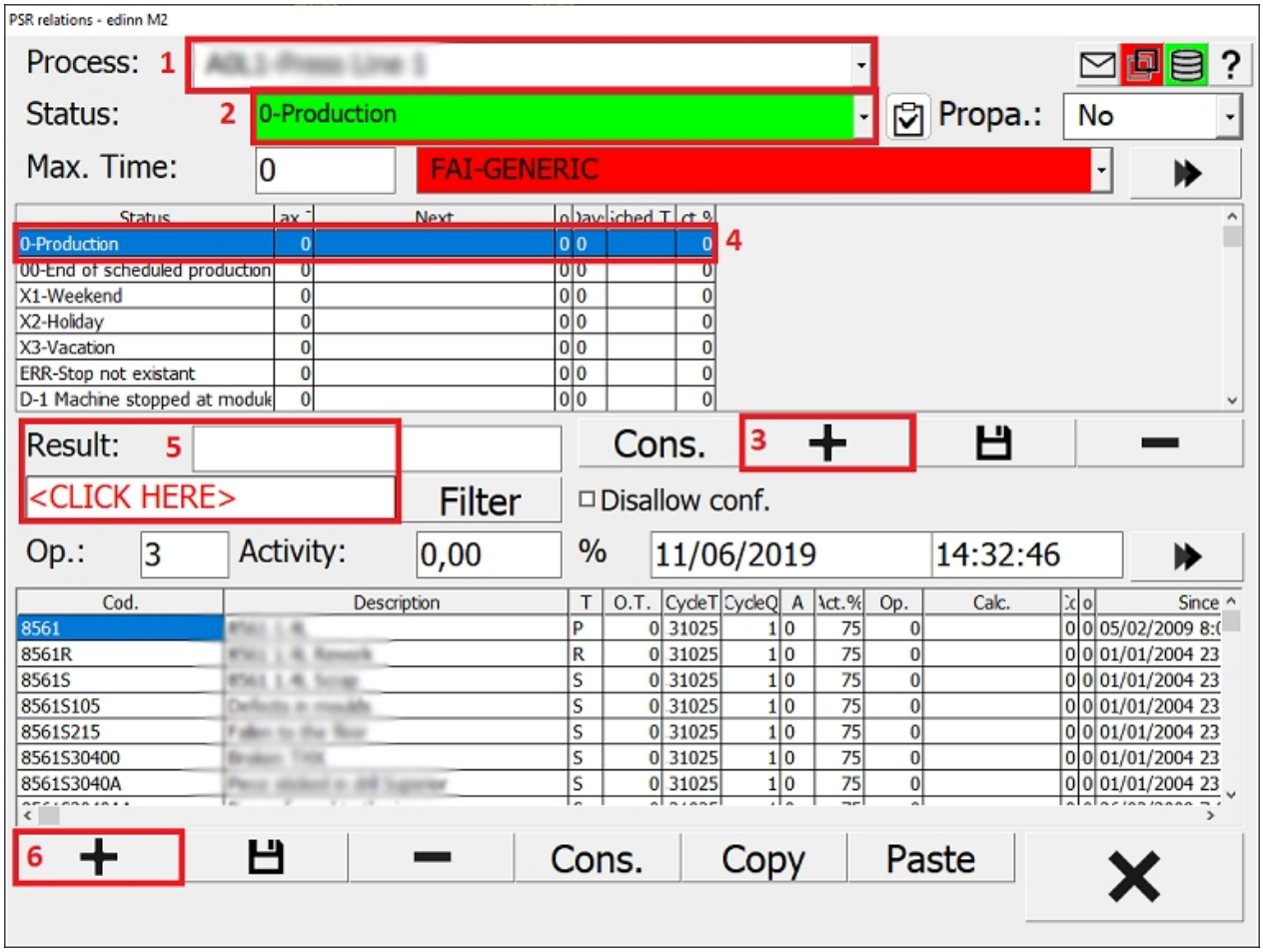
- **Copy**: there are three possibilities:
  - If olny the process is selected, it will copy state and results from the selected process.
  - If a status is selected, it will copy the related results of the selected satus.
  - If a result is selected, it will copy fields and consumptions of the selected result.
- **Paste**: paste the data copied with the previous button.

How to configure a Process-Status-Result (PSR) step by step

In [**Click to configure**] select [**PSR relations**]. To **relate** processes with specific statuses for specific results, complete the following configuration.

Relate the process with its statuses and with its results

1. **Select the process** for which the relation is established.
2. **Select the status** for which the relation is established. You can change the configuration associated parameters. These parameters, and not the general parameters for this status, will be applied when this process will be in this status.
   For example, by clicking on  , you can **schedule** the statuses for a certain processes, for more information, please see: How to schedule statuses.
3. **Insert, by pressing the 1st**  , the relationship between the selected process and status. Repeat for the statuses that you need.



4. **Select** those statuses which generate results. Usually the statuses of production type.

5. **Enter or select** (by clicking on <CLICK HERE>) the result that you need to relate with the production status and therefore with the selected process.

6. **Insert, by pressing the 2nd** , the relationship between the selected status and result. Repeat for the results that you need.

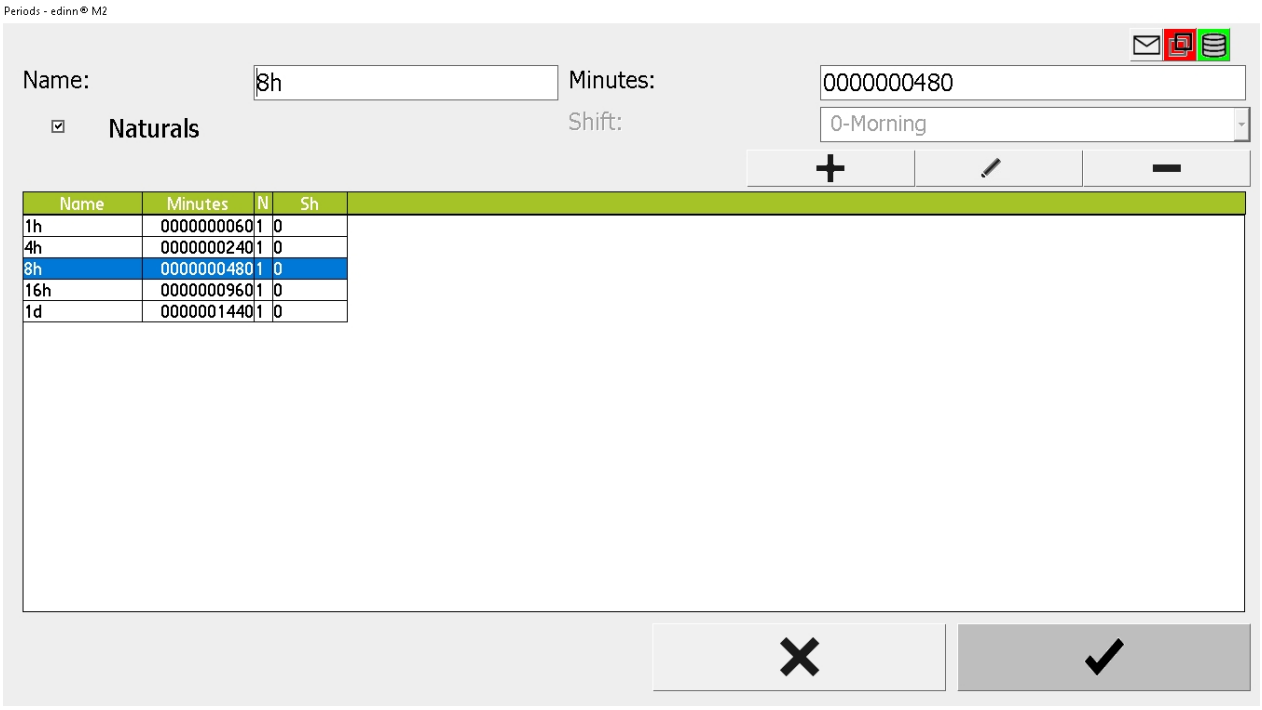
Periods

Click to configure > Periods

Periods mean the different periods that are considered by the system to calculate ratios and to show them on the terminal.

In the lower part: you will see a grid with the periods already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

In the upper part: you will see the configuration of each record, where you can configure the parameters of a new insertion or edition.



CONFIGURATION PARAMETERS:

- **Name** of the period.
- **Minutes** of duration of the period.
- **Naturals check:** indicates if the period is starting those minutes backwards from the moment that are selected on the Main Window of the terminal. If is not marked, will enable the Shift selector:
  - **Shift**: indicates when the counting of the indicated minutes start. the options available can be configured on the Shift configuration.

How to configure a period step by step

In [**Click to configure**] select [**Periods**]. To create and configure **periods**, complete the following configuration.

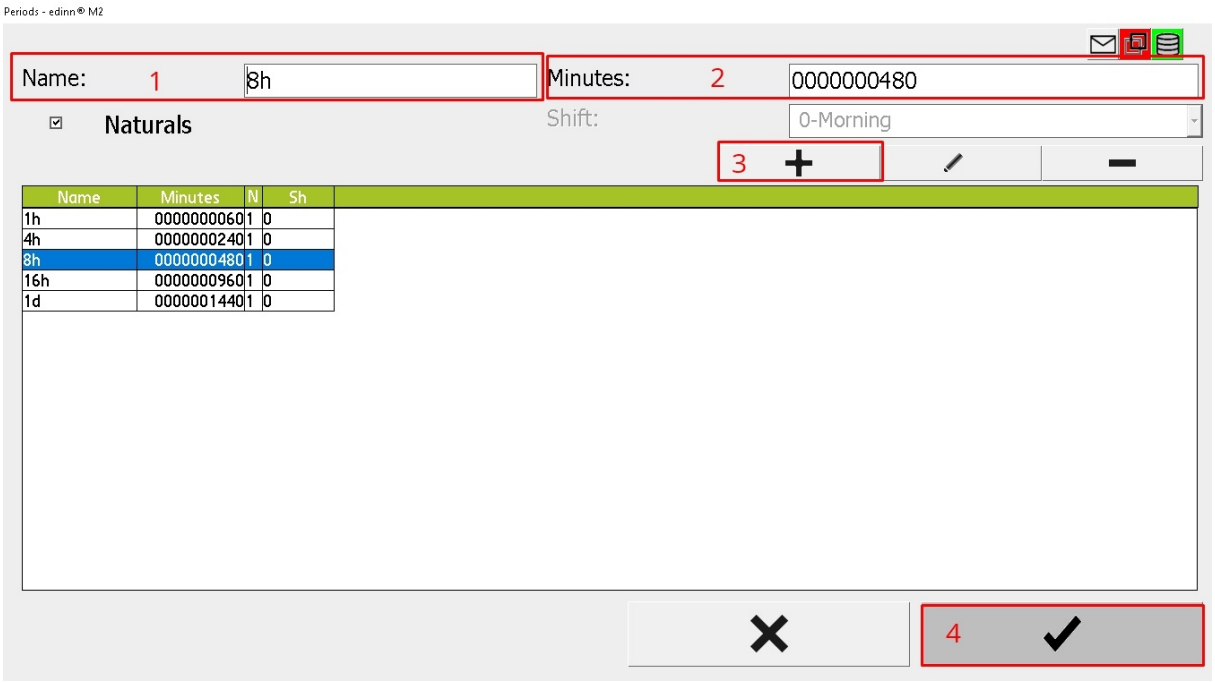
Step 1. List all the items

List all the **periods** that you want the system to consider to calculate ratios and to show them on the terminal, e.g: 1h. 4h, 8h, 1d...

Step 2. Enter the list in edinn

1. **Name:** of the period.

2. **Minutes:** of duration of the period.



Step 3. Save changes

3. Press ✚ to insert in the table the configuration of the period. Repeat this action for **each** period.

4. When all the changes have been done, press the ✓ button.

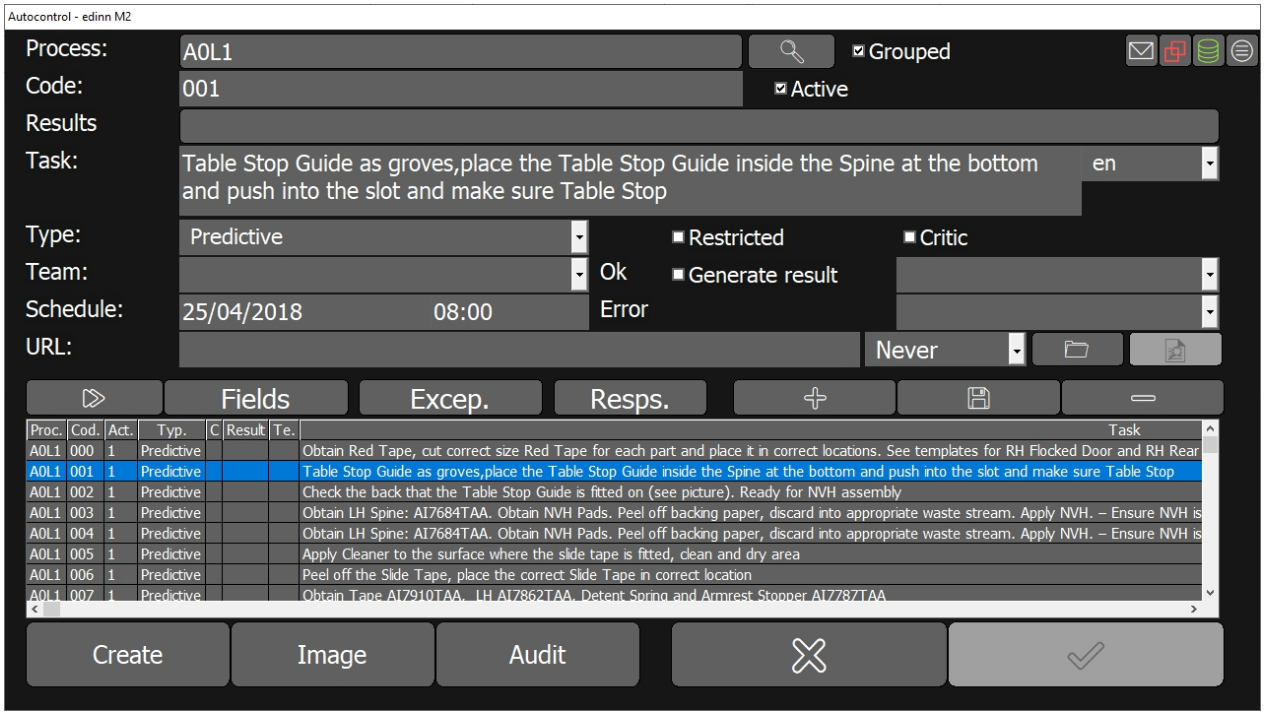
NOTE: Please see Period configuration for deeper configuration.

Autocontrol

Click to configure > Autocontrol

In the lower part: you will see a grid with the autocontrol tasks already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

In the upper part: you will see the configuration of each record, where you can configure the parameters of a new insertion or edition.



BUTTONS AND FUNCTIONALITIES:

- **Fields button:** configures new and customized fields, that allow the user to adapt the platform to a specific industry. For more information, please see: Fields Configuration.
- **Exceptions button**: configures of exceptions for this task. Please see the Exceptions Configuration.
- **Resps. button**: assigns persons to the autocontrol task who will be advised by email when the autocontrol task:
  - Was not done and was blocked by the system.
  - Should be done because is about to be blocked, this is, has entered in its tolerance +.
  - Should be done in less than 8 hours.
- **Create button:** creates now the autocontrol task.
- **Image button:** associates a graphic to the autocontrol task.
- **Audit button**: opens a window which allows to know all changes performed to this autocontrol task.

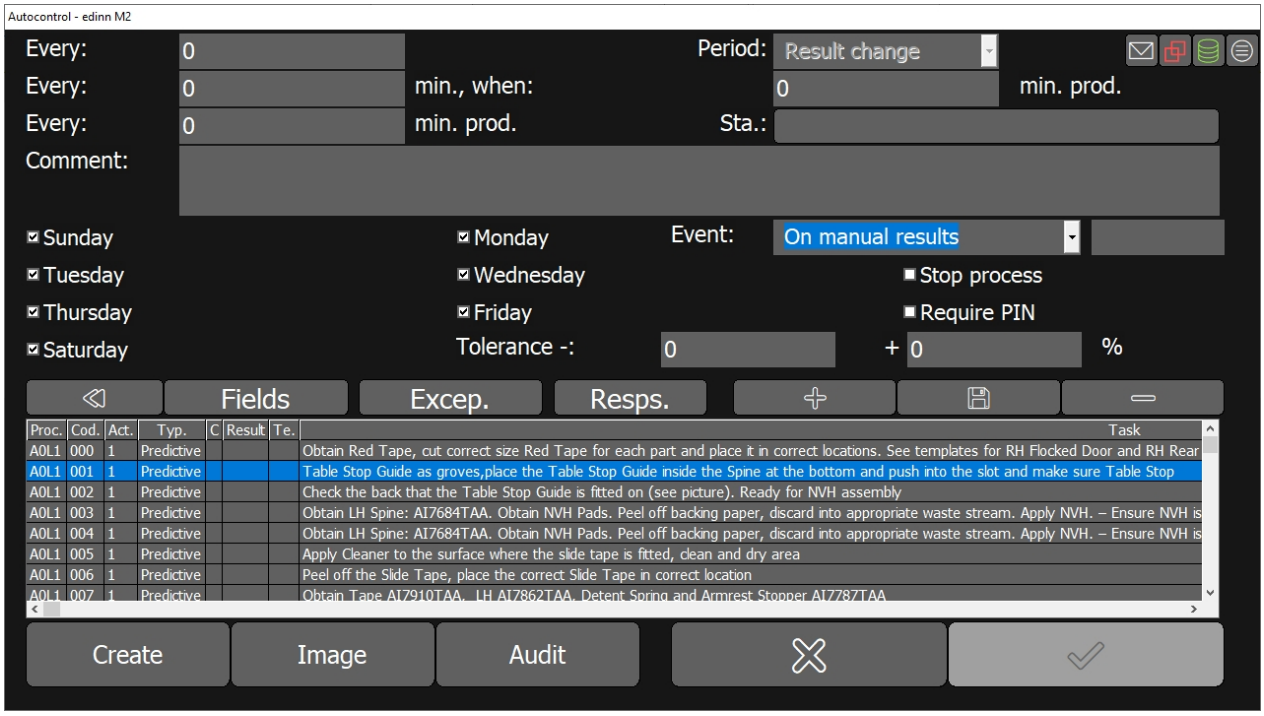
NOTES: 'Fields', 'Excep.', 'Resp.', 'Create' and 'Image' buttons will apply changes to all 'Proc.' column processes.

CONFIGURATION PARAMETERS:

- **Process**: process for which the autocontrol task is launched.

- **Grouped:** Allows to visualize processes that share autocontrol tasks (equal data) with the processes selected in the previus field ('Process:').
- **Code**: code of the autocontrol task.
- **Active**: allows activating or disactivating the automatic launch of this task.
- **Results**: results codes, if the autocontrol task is specific to certain results. It will only be launched when any of the indicated results are active.
- **Task**: description of the task, you can choose as well the default language.
- **Type**: the type of the autocontrol task, choose between the following options: *Predictive, Preventive, Corrective, Procedure or Quality.*
- **Restricted**: if the task can only be performed by members of the assigned team.
- **Critic check**: if the task is critical or not. The user will see on his screen if the task is marked as critical. Additionally, the critical tasks will appear before the rest.
- **Team**: if the task is assigned to a certain team.
- **Schedule**: when this task (date and time) will be launched for the first time or by the button **Launch**.
- **Ok**
  - **Generate res.**: indicates that when this task is carried out, this is when passes to the ok state, a result will be automatically produced. Therefore, a production counter with the Manual Driver should be configured in the Server Monitor.
  - **(Generate status)**: select the status that will be automatically generated when this task will be performed, this is, when it will pass to the **ok** state.
- **Error**
  - **(Generate status)**: select the status that will be automatically generated when this task will be marked with the **error** state.
- **URL**: document or application that will be opened when the user clicks on the select that appears after having clicked on the task, on the autocontrol window. Please see how to configure URLs.URLs can be selected from  and its syntax can be cheched from the button  . Configured documents can be opened automatically when changing the status task with the settings in the list on its right.
- (open document): Sets when documents are automatically opened:
  - Never (default value)

- Error (When a task is marked as error)
- Warning (when a task is marked as warning)
- Warning or error (when a task is marked either as warning or error)



 shows more options:

- **Every**: every x produced pieces the task will be launched.
- **Every… min.**: every x natural calendar minutes the task will be launched.
- **When… min.prod.**: Subordinates the launching of the previous field criteria, to the production time minutes indicated in this field.
- **Every… min.prod.**: every x minutes of production the task will be launched.
- **Sta.:** The task will start when the status has been activated in the process.
- **Comment**: comments related to the task.
- **Days of the week**: once the task is ready to be launched, because one of the launching criteria has been reached, the task will then have a “Due Date”, or suitable date in which the task should be carried out. When we mark the days of the week in which we want the task to be carried out, the Due Date will be modified in order to "adjust" to the marked days. In other words, the task will be carried out on the marked days if there is tolerance (next point).
- **Event:** in this field you indicate with which event you want the autocontrol task to be launched, it can be when:

- The result changes.
- The work order changes.
- The result or the work order change.
- A new result is manually inserted.
- A work order stops.
- A field changes in results. It only applies when starting a work order.

NOTES: The space located on the right side of "Event" is enabled only for the option "A field changes in results". In this field, the user can indicate the variable id of a field (user can add several Ids separated by ";") or it can be left empty to launch the task when there is a change in the value of any field.

NOTES: The launch of autocontrol tasks due to order change will not be carried out if the order was paused or pre-finished.

NOTES: autocontrol tasks launched by event remain pending unless the same tasks are triggered later, at which time they will go to "expired" status.

- **Stop process check**: if it is necessary to stop the process before realizing the task. The user will see it marked on his screen.
- **Req. PIN**: indicates if the user is obliged to introduce his/her PIN code before approving the task.
- **Tolerance -**: minor tolerance means that the task will be shown to the working user before the Due Date. For example, if the task is launched every 1440 natural minutes, the task will be launched every 1440 minutes and will have a Due Date every 1440 minutes. If there is a minor tolerance of, for example, 10%, the task will be launched every 1440 minutes and will have a Due Date of 1440 minutes after the previous task, but because of the minor tolerance, it will appear 144 minutes before the Due Date. In other words, the working user will be able to see the task and to anticipate to its realization.
- **Tolerance +**: the launching criteria is extended in % after the Due Date, before the task gets blocked and cannot be carried out anymore. For example, a task that is carried out every 1440 natural minutes, with a tolerance + of 10%, will remain on the working user's screen, if he did not carry out the task, 144 minutes after the Due Date.

NOTES: If all the fields 'Every' (first 4 fields of this form), are left zero, then the status of the task will never be converted automatically into 'Not done', which could cause the accumulation of old tasks. For more information, please see Autocontrol.

NOTES: Tolerances only affect autocontrol tasks scheduled to be launched by time or number of pieces.

- Use  to add new autocontrol tasks for all the selected processes from the 'Process:' dropdown.
- Use  to modify the selected autocontrol task. It will modify the task for all 'Proc.' column processes.
- Use  to delete the selected autocontrol task. It will delete the task for all the selected processes from the 'Process:' dropdown.

How to configure an autocontrol step by step

In [**Click to configure**] select [**Autocontrol**]. To create and configure autocontrol tasks, complete the following configuration.

Step 1. List all the items

List all the **autocontrol tasks** that are performed and its characteristics that will be later required.

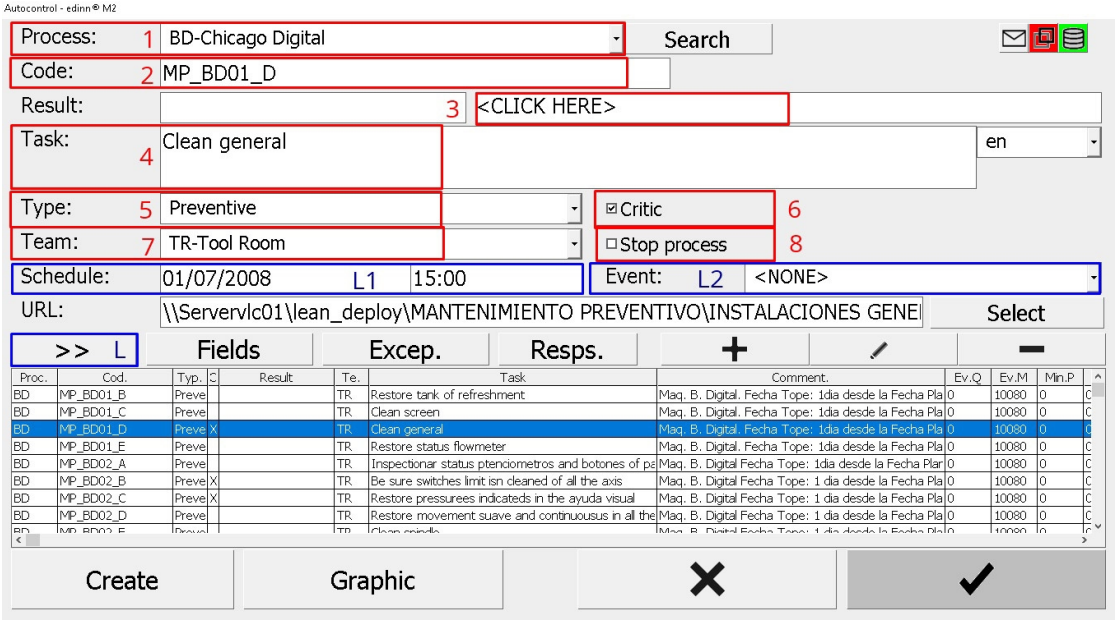
Step 2. Enter the list in edinn

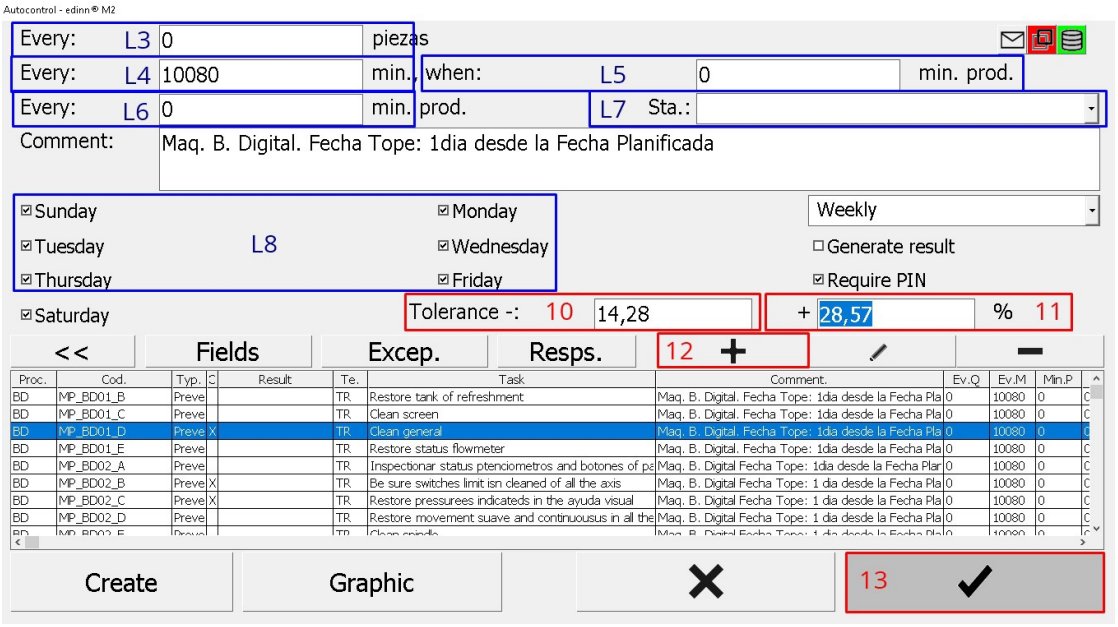
1. **Process**: select in the dropdown, the process for which the autocontrol task will be configured.
2. **Code**: define a code for the autocontrol task.
3. **Results**: select results if you want to restrict an autocontrol task to be launched only when any of those results is active.
4. **Task**: describe the task to be performed.
5. **Type**: choose between the following types of tasks: *Predictive, Preventive, Corrective, Procedure or TPM*.
6. **Critic check**: select it if the task is considered critical.
7. **Team**: choose a team (among those configured) if you want to asign the task to a certain team.
8. **Stop process check**: select it if it is necessary to stop the process before realizing the task.
9. **Launch:** you can define different reasons to launch an autocontrol tasks:
    - **L1: Schedule**: define when this task (date and time) will be launched for the first time.

- **L2: Event:** indicate with which event you want the autocontrol task to be launched: every time a result is changed, an order is changed or when a new result is manually inserted.
- **L3: Every**: indicate every how many produced pieces the task will be launched.
- **L4: Every… min.**: indicate every how many natural calendar minutes the task will be launched.
- **L5: When… min.prod.**: subordinates the launching of the previous field criteria, to the production time minutes indicated in this field.
- **L6: Every… min.prod.**: indicate every how many minutes of production the task will be launched.
- **L7: Sta.:** indicate the status actived with wich the task will be launched.
- **L8: Days of the week**: once the task is ready to be launched, because one of the launching criteria has been reached, the task will then have a "Due Date", or suitable date in which the task should be carried out. When we mark the days of the week in which we want the task to be carried out, the Due Date will be modified in order to "adjust" to the marked days. In other words, the task will be carried out on the marked days if there is tolerance (next point).

10. **Tolerance -**: set the period in % **before** the Due Date of an autocontrol tasks that the autocontrol task will be shown, so the working user can anticipate its realization.

11. **Tolerance +**: set the period in % **after** the Due Date of an autocontrol tasks that the autocontrol task will be shown, before the task gets blocked and cannot be carried out anymore.





Step 3. Save changes

12. Press ✚ to insert in the table the configuration of the autocontrol task. Repeat this action for **each** task.

13. When all the changes have been done, press the ✓ button.

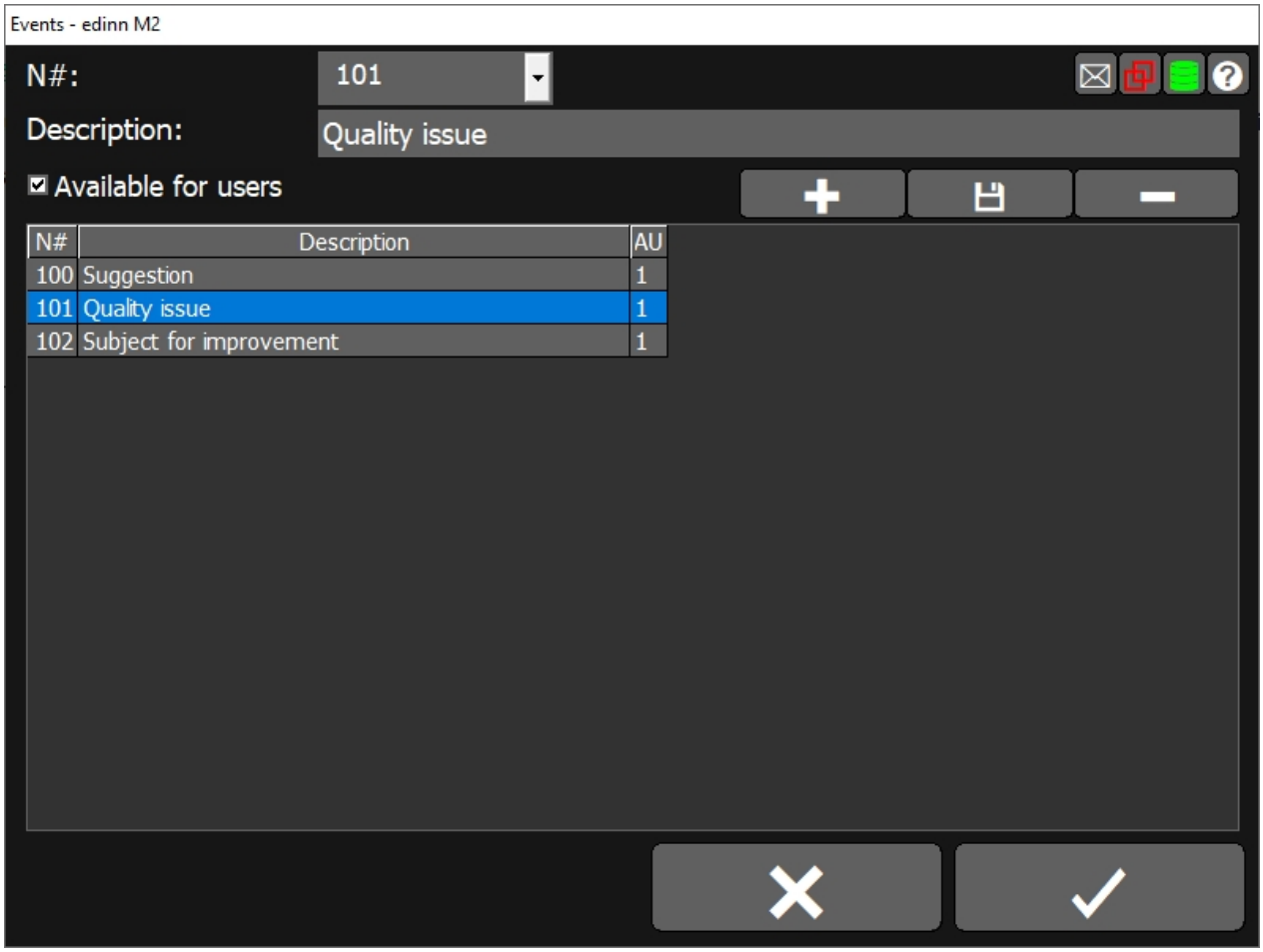
NOTE: Please see Autocontrol configuration for deeper configuration.

Events

Click to configure > Events

Events are the different events that occur in the system. Events from number 100 and higher can be configured within this window:

- **In the upper part:** you will see the configuration of each record, where you can configure the parameters of a new insertion or edition.
- **In the lower part:** you will see a grid with the events already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.



CONFIGURATION PARAMETERS:

- **N#**: number of the event.
- **Description**: description or name of the event.
- **Available for users**: if it will be available for users.

Routes

Click to configure > Routes

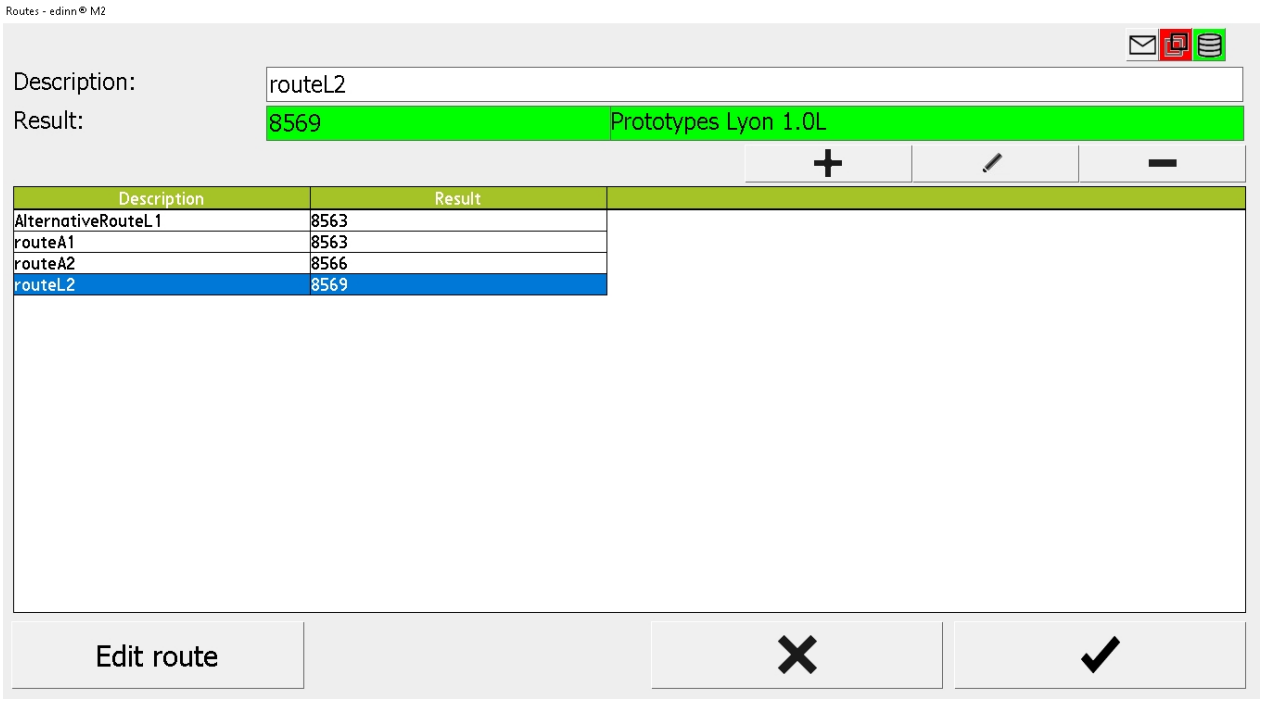
Routes are the different paths, upon the different processes, that results can follow to be generated.

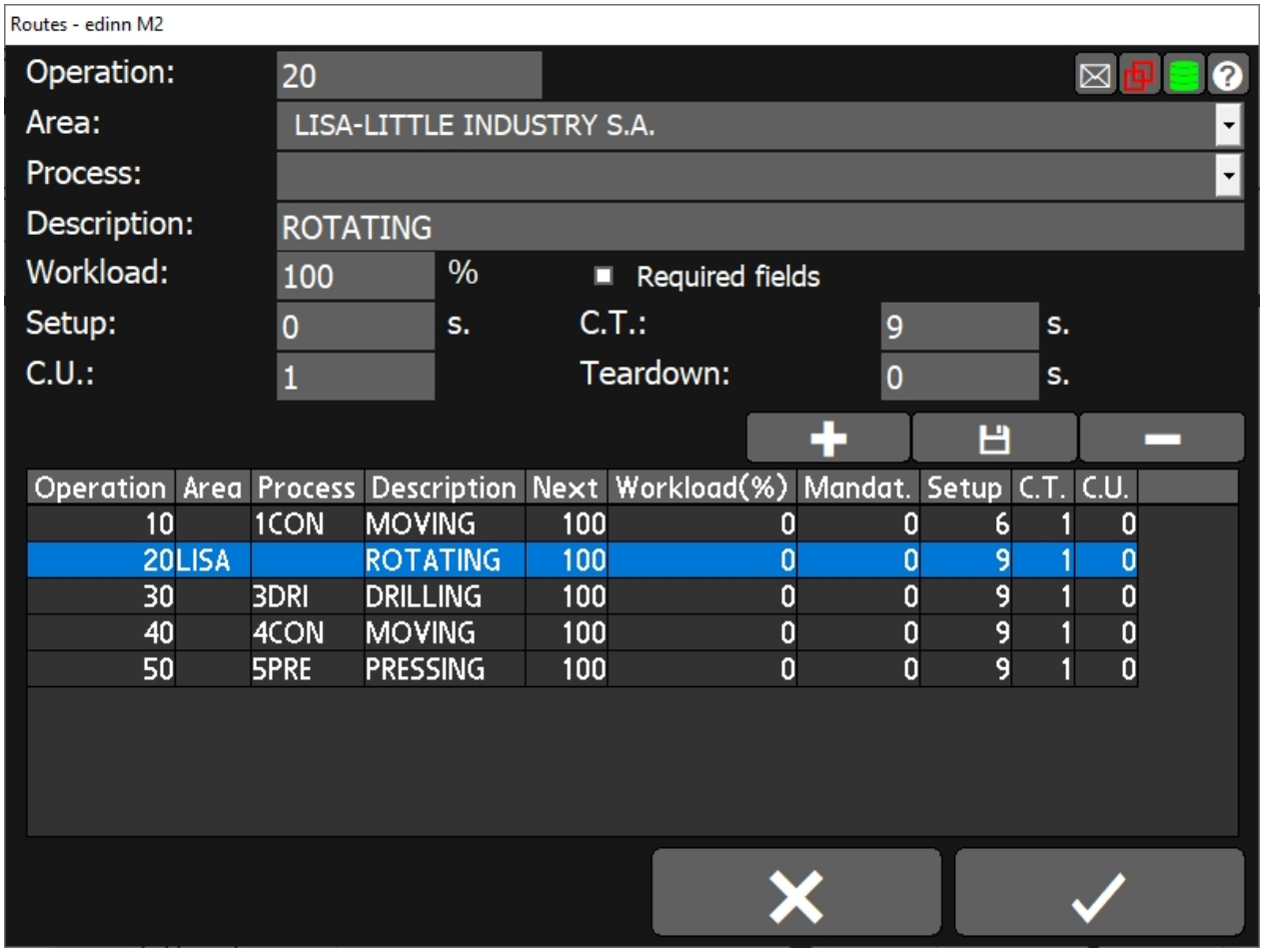
For example: *to generate a result, the #ProductA has to pass through 3 different processes, so the route will order the 3 processes in the order they follow: 10. #ProcessA 20. #ProcessB 30. #ProcessC*

In the lower part: you will see a grid with the events already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

In the upper part: you will see the configuration of each record, where you can configure the parameters of a new insertion or edition.

There are 2 windows for this configuration. The first defines the name of the routes and the result that they produce at the end. The second define the path for the route.





CONFIGURATION PARAMETERS:

- **Operation**: code to identify the step of the route.

> **NOTE:** it is advisable to go from ten to ten (10, 20, 30) to be able to add intermediate steps in case of forgetting them in the initial configuration

- **Area**: destination area. The system will choose the first process found, inside this area, not in an unscheduled status.
- **Process**: destination process, if an area was not selected in the previous field.
- **Description**: desciption of the step.
- **Workload**: if all the workload is passing from previous process to destination or for example 50% is passing to one destination process and the other 50% is passing to another destination process, indicated by another step.
- **Required fields check**: if the working user is forced to fill-in all fields before passing to the next step.
- **C.T.** (*Cycle Time*): time in seconds per cycle of production. Please see Maximum speed.
- **C.U.** (*Cycle Units*): number of units expected per cycle of production. Please see

Maximum speed.

- **Teardown:** dismantile the process in seconds.

Quality (SPC)

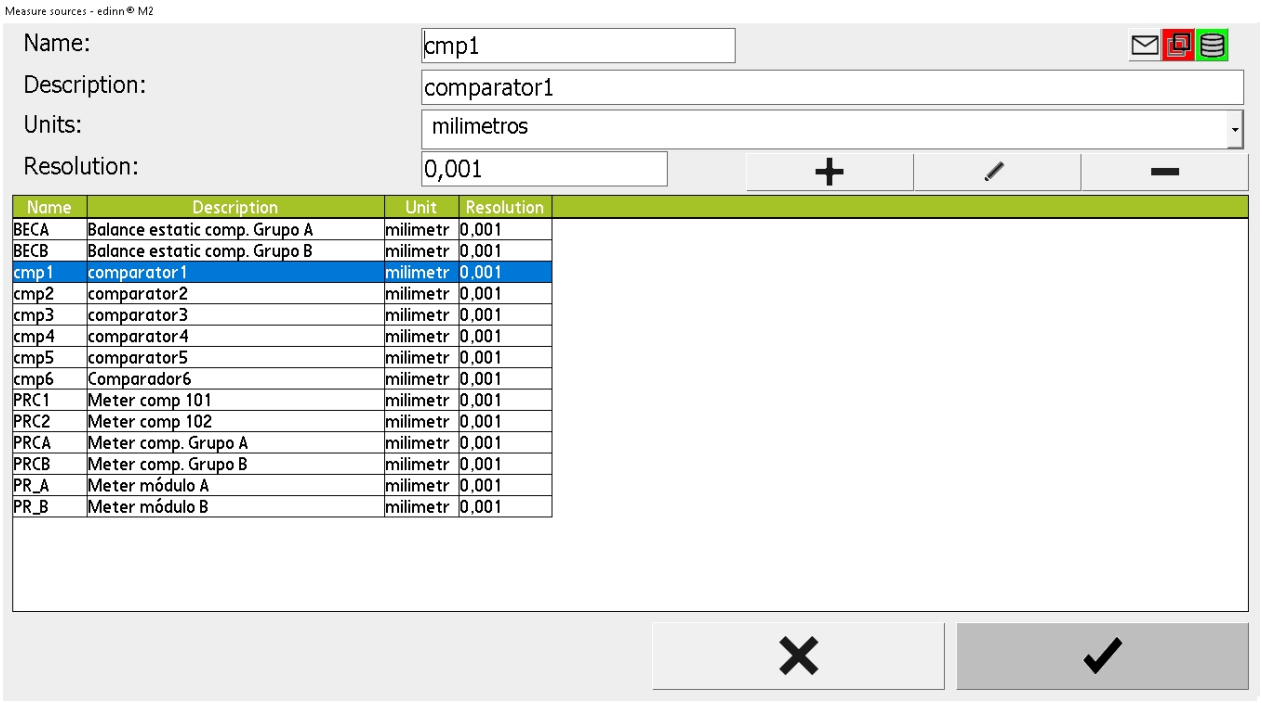
Measure sources

CLICK TO CONFIGURE > Measure sources

Measure sources are measurements devices that are used to take quality measurements for SPC (Statistical Process Control).

In the lower part: you will see a grid with the measurement devices already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

In the upper part: you will see the configuration of each record, where you can configure the parameters of a new insertion or edition.



CONFIGURATION PARAMETERS:

- **Name** and **description** of the measurement device.
- **Unit**: in what unit is measured.
- Maximum **resolution** of the measurement device. This will condition the

maximum resolution in which the working user can introduce measurements.

Result-Programme-Measure (RPM)

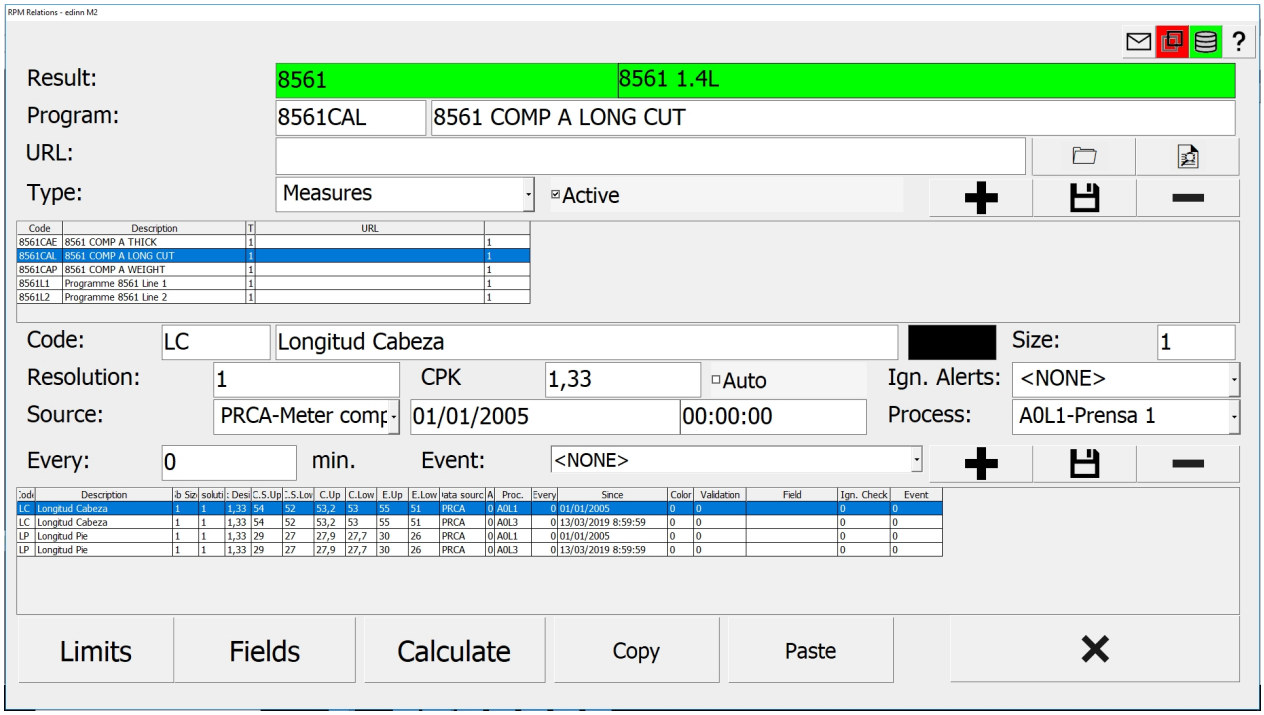
CLICK TO CONFIGURE > RPM relations

RPM relations relates Results, with Quality SPC programmes and Measures. This means that to each result, a different measurement programme can be associated, and to each programme, one or various measurements can be associated. Having various measurement programmes allows, for example, to apply a more strict programme in certain periods, without changing the measurements.

In the upper part: you will see the configuration for the **measurement programmes**, with a grid with the programs for each result already configured

In the lower part: you will see the configuration for the **measures** and a grid with the ones already configured

Above each grid you will see the configuration parameters, with which you can perform operations such as **Insert**, **Save to the grid** or **Delete**.



CONFIGURATION PARAMETERS:

- **Result**: result code and description.
- **Program**: code and description of the measurement programme.
- **URL**: document or application that will be opened when the user clicks on the

 that appears after having clicked on the measure, on the quality (spc) window. Please see how to configure URLs. URLs can be selected from  and its syntax can be cheched from the button  .

- **Type**:
  - **Results:** if measurements are taken measuring all variables of the same result.
  - **Measures:** if measurements are taken measuring the same variable of all results.
  - **All Measures:** if measurements are taken measuring all variables of all the results.
- **Active:** If not marked, program measurements will not be requested to the user.
- **Code**: code and description of the measurement.
- **Size**: size of the subgroup. Number of results to be measured at a time. In the program types "Pieces" and "All Measures", the size of the subgroup must be the same for all measures.
- **Resolution**: maximum allowed resolution.
- **CPK**: CPK target (see quality SPC for more information).
- **Auto (check)**: if the control limits should be recalculated automatically. It is necessary to configure the server to calculate every certain time and for specific data.
- **Ign. Alerts:** select in which case alerts are ignored:
  - **Trend:** the terminal will not warn nor request for justification when there will be 7 or more points ascending, descending, above or below the average.
  - **Limits and trend:** the terminal will not warn nor request for justification when there will be 7 or more points ascending, descending, above or below the average or when a point exceeds the control limits.
- **Source**: what measurement data source will be used.
- **Process**: if the measurement is associated and restricted to a specific process.
- **Every**: every how many productive minutes the user will be requested to introduce this SPC measurement. This will be indicated through the blinking of the SPC button on the main window. If a 20% of this time has been surpassed, a justification will be asked in the form of a text of why the working user is not introducing SPC data. The terminal will be blocked until this justification is made.

- **Event**: when any of these events happen, this measurement will be requested to the user. Events can be:
  - **Result change**: request the measure every time the result is changed.
  - **Order change**: request the measure every time the order is changed.
  - **Result or order change**: request the measure every time the result or the order change.
  - **Stop, 1 time**: request the measure when the process is stopped for more or equal than the time indicated in the previous field (Every). This measure will be requested only one time.

BUTTONS AND FUNCTIONALITIES:

- **Limits**: allows to configure the control, client and entry limits.
- **Fields:** allows to apply quality validations to fields. New validations can be added by using the development modules (API).
- **Calculate**: allows to calculate the control limits at this moment, according to the number of data indicated in the server. Only by clicking on the Modify button, the control limits that are introduced manually or automatically calculated, will be transferred to the data base.
- **Copy**: copies into memory what has been selected. That is, if only one program is selected, it will copy all the measures of the program. If a measure is selected, it will copy only that measure.
- **Paste**: pastes, from memory, what was previously copied with the previous copy button.

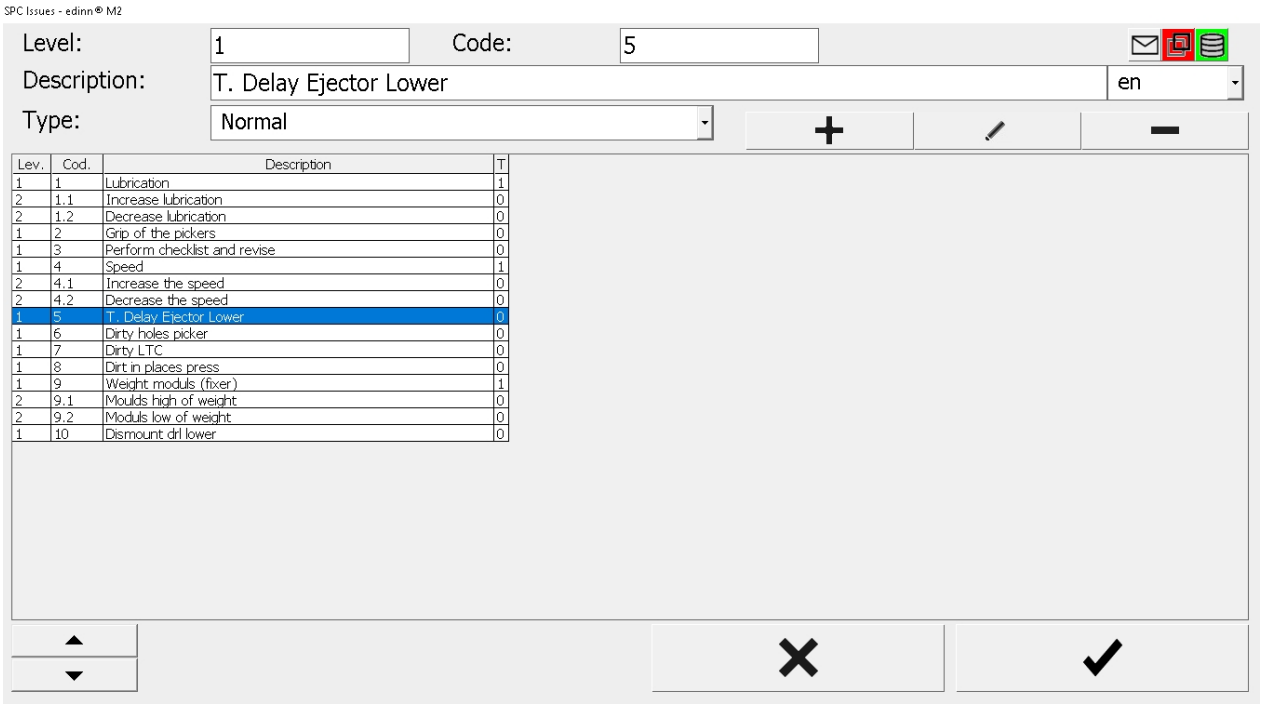
SPC Issues

CLICK TO CONFIGURE > SPC Issues

SPC issues are the reasons why a SPC point is outside the limits (control or client) and the actions taken to control it back.

In the lower part: you will see a grid with the SPC issues already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

In the upper part: you will see the configuration of each record, where you can configure the parameters of a new insertion or edition.



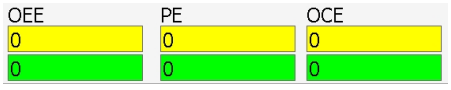
CONFIGURATION PARAMETERS:

- **Level**: issue level. This allows to structure the issues grouping them by levels, in a way that the user selects one and this opens new SPC issues in the next level. It is possible to indicate here 1, 2 or 3.
- **Code**: issue code or order.
- **Description**: issue description.
- **Type**: select between "Normal" or "Title" (to allow access to the next level).

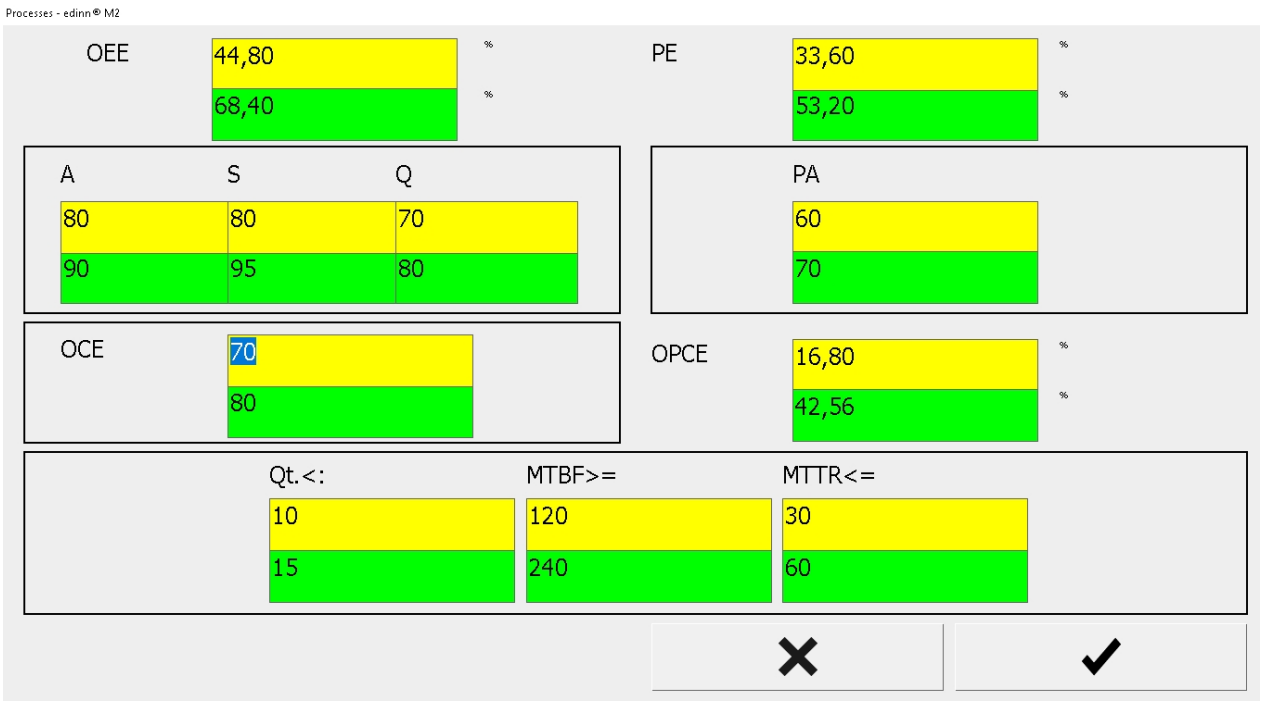
Other

Production targets

You can access to this window by clicking any of this records:



Only text fields that are inside boxes can be edited. Therefore OEE, PE and OPCE can not be edited and are automatically calculated upon the rest of the ratios.



CONFIGURATION PARAMETERS:

- **OEE and PE**: calculated upon the rest of the fields. Not editable.
- **Define OEE:**
  - **A**: availabilty.
  - **S**: speed.
  - **Q**: quality.
- Defines productive efficiency:
  - **PA**: Productive availability is the same as availability but without considering idle statuses.
- **OCE**: overall consumption effectiveness.
- **OPCE**: calculated upon the rest of the fields. Not editable. OPCE = OEE * OCE.
- MTBF Targets:
  - **Qt.**: target quantity of failures.
  - **MTBF**: number of minuts of duration of failures.
  - **MTTR**: number of minuts of duration of the failure until repaired.

Qt.<: yellow and green target for the maximum number of stops of the process, according the MTBF table.

- **MTBF>=**: yellow and green target for MTBF of the process, according the

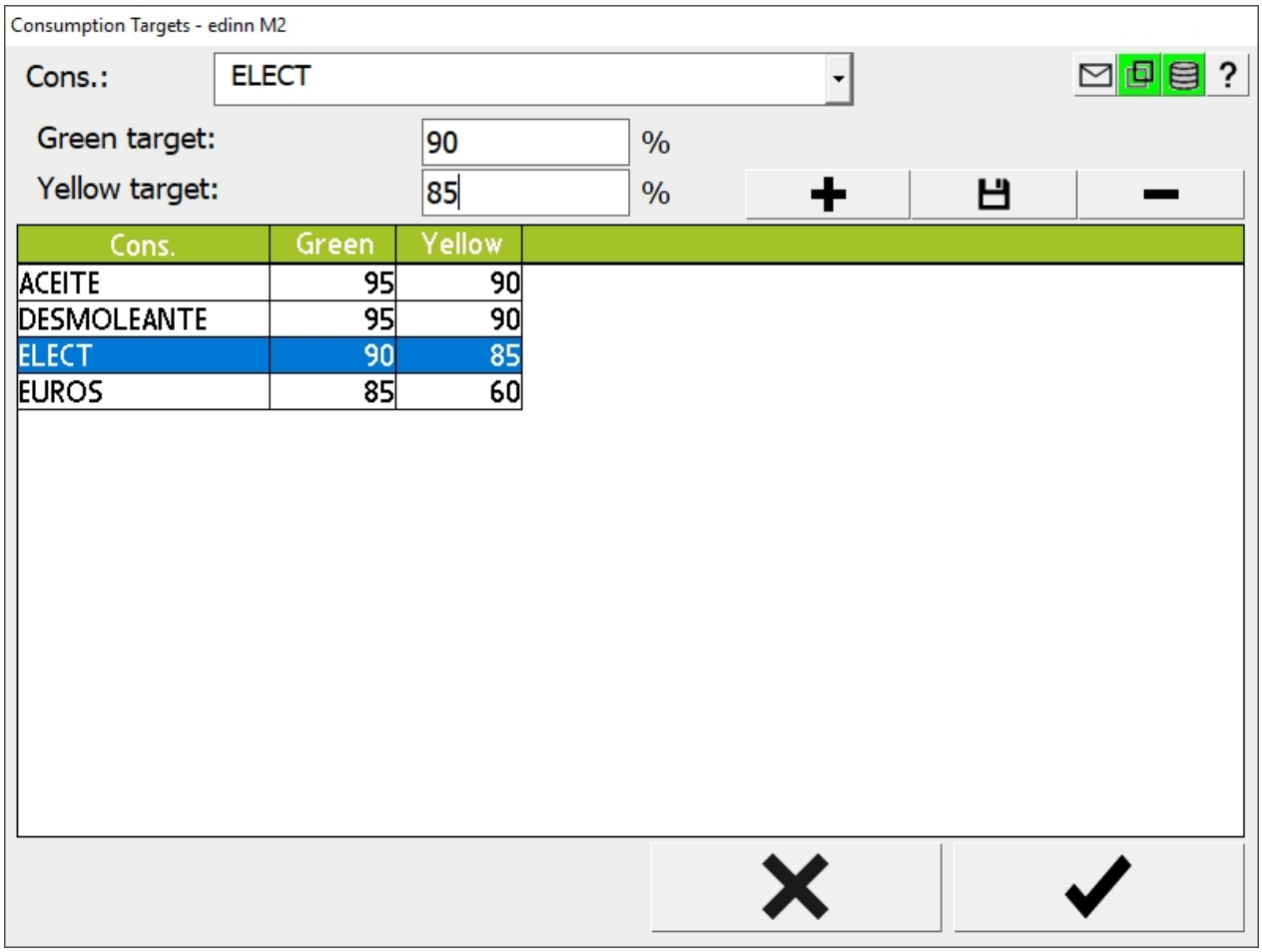
MTBF table.

- **MTTR<=**: yellow and green target for MTTR of the process, according the MTBF table.

Consumption targets

You can access to this window by clicking the Cons.Targ button.

Consumption targets configuration allows to define targets for consumption, which is useful to be able to see the consumptions in colors and therefore for the reduction of consumption losses.



In the lower part: you will see a grid with the consumption targets already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

In the upper part: you will see the configuration of each record, where you can configure the parameters of a new insertion or edition.

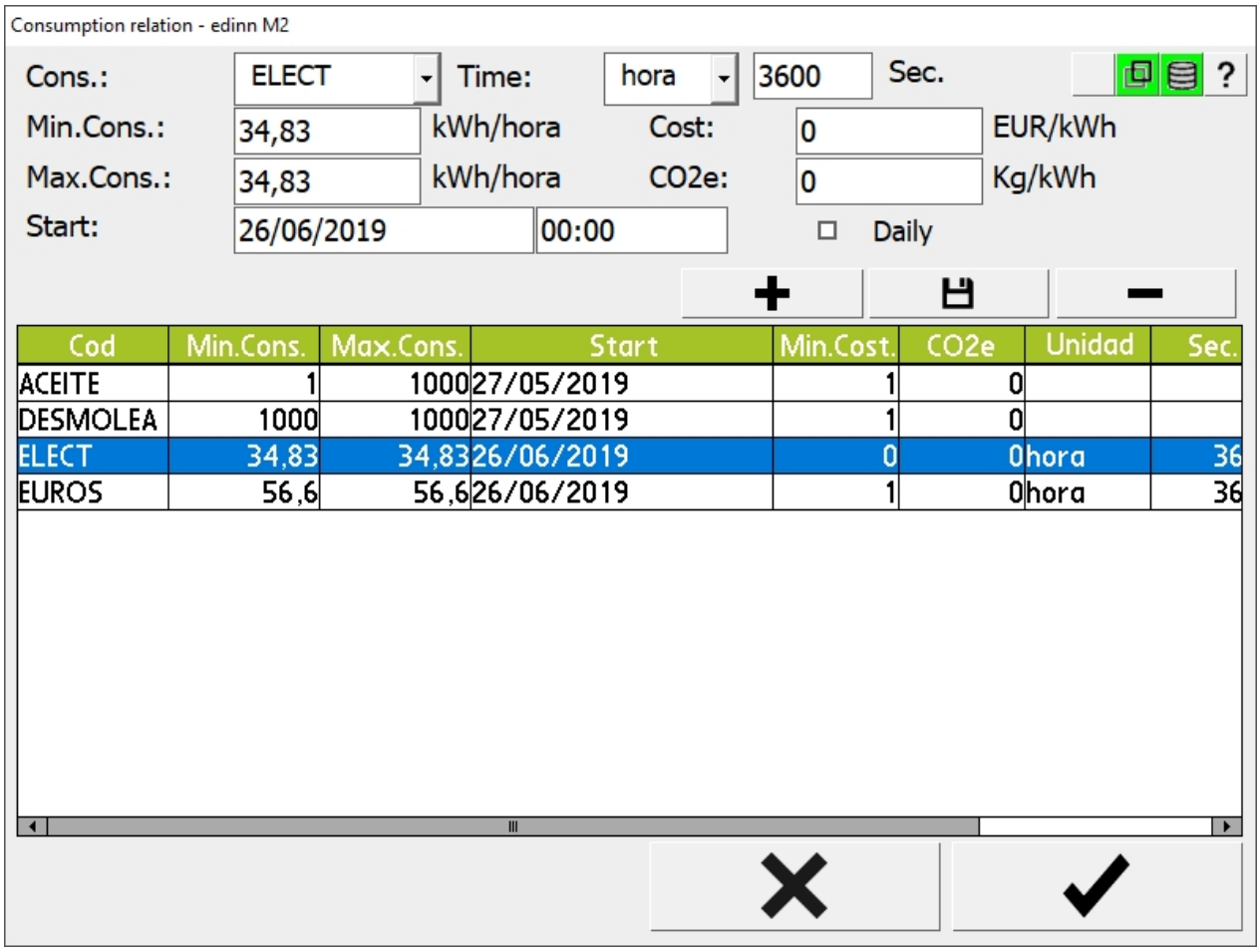
CONFIGURATION PARAMETERS:

- **Cons:** type of consumption.
- **Green target:** desired target for this consumption, above (+) or below (-) maximum, to be shown in **green**.
- **Yellow target:** desired target for this consumption, above (+) or below (-) maximum, to be shown in **yellow**.

Theoretical consumption

You can access to this window by clicking the Theor.Cons. button.

Theoretical consumption can be assigned to a number of elements, like processes, status, etc.



CONFIGURATION PARAMETERS:

- **Cons**: type of consumption.
- **Time**: select the unit (for example hour) and type the number of seconds that the unit of time implies. All consumption will be shown regarding this unit of

time.

- **Min. Cons. / Max. Cons.**: minimum and maximum consumption expected.
- **Cost.**: cost per unit of consumption.
- **CO2e**: CO2 gas equivalent per unit of consumption.
- **Start**: date and time on which this theoretical consumption starts applying.
- **Daily**: is this theoretical consumption applies every day at the same time. This is useful to define daily time frames. For example: to define 2 daily time frames, from 8 to 16 h. and from 16 h. to 8 h., you need to configure 2 theoretical consumptions and mark them "daily".

Exceptions

You can access to this window by clicking the 

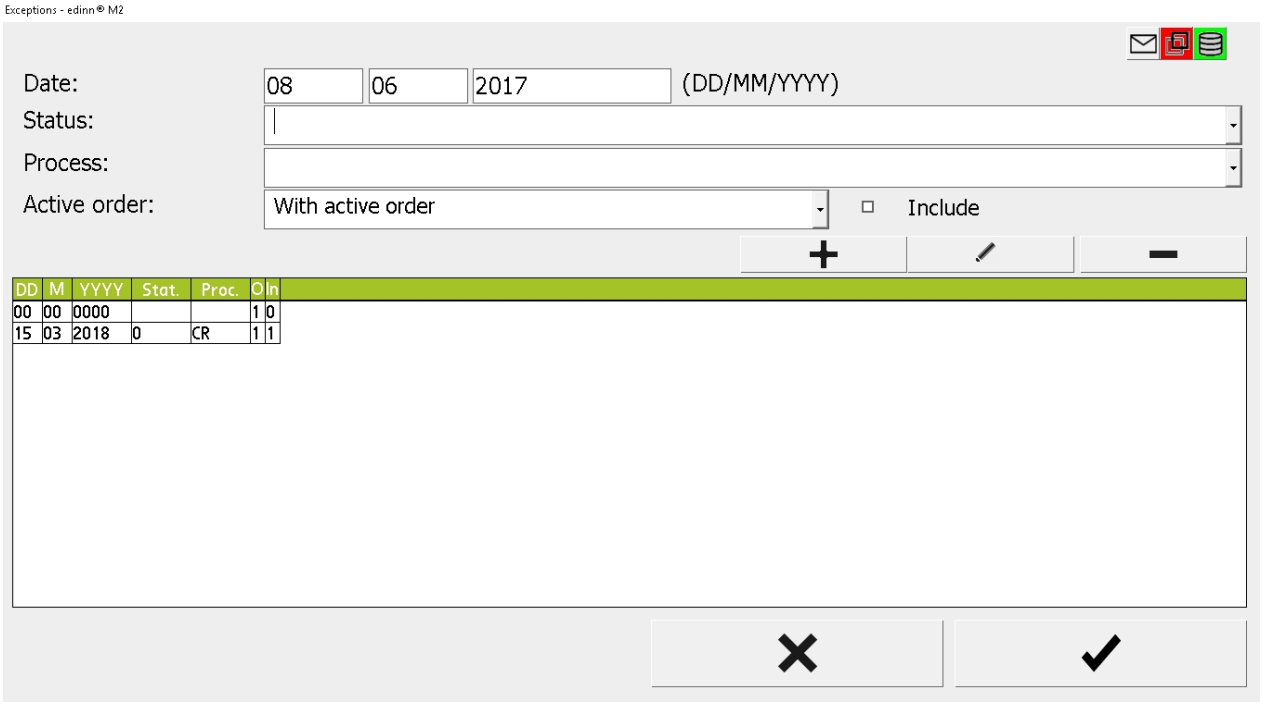
button.

Exceptions allow to skip the automatic launch of scheduled statuses or autocontrol tasks, or to force that they are launched at a specific moment. Exceptions can be created in various points of the system, mainly in statuses and autocontrol. Exceptions work this way:

- **Without exceptions:** it is launched when programmed.
- **With exception (exclusive):** it is launched when programmed, except if the exception is happening.
- **With inclusive exception:** it is launched when programmed, only if the exceptions is happening.
- **With exclusive and inclusive exception:** it is launched when programmed, only if the inclusive exception is happening and when the exclusive exception is not happening.

In the lower part: you will see a grid with the exceptions already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

In the upper part: you will see the configuration of each record, where you can configure the parameters of a new insertion or edition.



CONFIGURATION PARAMETERS:

- **Date Time**: Exception date. We can use zeros as a wildcard. Examples:
  - Indicating 00/08/0000 means every august of every year.
  - Indicating 01/05/0000 means every May 1st of every year.
- **Status**: useful to indicate an exception on certain statuses. That is useful, for example, to avoid the launch of an autocontrol task when the process is on a certain status.
- **Process** (when configuring exceptions for statuses): useful to indicate an exception on certain processes. That is useful to, for example, avoid the launch of an status for certain processes.
- **Active order**: useful to indicate exceptions when there is no active order. The system will consider that there is an active order when automatically calculating the future calendar.
- **Include**: Indicates if the exception is inclusive or exclusive. If not marked, it is an exception. If it is marked: it is an inclusion (do it only for this case).

Fields

You can access to this window by clicking the Fields button.

Defining **fields** allows to adjust the system to whatever the sector, no matter the particular

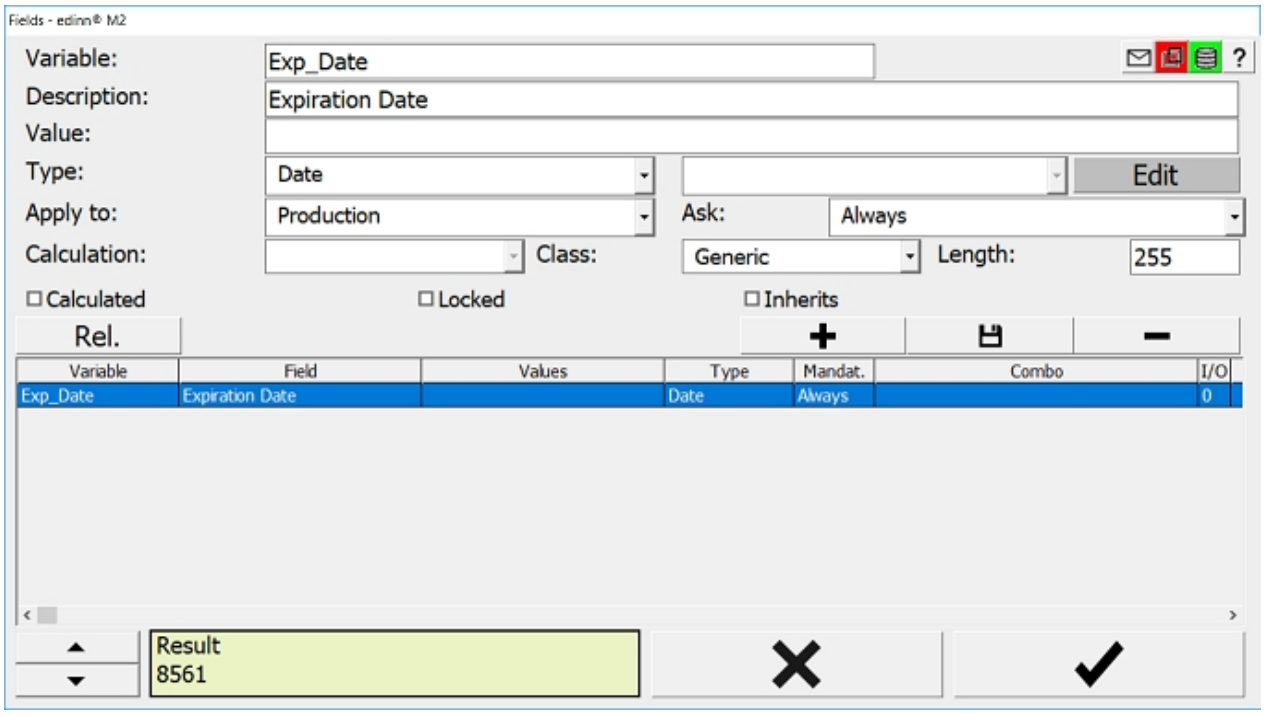
need of information that sector has.

For example, let us imagine that there is an industrial sector which typically needs to:

- Indicate color and quality finish for every result it generates: therefore these specific fields can be created for the results grid.
- Take certain context data like temperature, tool used, etc. whenever a new task is performed: therefore these specific fields can be created for the autocontrol grid.

In the lower part: you will see a grid with the fields already configured, on which you can perform operations on the grid such as **Insert**, **Save** or **Delete**.

In the upper part: you will see the configuration of each record, where you can configure the parameters of a new insertion or edition.



CONFIGURATION PARAMETERS:

- **Variable**: internal name of the field. Allows to carry out statistical studies, links with algorithms (types of calculation), etc. You can change the description (field) field, but not the variable name.
- **Description**: literally that appears to the working user.
- **Value**: default value or the possible answers that the user could give (separated by commas).
- **Type**:
  - **Real**: number with decimals.

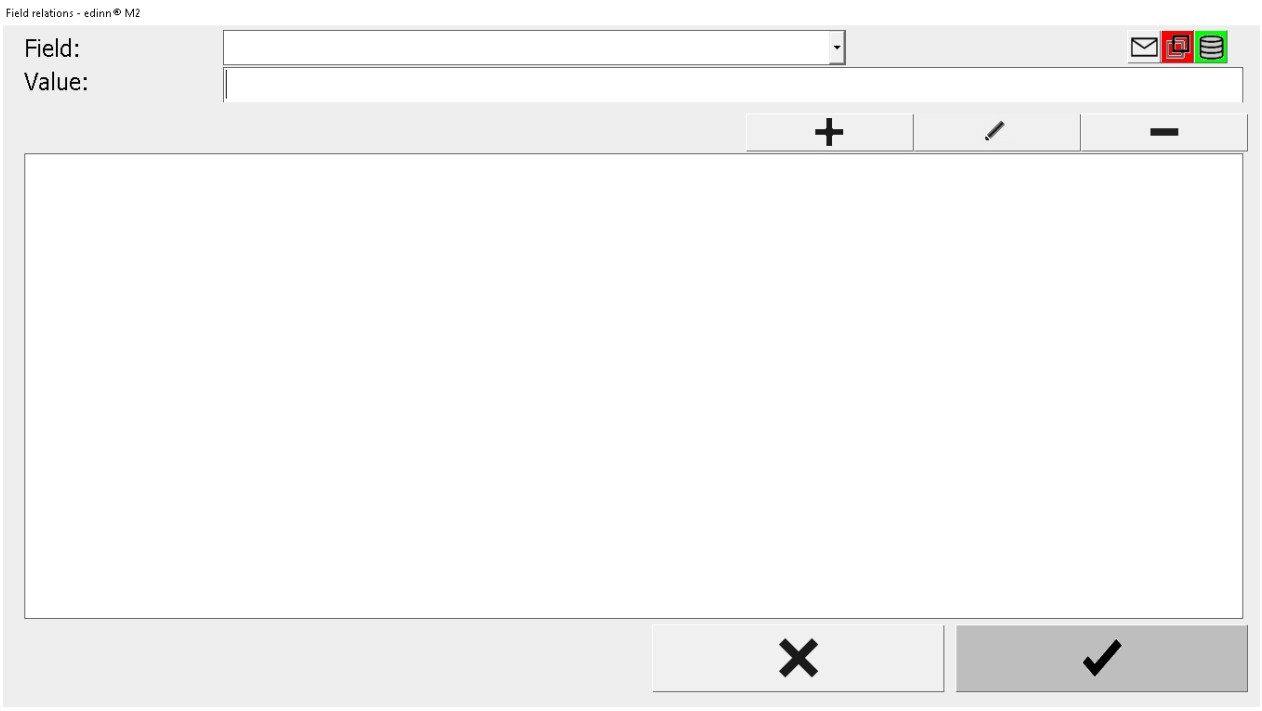
- **Integer**: number without decimals.
- **Date**: date in the dd/mm/yyyy format.
- **Time**: time in the hh:mm format.
- **Text**: free text.
- **IP Address**: IP network address.
- **Time Secs**: time in hh:mm:ss format
- **Selection**: allows to make a drop-down menu. Use the combos configuration window to configure options.
- **URL**: url upon the system syntax. Please see how to configure URLs.
- **Alphabetic**: only letters.
- **Date and time**: dd/mm/yyyy hh:mm format.
- **Date and time (secs.)**: dd/mm/yyyy hh:mm:ss format.
- **Mult. Selection**: multiple selection.
- **MAC**: MAC address.
- **Percentage**: number for percentages.
- **Autoincrement:** unique numeric value for the field independent of the result or task to which it is associated.
- **Button:** calls a function declared in a field management UDL.
- **TCP port:** TCP port between 1 and 65535
- **Now date:** current date in dd/mm/yyyy format.
- **Now time:** current time in hh:mm format.
- **Now time (secs.):** current time in hh:mm:ss format.
- **Now date and time:** current date and time in dd/mm/yyyy hh:mm format.
- **Now date and time (secs.):** current date and time in dd/mm/yyyy hh:mm:ss format.
- **E-mail:** Email address.

- For the cases of type Selection or Multiple selection: the next field is a drop down combo to indicate that the values of the field must be chosen from a predefined combo (list). The Edit button allows to create this combo lists.

- **Apply to**: indicates to what this field should be apply (Production, Inputs, Outputs, I&O [Inputs and Outputs], etc.).
- **Ask**: when the field is mandatory. The options are the following:
  - When configuring fields for **results**:
    - **If empty:** only if empty.
    - **Always:** the field is always mandatory.
    - **Never:** the field is never mandatory.
    - **On manual results**: when results are being inserted manually.
    - **On finish**: when finishing or pausing a order. This option will only apply for fields on inputs.
    - **On start**: when starting an order.
  - When configuring fields for **autocontrol**:
    - **Never:** never is mandatory.
    - **Status change:** when the status of the task changes.
    - **OK:** if the task status is changed to Ok.
    - **Warning:** if the task status is changed to warning.
    - **Error:** if the task status is changed to error.
    - **Error or warning:** if the task status is changed to error or warning.
    - **On start:** when the task is started.
- **Calculation**: calculation type to be applied to the field. This is useful to convert the values of the fields.
- **Class**: this is to indicate to other computer systems the type of the field. Please see *integration with other systems*.
  - **Generic:** General purpose field class.
  - **Unit of Meas.:** Unit of measure. If informed, prevails over the unit of measure of the process.
  - **Cycle T.:** Cycle time. Is informed, prevails over the one of the process and the PSR relation.
  - **Cycle Q.:** Cycle quantity or cycle units. Is informed, prevails over the one of the process and the PSR relation.

- **Lot:** inform about the material lot.
- **Enterprise:** stores the corresponding B2MML tag.
- **Site:** stores the corresponding B2MML tag.
- **Area:** stores the corresponding B2MML tag.
- **ProcessCell:** stores the corresponding B2MML tag.
- **Unit:** stores the corresponding B2MML tag.
- **ProductionArea:** stores the corresponding B2MML tag.
- **WorkCell:** stores the corresponding B2MML tag.
- **ProductionUnit:** stores the corresponding B2MML tag.
- **StorageZone:** stores the corresponding B2MML tag.
- **StorageUnit:** stores the corresponding B2MML tag.
- **WorkCenter:** stores the corresponding B2MML tag.
- **WorkUnit:** stores the corresponding B2MML tag.
- **EquipmentModule:** stores the corresponding B2MML tag.
- **ControlModule:** stores the corresponding B2MML tag.
- **Equivalence:** every production counting is equivalent to how many pieces. For example: 1 unit (a box) is equivalent to 6 pieces.
- **Unit Conversion:** visual conversion from production units to visual units.

- **Length**: maximum length of the field.
- **Calculated**: indicates if the field will be calculated by a UDL.
- **Locked**: indicates if the field cannot be changed by the user.
- **Inherits**: indicates if the field inherits its value from the field with the same variable name of the previous production record.
- **Rel button**: relates fields.
  - **Field** to relate
  - **Value** for the specific relation.



How to configure a field step by step

This procedure allows to **add fields** to a result or to an autocontrol task.
Defining **fields** allows to adjust the system to any sector. For further information, please check Fields.

Next we will develop two examples where defining fields would be necessary.

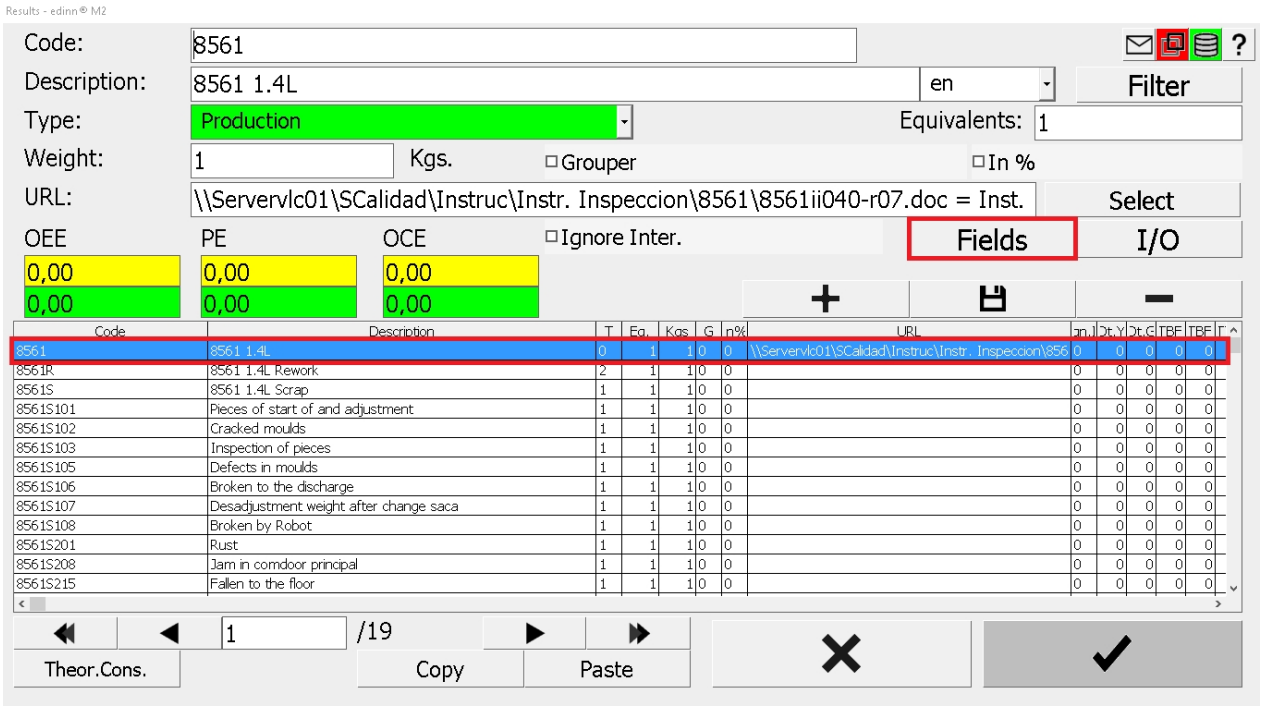
Adding fields to Results

The following company needs to indicate the color and the quality finish for every result it generates. Therefore it will be necessary to add these two fields to each result type. The following steps are necessary to do so:

In [**Click to configure**] select [**Results**].

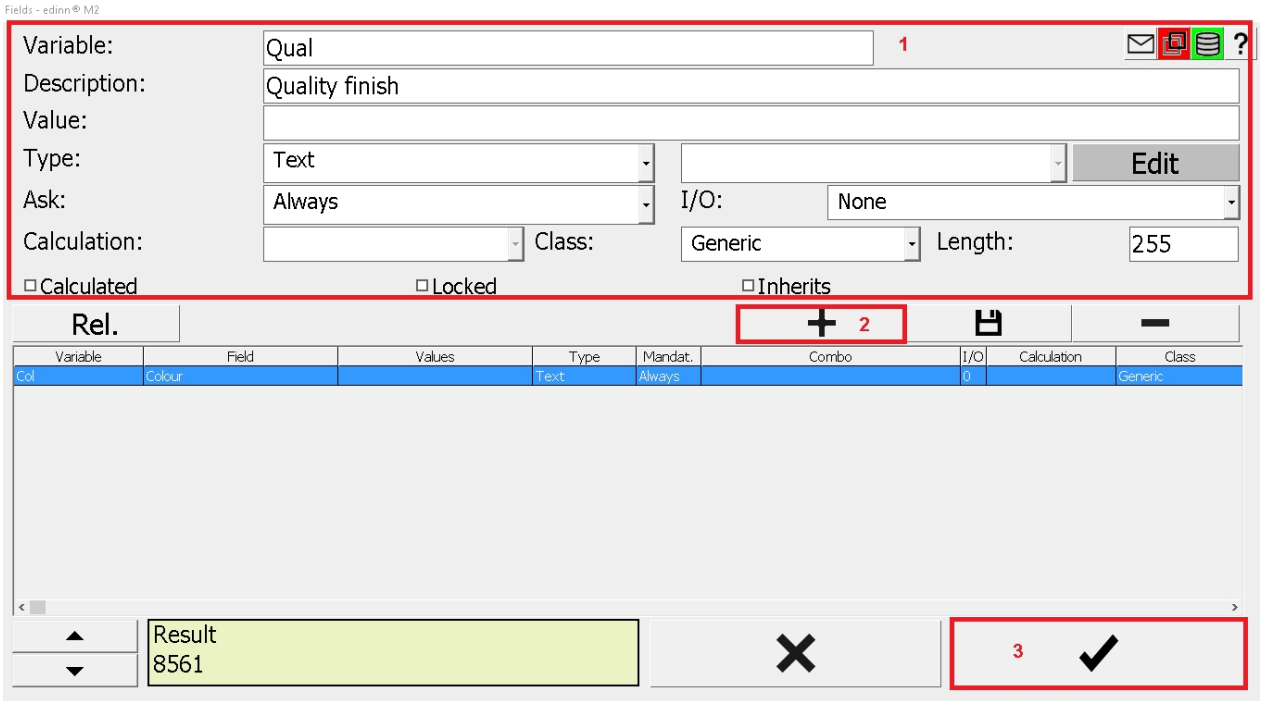
Step 1. Select result

Select the row of the result on the table below and press the button **Fields.**



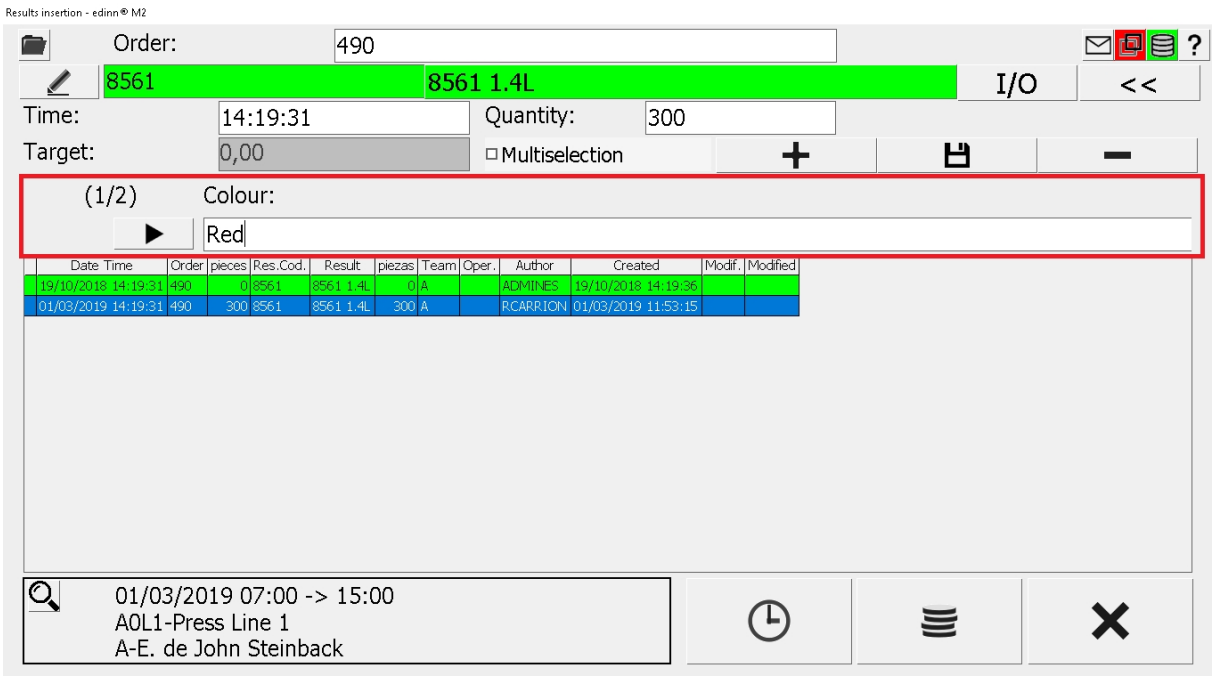
Step 2. Configure the field

1. Fill each field with the information it needs. For further information about each field, please check Fields.
2. Press ✚ to insert the new field.
3. Press ✔ to confirm the changes.



Step 3. Users fill the new fields

Now, users can put the required information in the Results screen.



NOTE: Please see Results configuration for deeper configuration.

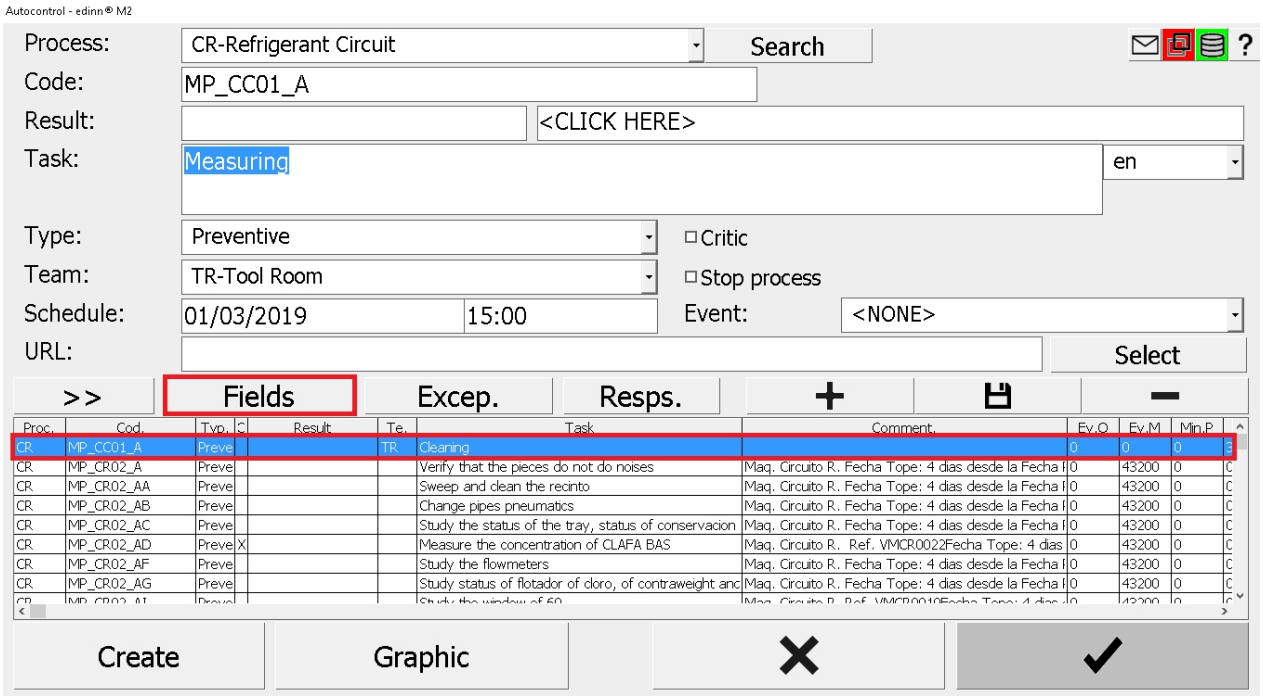
Adding fields to Autocontrol tasks

The following company needs to take data about the temperature of the product. Therefore it will be necessary to define an specific field for the autocontrol grid. The following steps are necessary to do so:

In [**Click to configure**] select [**Autocontrol**].

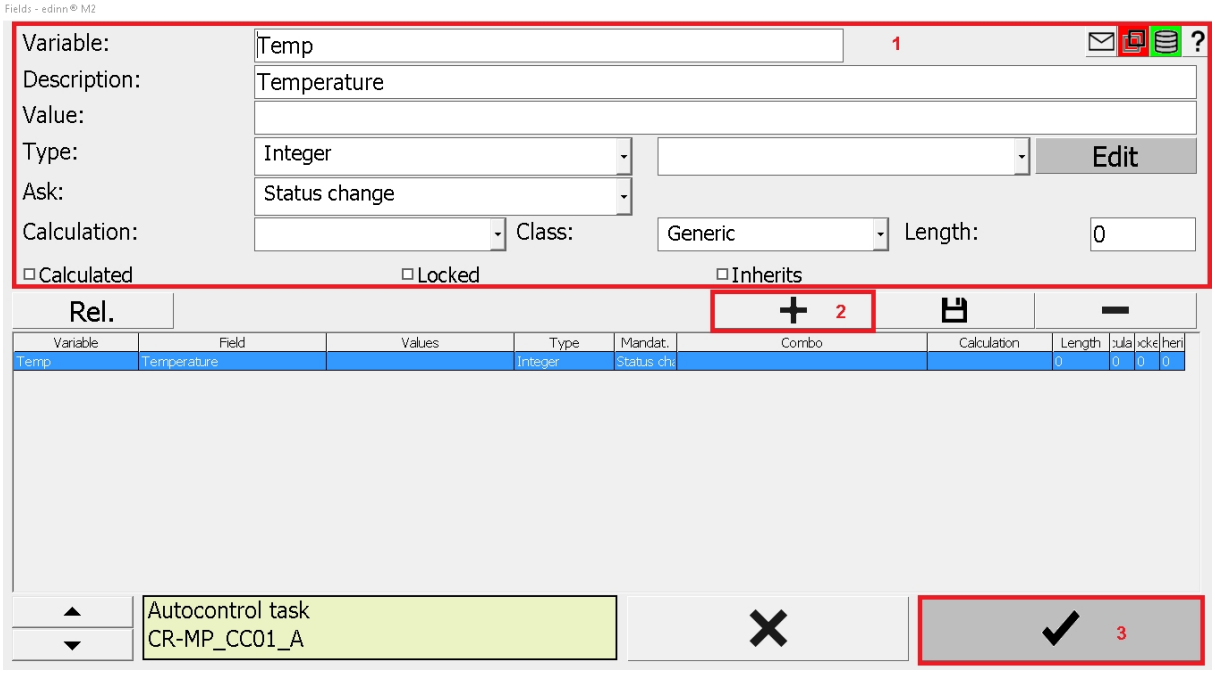
Step 1. Select autocontrol task

Select the row of the task on the table below and press the button **Fields.**



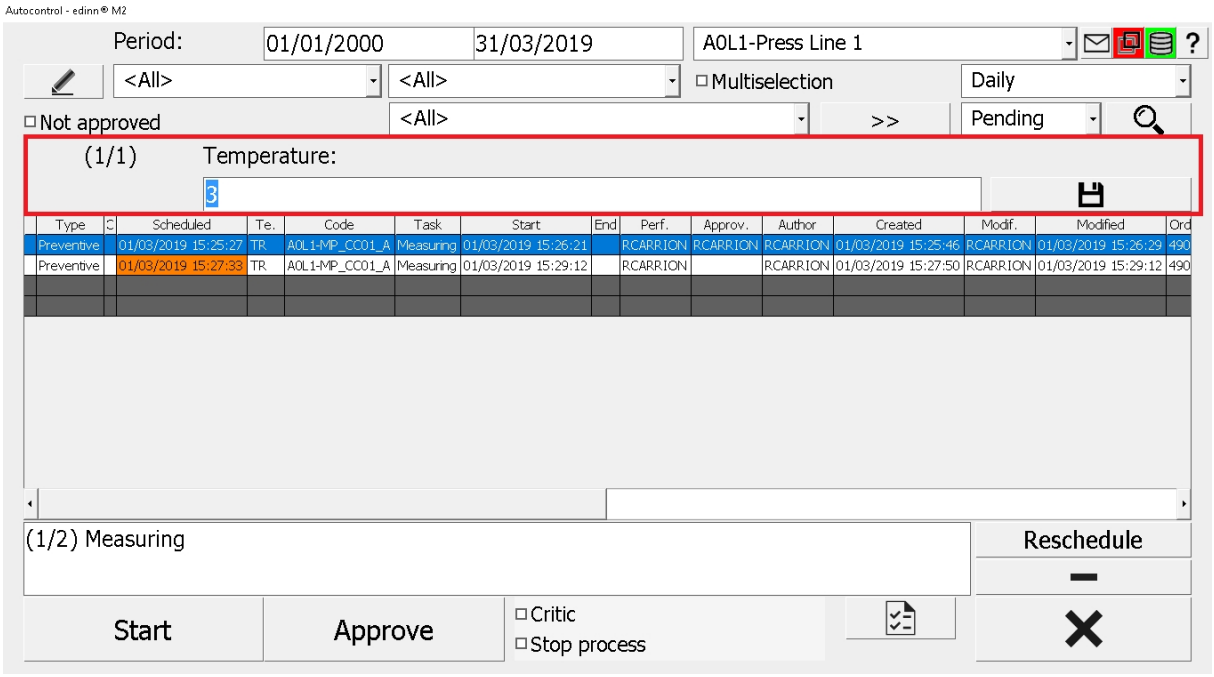
Step 2. Configure the field

1. Fill each field with the information it needs. For further information about each field, please check Fields.
2. Press ✚ to insert the new field.
3. Press ✔ to confirm the changes.



Step 3. Users fill the new fields

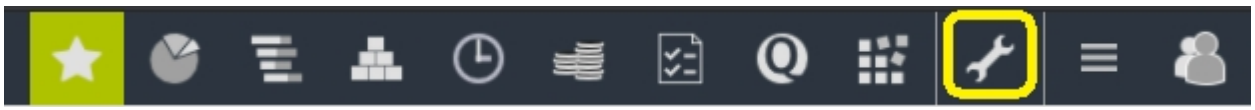
Now, users can put the required information in the Autocontrol screen.



NOTE: Please see Autocontrol configuration for deeper configuration.

Web wizard

The configuration web wizard is only shown for those users who have **administration** roles. It is accessed from the web reports in the options tab.



If you do not see this options, most probably it has been disconnected from the Console.

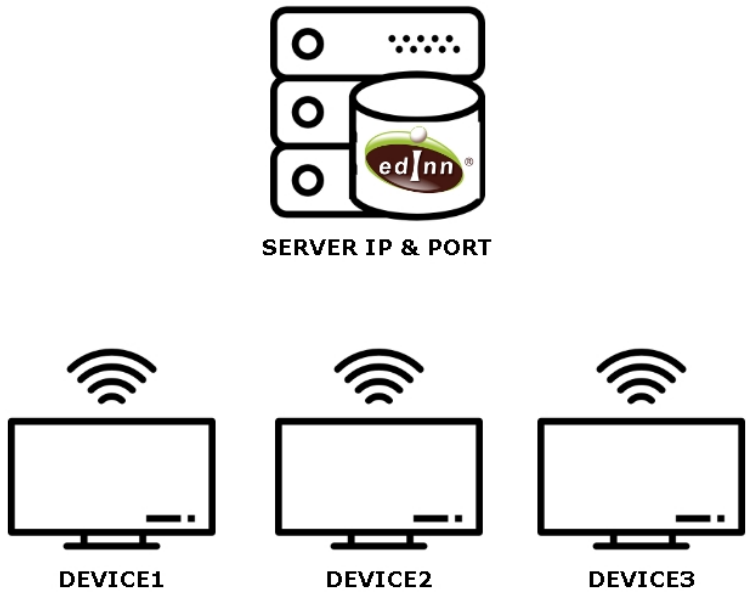
This wizard allows you to configure the system in a few steps.

This wizard is intended for basic configuration. For an advanced configuration please download, install and use the terminal application.

WARNING: This wizard is intended to be used to configure new systems as it could delete detailed configuration that your system already has. You can desactivate it from the server Console.

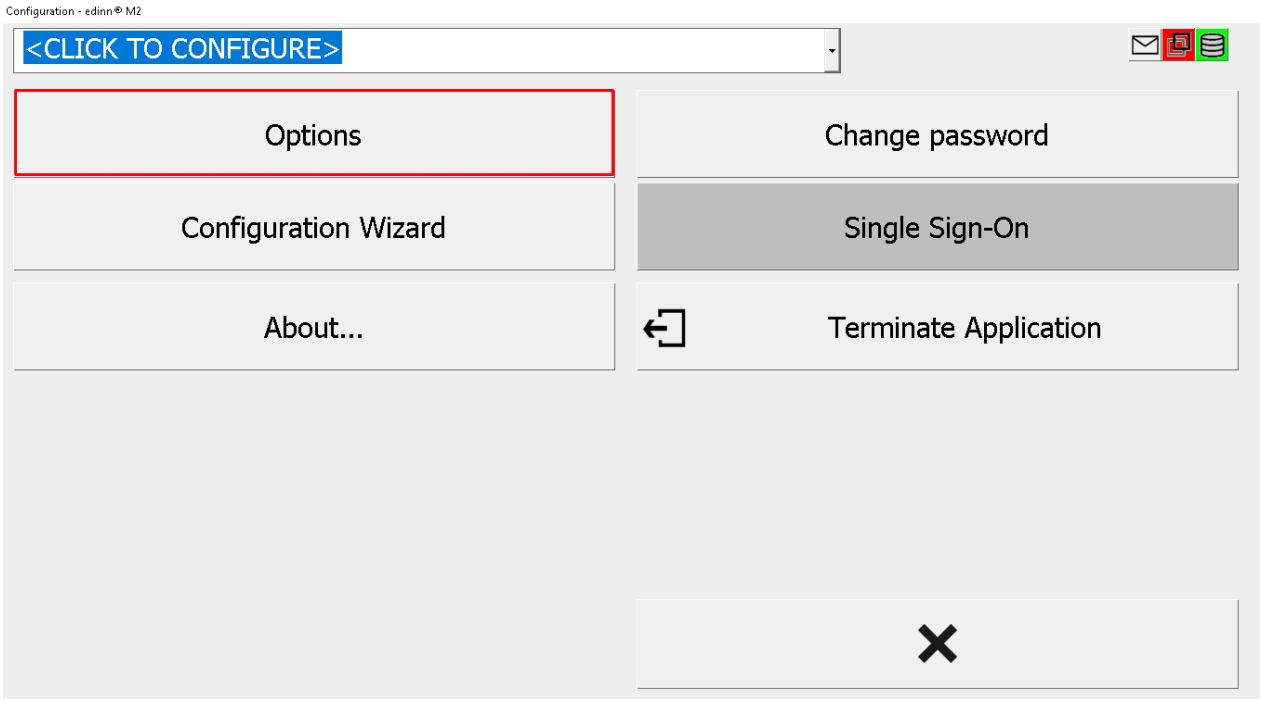
Options

This configuration must be carried out for each terminal in the plant, since each device is informed of the IP of the server to which they must connect.



Step 1. Access General Options

1. From the configuration window, click on [**Options**]

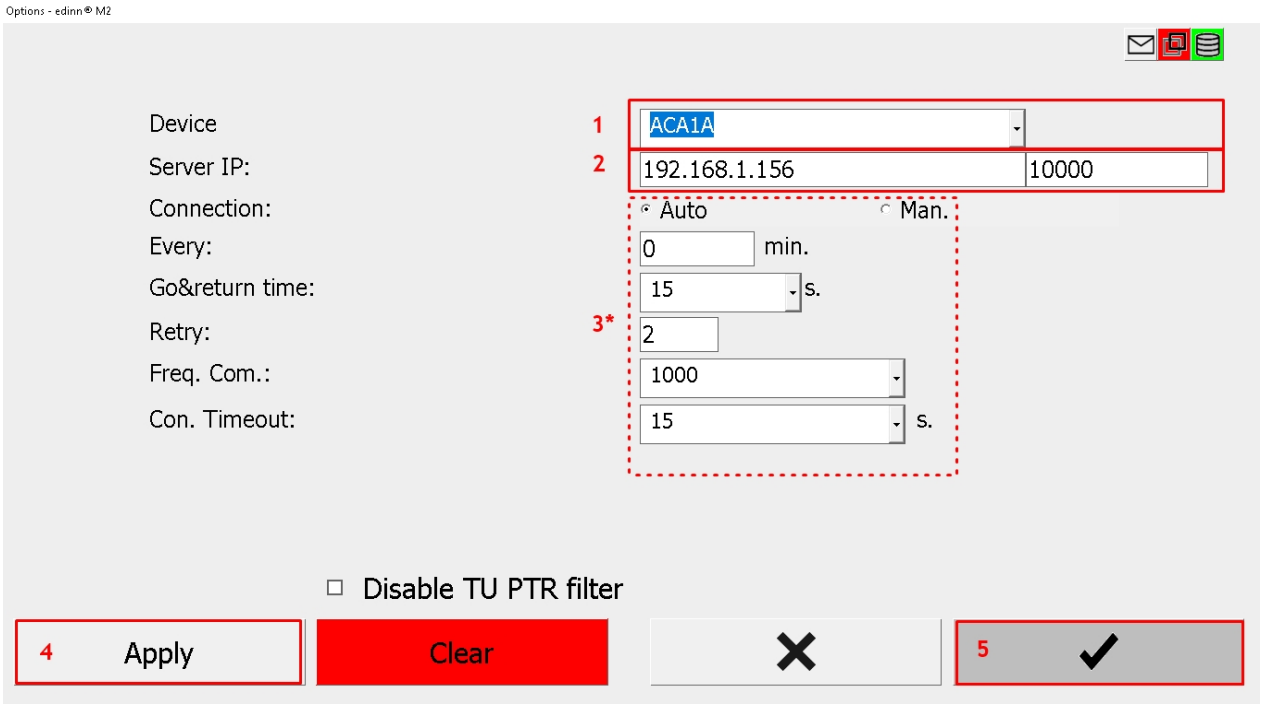


Step 2. Fields configuration

1. **Device:** choose the device you want to use to login on the terminal you are configuring.

RECOMMENDATION: relate the name or identifier with the machine that monitors or the area in which it is located.

2. **Server IP:** Enter the IP and port of the server where edinn has been installed for the terminal to connect.
3. **Optional:** these general parameters are predetermined and are optionally configurable, they refer to the type of connection.
4. Press [**Apply**] to apply the changes.
5. Press ✓ to make the changes effective.



Server

Introduction

What is edinnM2 Console?

edinn M2 Console is an application to configure the general parameters for all the services. Allows to manage the following services:

- **A SQL database**: stores all the edinn M2 data.
- **Central**: is the "orchestra director" of the system.
- **Monitor**: is responsible to gather information automatically.
- **Reports**: provides the web reporting tool.
- **Supervisor**: supervises the whole system and informs if any error or warning occurs.
- **eCloud Sync**: keeps the local system synchronized with other eCloud edinn M2 systems.

Drivers

Drivers are programs that interface between a source of data and the edinn system. Current drivers are:

- **edinnBOX_1**: specifically designed to work with edinn monitoring BOXes (known as eBOXes).
- **edinnHIP**: suitable to receive and send data through the edinn Hardware Integration Protocol (HIP).
- **edinnSIM_1**: suitable to receive and send data from and to simulations (FlexSim and others).
- **Manual**: gathers data from the work of the user with the edinn® M2 Terminal. Does not require any additional hardware.
- **OPCDA**: reads or writes data from and to an OPC DA Server. Requires an OPC Server compatible with additional hardware.
- **OPCUA**: reads or writes data from and to an OPC UA (Universal Architecture) Server. Requires an OPC Server compatible with additional hardware.
- **Vision**: reads data from a camera installed on a local (to the computer that executes the Monitor Service) USB port.
- **Vision IP**: reads data from a TCP/IP compatible camera.

Up to this point, the drivers above have been confirmed to be enough for the majority of the projects. For more information, please read here.

NOTE: If you need a specific driver for your application please refer to the technical support team.The other drivers are left for compatibility purposes but should not be used as they are deprecated.

Counters and signals

Although the elements that are monitored by the Monitor service are generally referred as signals, there are 2 types of real signals:

- **Counters**: they are read periodically (indicated in the 'Check status every' field of the Monitor tab) by the Monitor service. No matter how frequently these signal change, the amount will be aggregated.
- **Signals**: they are read whenever they change. By this, we get sure that every value of the signal is stored into the system.

Apart from the real signals, the user can also define virtual signals that will be calculated upon the need of the project:

- **UDL signals**: this signals are managed into the UDL module, or User Defined Logic, useful to develop new functionality on top of the system.

WARNING: to use Pulses signals instead of Counters signals to count results or consumptions it is not recommended since the system might not read these signals fast enough or possible failures in the conexion with the OPC Server could result in an information lost.

eBOX

eBOX is the name of the standard hardware provided by edinn®. There are different models as they cover all the needed functionality to automatically monitor data from different sources and for different purposes: industries, solar panels, electricity consumption, etc.

eBOXes have at least 2 green LED lights indicating:

- **Power on**: the eBOX is powered on.
- **Data on**: the eBOX is correctly sending data to an edinn® system.

Notifications

Email notifications can be sent **inmediately** since the systems generates them at the same moment that they occur, but some problems might happen:

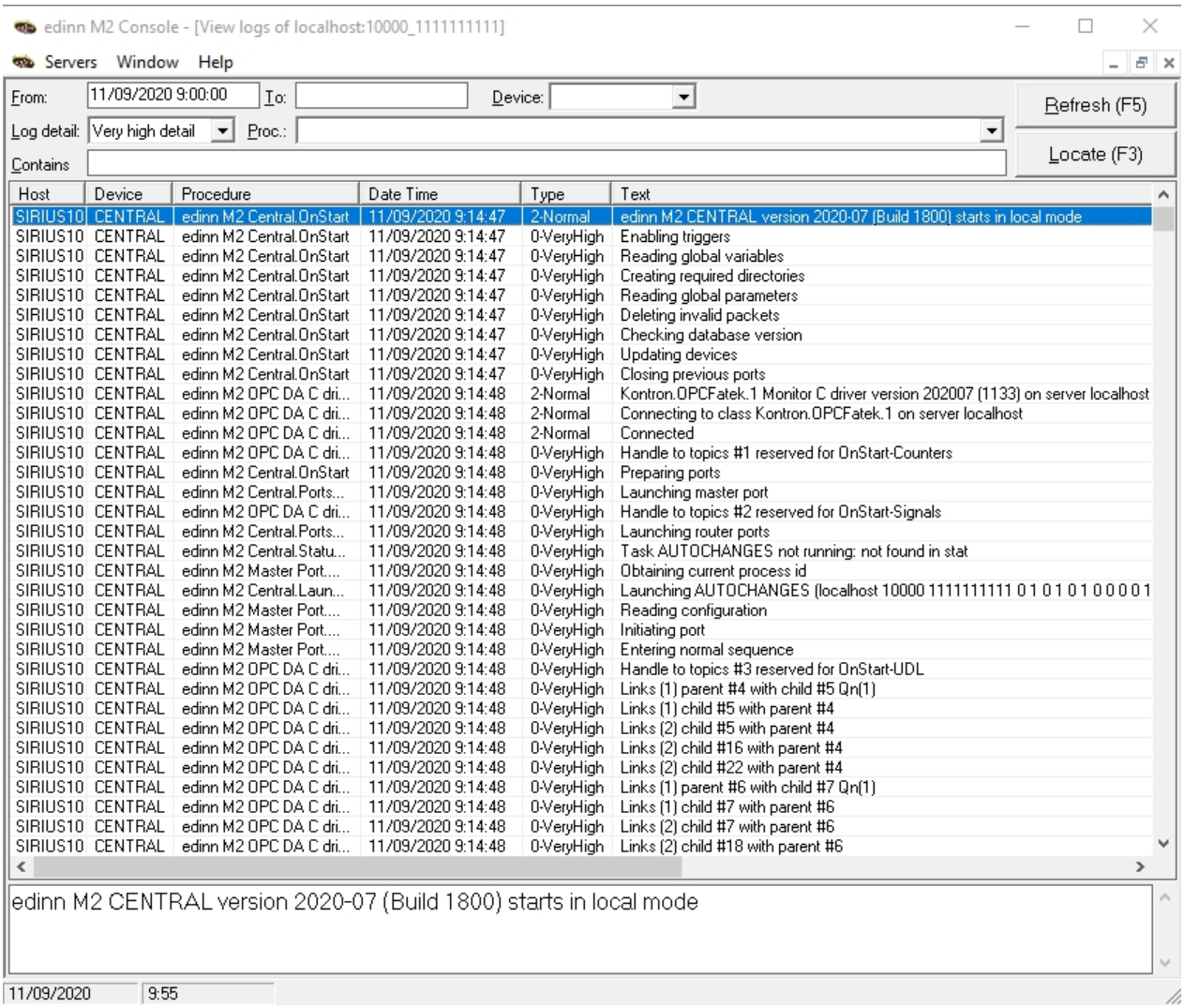
- A delay of certain seconds generating the notification.
- A delay of up to 1 minute when trying to sent it to the network.
- A delay within the network.

In normal circumstances the user will receive the email in a total of few seconds.

View logs button

Once in the console, you will see the ***Views logs*** button at the bottom left corner of the window. The functionality of this tab is to see in a chronologically order the different events that have affected the system.

Although logs are periodically and automatically deleted, you can delete them manually and permanently with the ***Clear logs*** button.



Available filters to search in the log:

- **From / To:** selet the period to search.
- **Device:** select the device to search.
- **Log detail:** Please see General tab.
- **Proc.:** select the process to search.
- **Contains:** type the text to filter your search (Refresh [F5] button) or to locate (Locate [F3] button) a record in the list.

> **NOTE**: if you experience any problem with the system, search on this log for the reason and, if the system is not writing on this log, search in the Windows Event Viewer. This system is designed to whether inform on this system log or in the Windows Event Viewer.

Console

To configure an edinn® M2 Server you need to use the **edinn® M2 Console**. After the installation and setup of the server, you will have the application on your Windows.

Click on Start, All Programmes, edinn® M2 and edinn® M2 Console. You will see:

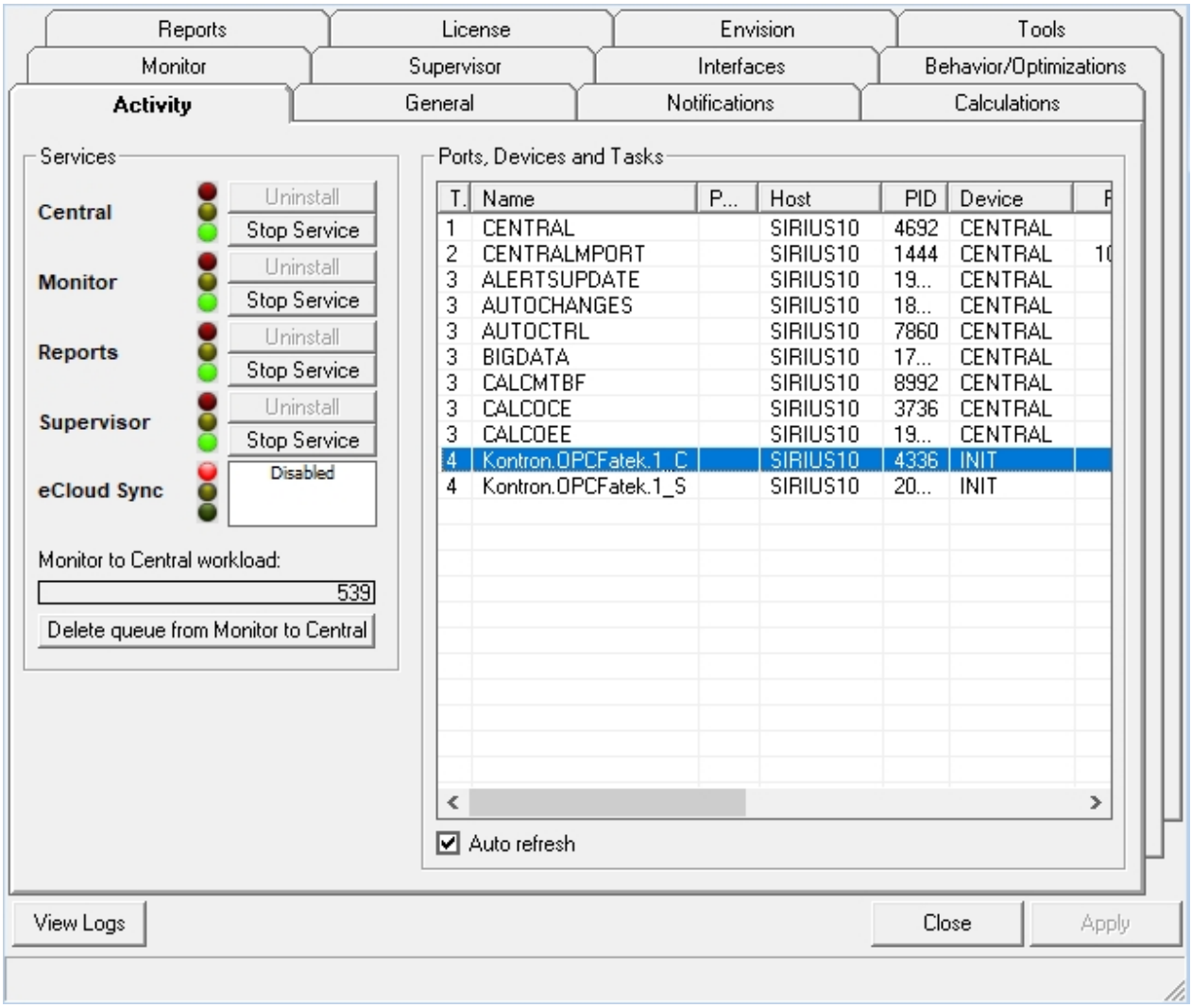
- **Host:port**: IP and port address of MySQL.
- **Company Id**: company identifier, it is generated automatically by the web configuration wizard when creating a company in the installation of the server.
- **User**: your MySQL administrator user.
- **Password**: your MySQL administrator password.

Once you access the console, you will see:

1. The different tabs to configure the server.
2. The View Logs button, which will give you access to browse the system general logs.

Activity

Once in the console, you will see the Activity tab, which shows the activity and status of the system:



Functionality is as follows:

- **SERVICES**: allows viewing the status of all services:
    - **Services 'traffic lights':** indicate the status of the service:
        - **Green:** Working
        - **Yellow:** Stopping

        - **Red:** Stopped
        - ●: Uninstalled
    - **Uninstall / Install:** uninstall or install the service on the server specified above.
    - **Start / Stop device:** when the service is installed, start or stop the service.

ATTENTION: If the Supervisor service is running, it will try to keep running all the installed services. Supervisor service only supervises services that are installed in the same computer. Therefore, if you install different edinn M2 Services in different computers, you need to install and run a supervisor service in every computer.

NOTA: The Reports service cannot be operated in companies running in the edinn cloud (ecloud) as it could affect other companies. If you need to operate it please contact support.

    - **eCloud Sync**: can not be installed, uninstalled, started or stopped from the console. Please refer to Behaviour in order to see the basic configuration of this service, although manual intervention is needed by the edinn support team or an authorized partner.
    - **Monitor to central workload**: indicates the current queue of pending messages between the services Monitor and Central. **It will be shown in yellow if the unprocessed data from the Monitor service are delayed more than 5 minutes and blinking in red when more than 10 minutes**. It will also show inside the date and time of the last processed message, which indicates how updated is the information that the users have.

      In general conditions, the Central service consumes data faster than the Monitor service generates it, and therefore, there should not be any problem associated with this queue. But, in certain cases:

        - When an edinn M2 server goes out of license: the information gathered by the monitor is not discarded but the central service does not process it. Therefore the messages queue will grow rapidly. Once the server is licensed again, all the pending data from the Monitor service will be processed.
        - When the Monitor service is configured to generate data very frequently: the central service cannot process it on time.
    - **Delete queue from Monitor to Central**: deletes permanently (they cannot be recovered) the data generated by the Monitor service which are pending to process by the Central service. This will cause a permanent data loss. The only documented case where this function is useful is when an edinn server has been a lot of time without license and, when licensed back, it would need a lot of time to work again as it is processing very old monitoring data which the users do not want.

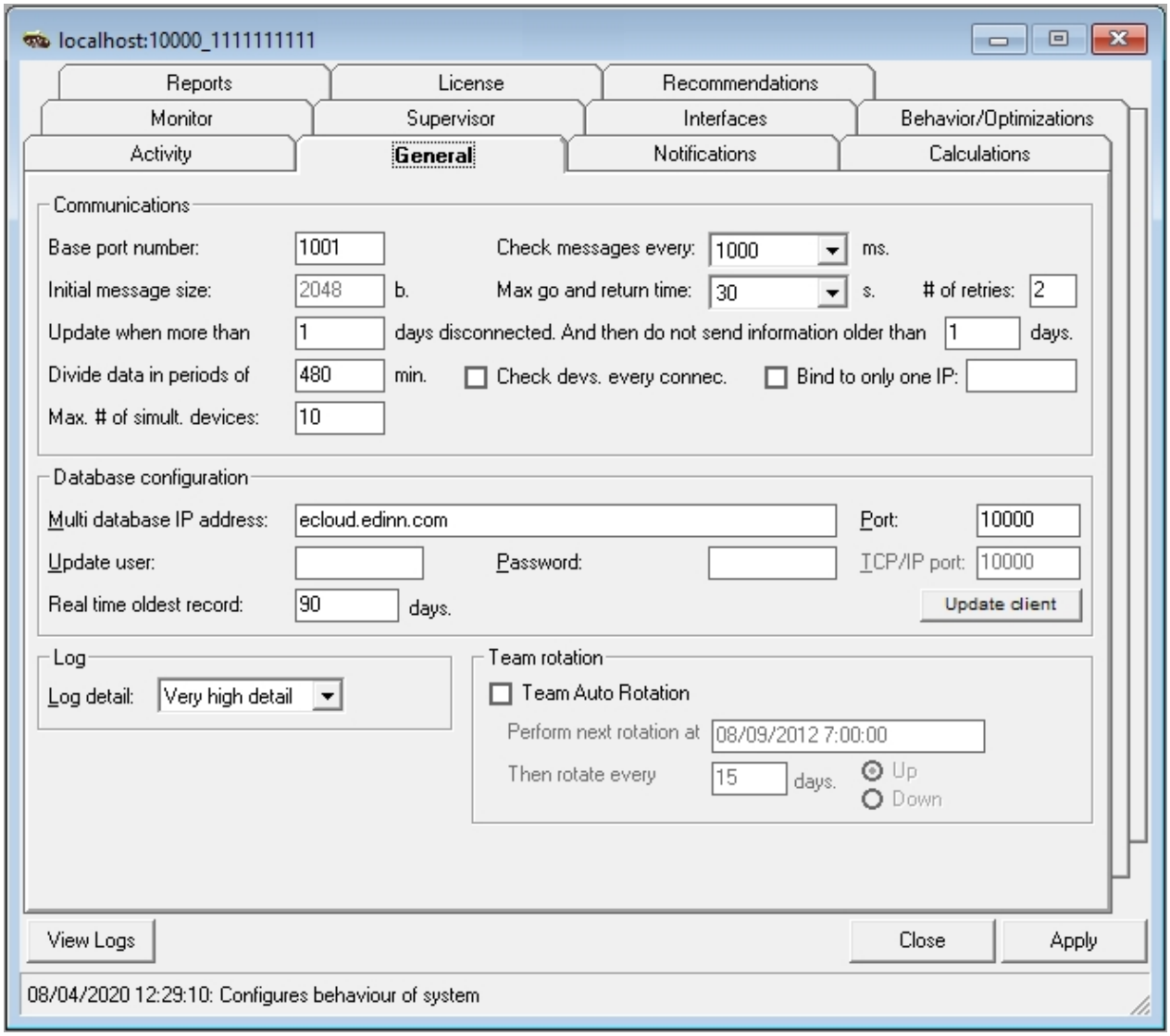
- **PORTS, DEVICES AND TASKS**: shows actual status of the terminals connected to ports

(devices) and the tasks being executed.

NOTE: The database service is not seen with a traffic light, because the database is validated by the connection of the console itself (if the console connects, the database is working). If the database fails while running the console, an error message will be shown on the status bar (at the bottom of the console: "Single point to manage edinn M2 Server").

General

Once in the console, you will see the General tab, which has general configuration parameters:



Functionality is as follows:

- **COMMUNICATIONS:**
    - **Base port number:** from which port the edinn® M2 Terminals will connect.

- **Check messages every (ms):** time in milliseconds between a bunch of messages (from the monitor or from terminals) are processed.
- **Initial message size (b):** can not be changed, reserved.
- **Max go and return time (s):** total time for a message to go from the server to a terminal.
- **# of retries:** number of retries to send a message to a terminal.
- **Update when more than ... days disconnected:** reserved only for off-line communications.
- **And then do not send information older than... days:** reserved only for off-line communications.
- **Divide data in periods of... mins:** reserved only for off-line communications.
- **Check devs. every connec:** reserved only for off-line communications.
- **Bind to only one IP:** in case the server computer executing the Central service has more than 1 Network Interface Card, you can specify on which the edinn® M2 Terminals will be found. Type the IP address of the desired Network Interface Card.
- **Max. number of simult. devices:** maximum number of simultaneous terminals.

- **DB CONFIGURATION**:
  - **Multi database IP address:** IP address of the main database which contains the directions to the companies' databases.
  - **Port:** port of the main database which contains the directions to the companie's databases.
  - **Update user, password and TCP/IP port:** in order to use the automatic update functionality of the edinn® M2 Server, you need to specify here a user and password that has permission to alter the database structure and the port where the database is accessible.
  - **Real time oldest record (days):** indicate here the number of days from now to the past, that the terminals win32 version will be able to view. It also determines the number of future days that the automatic calendar of the system will contain.
  - **Update client:** updates the database client user which is the one used by the terminal win32 version. Normally you do not need to use this button if you use the automatic updater of the server versions. After a server version upgrade, if the automatic updater of the server indicates so, or if it is indicated in the Revision notes, then you need to click this button after the upgrade. If your version needs you to activate this button, please remember that you must reset

(answer Yes) the user at least once in your server, and then click again on this button, without resetting the user (answering No), for each company other that you have in your server.

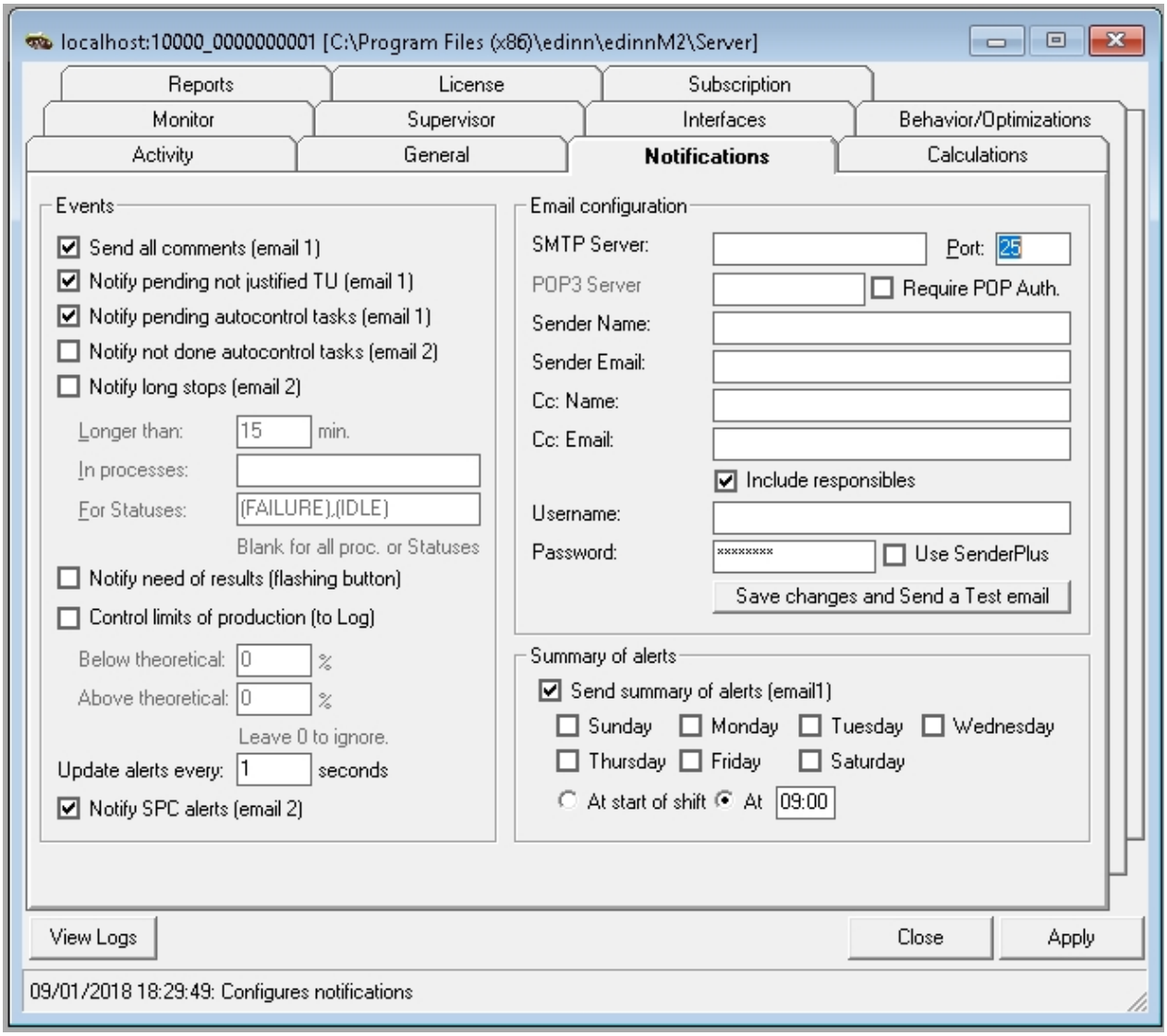
- **LOG**: changes the log detail.
    - **Normal:** use this level of log detail when the edinn® M2 System is working correctly and you do not need detailed information.
    - **High detail:** use this level of log to have more information on the logs.
    - **Very high detail:** use this level of log when the edinn® M2 System is not behaving as you expect and need to have an explanation of what is exactly doing.

WARNING: If left for a long time, "High detail" and "Very detail" options can generate a database overload and therefore problems on the system maintenance and on the response time for the users. To prevent these problems the system will automatically change to Normal after 7 days in "High detail" or "Very detail".

- **TEAM ROTATION**: allows that if, for example, team A is associated with shift 1, then rotation will change team A to be associated with shift 2 (down rotation) or with the last shift (up rotation).
    - **Team Auto rotation:** activates or dis activates the team auto rotation.
    - **Perform next rotation at:** indicate the date and time when the next team rotation should be performed.
    - **Then rotate every:** number of days after the next rotation for the subsequent rotations.
    - **Up / Down:** if the rotation must be incrementing (down) or decrementing (up) the number of the shift.

Notifications

Once in the console, you will see the Notifications tab, which has the configuration for the notifications that the system performs:



Functionality is as follows:

> **NOTE:** All notifications will be sent to the contact info (usually email) indicated in the resources configuration for, what is available first: resource associated to a process, process responsible, team responsible or shift responsible.

- **EVENTS**: indicate which events are notified.
    - **Send all comments (email 1):** if marked,
        - All comments created by the users will be notified to the destination resource, if email 1 is configured.
        - Certain events related with scheduling will be notified to the corresponding resource, if email 1 is configured.
        - The reason for being changed to an stopped status will be sent to those resources which have email 1 configured and whose associated process was put into an stopped process by the monitor.

- **Notify pending not justified TU (email 1):** if marked, and if the process is configured with the Lock parameter different than 0, then all statuses which are about (in the last quarter of the period indicated in the Lock parameter) to be locked, will be notified.
- **Notify pending autocontrol tasks (email 1):** if marked, all autocontrol tasks that become RED, and have less than 480 minutes of tolerance, will be notified to the responsible of the autocontrol task.
- **Notify not done autocontrol tasks (email 2):** if marked, all autocontrol tasks that become undone will be notified to the responsible of the autocontrol tasks.
- **Notify long stops (email 2):** if marked, all not productive statuses that are longer to the time specified below, will be notified. This notification will only occur when the status is active.
  - **Longer than (min):** specify the time in minutes.
  - **In processes:** specify the list of processes you want to supervise. If left blank it will apply to all processes. An example to indicate to supervise only processes PRO1 and PRO2, will be: PRO1, PRO2
  - **For statuses:** specify the list of statuses you want to supervise. If left blank it will apply to all statuses.
- **Notify need of results (flashing button):** if marked, it makes the results button of the terminal to flash in red whenever the process should be generating results. This notification does not generate an email.
- **Control limits of production (to log):** if marked, a message will be written in the log indicating when production is below or above the nominal maximum capacity of a process. This notification does not generate an email, unless the notify warnings are activated at the supervisor configuration.
  - **Below theoretical (%):** percentage below the theoretical production from which a message will be written in the log.
  - **Above theoretical (%):** percentage above the theoretical production from which a message will be written in the log.
- **Update alerts every (seconds)**
- **Notify SPC alerts (email 2):** if marked, then SPC alerts will be notified.

- **EMAIL CONFIGURATION**: configure your email parameters. Use the button 'Save changes and send...' to test your system configuration. If well configured, if the server has access to internet, etc. you will receive an email immediately.
  - **SMTP Server:** indicate the network protocol used for the exchange of email messages between the system and other devices. For more information, please see: https://en.wikipedia.org/wiki/Simple_Mail_Transfer_Protocol

- **Port:** port of the SMTP Server.
- **POP3 Server:** is an application-layer Internet standard protocol used by local e-mail clients to retrieve e-mal from a remote server over a TCP/IP connection. For more information, please see: https://en.wikipedia.org/wiki/Post_Office_Protocol
- **Require POP Auth.:** if a Server POP autorization is required.
- **General parameters of notifications:**
  - **Sender name / email.**
  - **Cc: Name / email.**
- **Include Responsibles:** if marked, the responsibles of a task will be notified in the email #1.
- **Username:** of the SMTP Server.
- **Password:** of the SMTP Server.
- **Use SenderPlus:** to send emails with SSL (Secure Sockets Layer), cryptographic protocols that provide secure communications over a network.

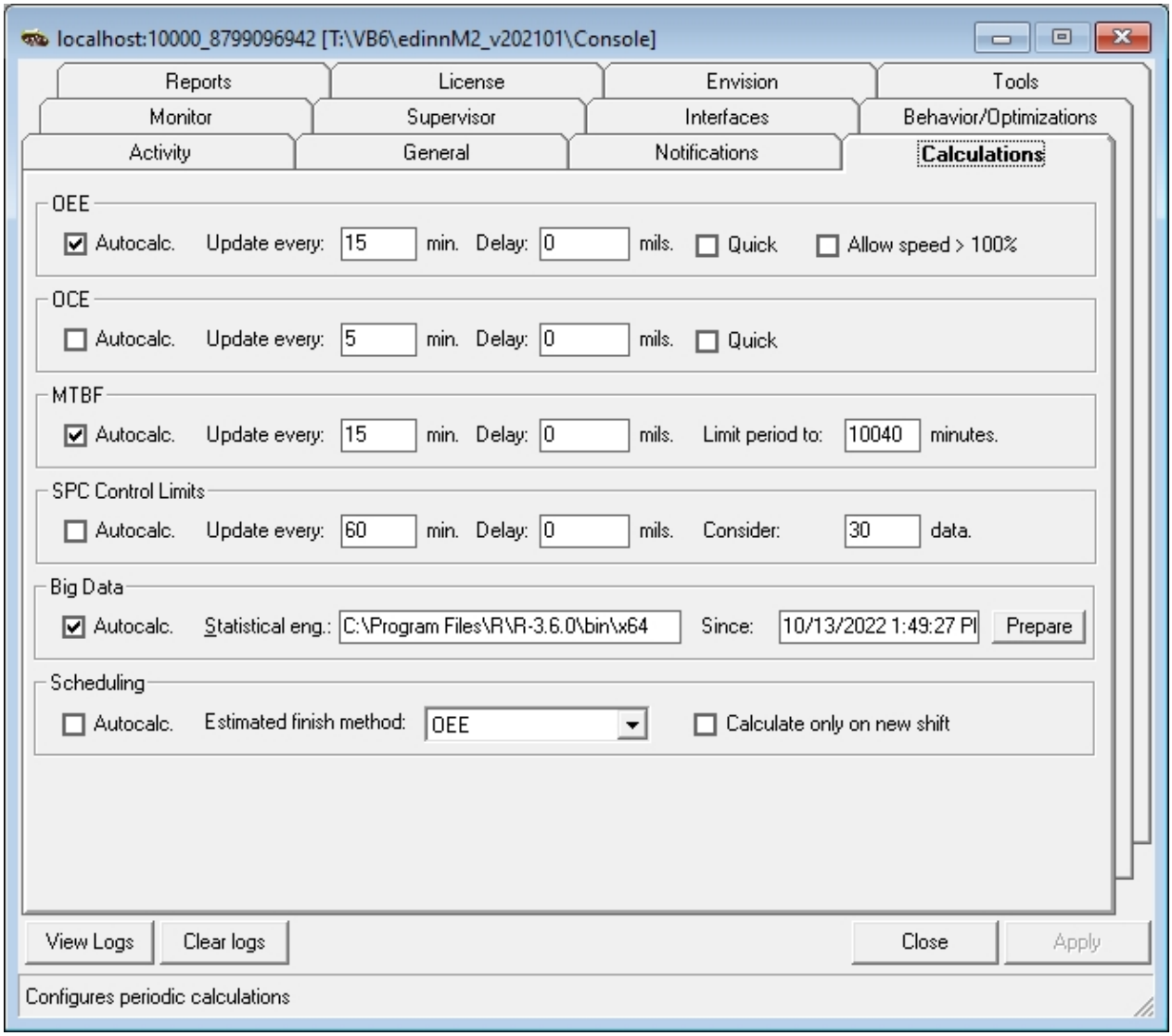
NOTE: to configure the email 1 and email 2 please see: resources configuration.

- **SUMMARY OF ALERTS:** to send to the responsibles a summary of the current status of all the alerts.
  - **Send summary of alerts (email1):** if marked, the summary of alerts will be sent to all the personnel that has configured the email #1.
  - **At start of shift / At:** if the summary will be sent at the start of the shift or at a determined hour.

NOTE: this is a general configuration, individually each user can change the parameters for the summary of alerts, please see: resources configuration.

Calculations

Once in the console, you will see the Calculations tab, which has the configuration of the periodical calculations performed by the server.



Common options:

- **Autocalc:** if marked, automatic calculation will be performed.
- **Update every (min):** indicate the number of minutes to wait before starting the calculation again.
- **Delay (miliseconds):** field used to delay the calculations to reduce the server workload.
- **Quick calculations:** if marked performs a less detailed calculation which is useful to reduce the workload of the server.

WARNING: In general conditions it should be **not** marked. Marking this option could cause from 5% to 10% level of error on the ratios. It should be only marked on servers which have sever performance problems.

Functionality is as follows:

- **OEE**: calculates OEE for the predefined periods, for the API and the operation terminals

to work faster.

    - **Allow speed > 100%**: if marked, it will allow the speed component (also known as performance) of the OEE to be greater than 100%. This options also affects the reports and, in general, to the whole system.

- **OCE**: calculates OCE for the predefined periods, for the API and the operation terminals to work faster.
- **MTBF**: calculates MTBF for the predefined periods, for the API and the operation terminals to work faster. Here you specify how frequently this information is updated for the terminals. Same parameters as before, except:
    - **Limit period to:** indicate to limit the analysis to a certain number of minutes. This means that, for example, the users will have the MTBF table for the last week (10080 minutes), or for the last day (1440 minutes), etc.
- **SPC CONTROL LIMITS**: SPC control limits can be automatically calculated and adjusted. Same parameters as before, except:
    - **Consider:** indicate the number of SPC data to adjust the SPC Control limits.
- **BIG DATA:** big data prepares data to to improve the speed of operations and the performance of data analysis. Same parameters as before, except:
    - **Statistical engine**: indicates the path to an additional statistical engine.
    - **Since**: indicates the date and time since when the system (reports and other parts of the system) will use Big Data information instead of regular data.
    - **Prepare:** normally it is not necessary to use this button because the system prepares data automatically when the Autocalc option is activated. This button will: check if your license if valid to use Big Data, require from you if you want to generate Big Data information from one process, in which case please type only the code of the process, or leave blank for all the processes; and later will propose you a period to generate Big Data information.
- **Schedulling:** groups the options used to make automatic periodic calculations of the scheduling modul.
    - **Estimated finish method:** sets the method that will be used to estimate the time required to complete current orders and their dependent orders:
        - **Nominal capacity:** With this method, the pending time will be calculated by multiplying the amount pending by the target speed or nominal capacity.
        - **OEE (default value):** with this method the pending time will be corrected by dividing it by the OEE of the longest period configured in edinn. This will generally give a better approximation for the order pending time.

The expected completion date of the orders will be calculated by adding the calculated pending time to the current moment, taking into account that there may be periods of time scheduled as non-production times, such as non-working shifts, holidays, scheduled maintenance, etc.

- **Calculate only on new shift**: Performs the calculation at shift change instead of doing it continuosly.

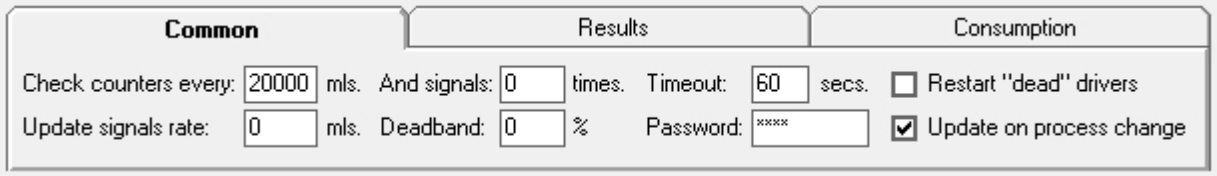
Monitor

Once in the edinn® M2 Console, you will see the Monitor tab, which has the configuration of the Monitor service.

WARNING: All the changes will not be applied until the restart of the Monitor Service.

Subtabs

Functionality of the subtabs are as follows:



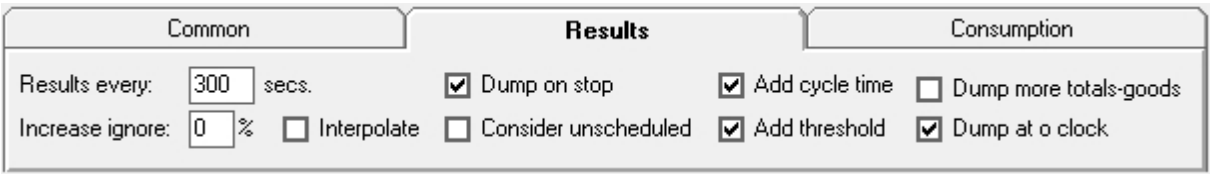
- **COMMON (subtab)**: common parameters for production and consumption.
  - **Check counters every (mls):** makes the system read the counters every this miliseconds.

WARNING: If you are using pulse counters with the methods **edinnHIP**, **edinnManual**, **edinnBOX_1** and **edinnSIM_1**, put here a frequency which reads at least 2 times per counters pulse. For example, if the pulse counter signal lasts 1 second active, then put here 500 miliseconds.

  - **And signals (times):** makes the system read the signals every 'Check counters every' multiplied by this field, miliseconds. If 0 is indicated here, the signals will only be read when they change.

NOTE: Signals work by subscription (except in methods **edinnHIP**, **edinnManual**, **edinnBOX_1** and **edinnSIM_1)**: each time they change, the system registers them. But due to the deadband, it could take a long time without a signal to be registered. Using "And signal (times)", you ensure that the signal is recorded every time it changes more than a certain % of its value, and avery certain time ('**Check counters every**' * '**And signals**' miliseconds).

- **Timeout (secs):** if the driver is more than these seconds without activity, the system will restart it, considering as activity any data received of counters or signals with a change in its value.
- **Restart "dead" counters:** if marked, the system will restart the driver when at least 1 counter is more than the "Timeout" time without activity, considering as activity a change in its value.
- **Update signal rate (mls):** If 0 is not indicated, the system will not capure 2 consecutive values for the same signal in less than this miliseconds. It is useful to ensure the maximum rate of capture of a signal.
- **Deadband (%):** allows to ignore small changes in **all** the signals. For example, a deadband of 10% will ignore changes in signals which are less or equal to 10% of the previous value. This means it will process the signal only when Abs(NewData - OldData) / OldData > Deadband, where Abs means the absolute value.
- **Password:** if a password is required to modify the Monitor configuration. It is generated by the web configuration wizard when creating a company.
- **Update on process change:** generates a standard configuration each time a new product or process is to be configured.

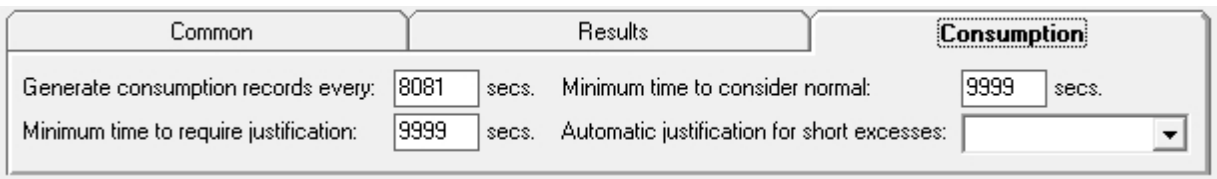


- **RESULTS (subtab):** common parameters for production.
  - **Results every (secs):** number of seconds between production records. For example: if a process generates results every second, if we type 1 sec. on this field, all the results generated in 1 second will be dumped to the database. If we typed 300 seconds on this field, all the results generated in 300 seconds will be dumped **aggregated** to the database. This is to reduce the number of results records in the database. The ratios and calculations are not affected by this value. It only affects to the perception of the user.
  - **Dump on stop:** if marked, a results record will be generated every time a process stops. This is useful for a better analysis of the data since we have a results point on every status change of the process.
  - **Add cycle time:** if marked, to record more accurate results, the system will substract to the date and time at which the system has determined that a process is working, the cycle time of the results. For example: if the process generates the first result at 10.00, but the result needs 1 minute to be generated, then the production status will be marked as started at 09.59, which is more accurate.
  - **Dump more totals - goods:** dumps more frequently scrap and rework calculated

by the difference between totals - goods. Normally it is dumped only then the line has been discharged, but with this marked, it is dumped additionally: at o'clock, when a result or order is changed and at the end of shift. This option could generate **negative quantities of scrap or rework results** when forcing the dump if, since the last dump, the counters of goods grew more than the counters of totals. This should not be a problem as, in a longer term, negative quantities should be compensated with the positive.

NOTA: A line of processes, configured with a counter of totals and later a counter of goods, is considered as "discharged" when more time than the minimum production time has passed between the last pass of a piece by the counter of totals and by the counter of goods, which indicates that all pieces that fit into the line have passed by the counter of goods without any new piece entering through the counter of totals.

- **Increase ignore**: ignores those counters that, after resetting, would generate results exceeding the maximum production capacity by the percentage indicated in this field. For example, puting the value 500 in this field means that, after a counter resets, it will be ignored if would generate results exceeding by a 500% of the possible theoretical production. This field is useful when communications with the data source are unstable.
- **Interpolate:** if marked, if the monitor stops working, and during this time the shift is changed, the system will do a proportional division of the results and the productive time that have occurred since the monitor stopped. This is useful to not assign to the currect shift the results of the previous shift.
- **Consider unscheduled:** if marked, all not scheduled time will be subtracted from the cycle time of the process to generate the next result. For example: if a process has stated that it will produce results in 2 hours, but then later, before this 2 hours pass, it has 30 minutes of "not scheduled" time, then the process will be allowed to generate results in 2,5 hours without considering that the status of the process is stopped and without the need to justify those additional 30 minutes.
- **Add threshold**: if marked, and if **Add cycle time** is also marked, then it will add the cycle time but multiplied by the effect of the **threshold**.This option is useful to avoid the generation of microstops when the production data does not arrive in real time or when the monitor service has been down during a while.
- **Dump at o clock:** dumps results also at every hour (at XX:00).



- **CONSUMPTION (subtab)**: parameters for consumption.
  - **Generate consumption records every (secs):** same as in production.

- **Minimum time to consider normal (secs):** minimum time that a consumption level must be maintained to consider it normal consumption.
- **Minimum time to require justification (secs):** minimum time that a consumption level must be maintained to require justification from the user.
- **Automatic justification for short excesses:** those excess in consumption that do not last up to the 'Minimum time to require justification', are automatically justified with this consumption issue status. This is the equivalent to micro stops in production.

Table

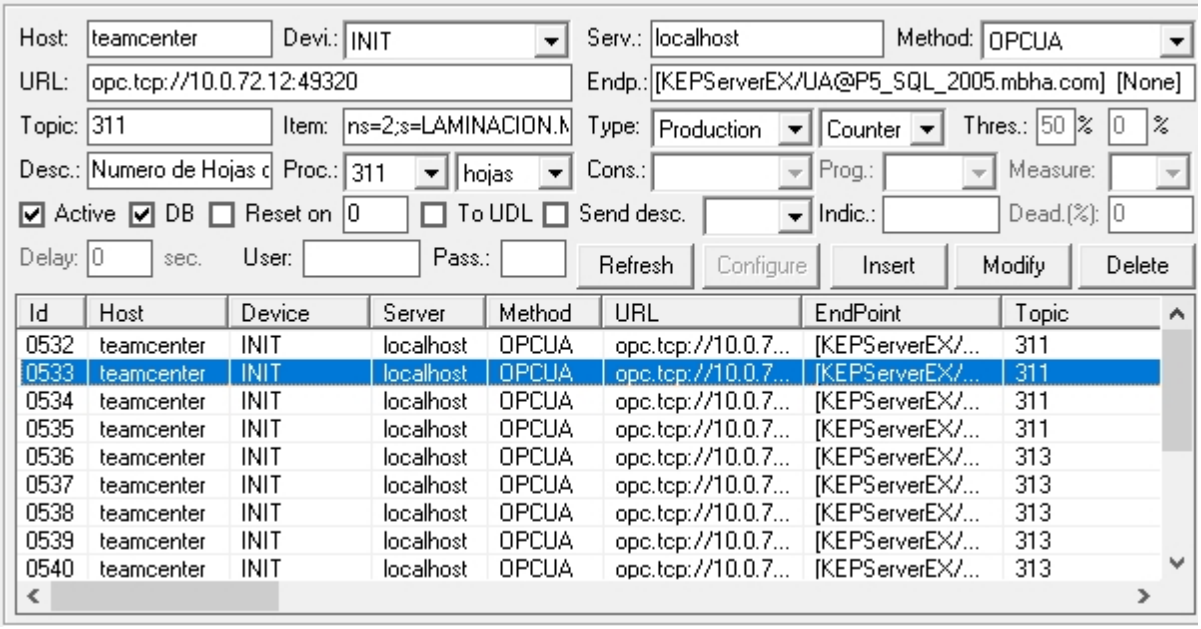


Table functionality to gather information is a follows:

- **Host**: host name of the machine which is executing the Monitor service as shown in the Activity tab.
- **Device**: select the device from the devices configuration (use and edinn® M2 Terminal to configure them) that will be used for the monitor. This is useful to trace certain information later.
- **Server:** IP address of the OPC Server (OPC DA) or name of the computer providing the data (edinnHIP or edinnBOX_1, using edinn OPC Bridge).
- **Method:** Any of the available methods or protocols (edinnBOX_1, edinnHIP, edinnSIM_1, Manual, OPCDA, OPCUA, Vision, VisionIP or MQTT) explained here.
- **URL**: IP address of the OPC Server (OPC UA) or IP Address and port (IP_Address:Port) of the eBOX (for edinnBOX_1).

- **Endp.:** Endpoint (name of the driver for OPC UA), DCOM (Distributed COM for OPC DA) or JSON Template (MQTT).
- **Topic:** tag or word to identify and group signals of the same process or group. It is very important to type here exactly the same word for signals that belong to the same process.
- **Item:** has to be exactly as configured in the OPC Server for the OPCDA: identifies the exact signal or plc to monitorize, can be obtained in the OPC Server proyect. In the MQTT method the item is known as Topic.
- **Type:**
    - **Production:** to indicate that the signal is a production counter of total results generated, including good and bad (scrap and rework) results. Records of counters need, to consolidate results, at least another identical record of totals.
    - **Scrap:** to indicate that the signal is a production counter of scrap results generated. Records of counters need, to consolidate results, at least another identical record of totals.
    - **Rework:** to indicate that the signal is a production counter of reworked results generated. Records of counters need, to consolidate results, at least another identical record of totals.
    - **Consumption:** to indicate that the signal is a consumption counter.
    - **Generic:** to indicate that the signal is generic: does not have a specified function in edinn.
    - **Avail. Prod.:** to indicate that the process has inputs to process available at its entrance.
    - **Saturation:** to indicate that the process has outputs saturating its exit.
    - **Prod. Pulse:** a pulse signal to indicate that a single Production result was generated. **Records of pulses need, to consolidate results, at least another production counter of totals for the same topic and process.**
    - **Scra. Pulse:** a pulse signal to indicate that a single Scrap result was generated. Records of pulses need, to consolidate results, at least another identical record of totals.
    - **Rewo. Pulse:** a pulse signal to indicate that a single Rework result was generated. Records of pulses need, to consolidate results, at least another identical record of totals.

WARNING: it is not recommended to use Pulses signals instead of Counters signals to count results or consumptions since the system might not read these signals fast enough or possible failures in the conexion with the OPC Server could result in an information lost. Also please remember to adjust the Check counters every parameter.

- **On/Off:** to indicate that the process is working.
- **On/Off_T:** to indicate that the process is working. This will take micro stops into consideration.

WARNING: To use On/Off signals can lead to errors, instead. For example: when a process is not producing anything, but the operator of the process left it powered on. It is only recommended when the process needs a lot of time (hours or days) to produce a result, and we have another way (manually or electrically) to determine when the process is working, so the system will manage On/Off signals complementary. If not, we recommend the use of production counters.

- **Good_S:** counter that automatically determines the quantity of Scrap production by substracting the amount of good productions to the total amount of production. Will not calculate the quantity of Rework.
- **Good_R:** counter that automatically determines the quantity of Rework production by substracting the amount of good productions to the total amount of production. Will not calculate the quantity of Scrap.
- **Good_SR:** to indicate the final quantity of good results. This counter, in conjunction with the Production counter, will automatically determine the bad (rework or scrap) results.
- **RGY:** when a certain button (Sched. [Schedule], Resul. [Results], Statu. [Statuses], Consu. [Consumptions], Autoc. [Autocontrol], SPC [SPC] and OEE) of an associated terminal for a certain process is blinking in red (R), green (G) or yellow (Y) the system will write a 1 in this signal. This is useful to turn on beacons.
- **LifeBit:** the system will write a 1 here every minute if the data from this driver is being obtained, processed and stored correctly.This is used to light the Data On LED on *eBOXes*.
- **Justify:** this signal will automatically justify a stop since it was activated.
- **Justify_A:** this signal will automatically justify a stop since the beginning (justify all) of the stop.
- **ResChange:** when this signal becomes 1, it will change the result of the assigned process to the result code indicated in the field **Indic**.
- **Subscription:** look for the subscriptions of the process and if there are any, this signal will be written to 1.
- **Status Active:** this signal will be written to 1 when the current status of the assigned process is the same indicated, by its code, in the field **Indic**.
- **Measure_Res:** it will store the values of this signal inside the assigned process, considering the active result for the selected program and measure. This signal will enable the **Prog**. and **Measure** fields.

  - **Result_Id**: this signal will change the result of the assigned process. The id of the new result will be taken from the value of the signal.
  - **Measure**: it will store the values of this signal inside the assigned process, without considering the active result but for the selected program and measure. This signal will enable the **Prog.** and **Measure** fields.

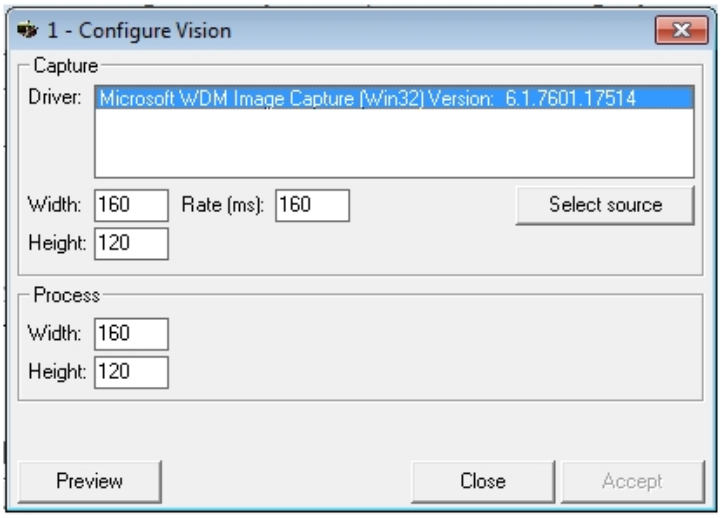
- **Counter / Signal / UDL**: to indicate the behavior of the system with the signal: used as a counter, a signal or UDL. If UDL is selected, the UDL will be executed every time the system needs to read the signal.
- **Thre (%):** to indicate in percentage the threshold in the process production speed that has to be surpassed in order for the system to determine that the process is working. For example: if a process has been configured to produce 1 result per minute, if we type 50% here, we are configuring the system to consider the process as working when the process is producing 0,5 results / minute, that is a cycle time of 120 seconds.
- **(%):** to indicate the minimum percentage of the cycle time established that has to be surpassed in order for the system to determine that a product has been done. If not, the system will ignore it. This is useful to avoid spurious pulses.
- **Desc.**: free text to describe the signal.
- **Proc.**: process to which the signal is associated. On the right you can select whether the register will be total or by pieces.
- **Cons.:** type of consumption.
- **Prog.**: program that will be used to store the value of this signal.
- **Measure**: measure that will be used to store the value of this signal.
- **Active**: if not marked, the signal is totally ignored by the system.
- **DB**: if not marked, the signal is not stored into the database. Although generally not used, it is useful to monitor very fast signals.
- **Reset on:** mark this check if you want the system to write into the item to reset the counter when the value of the next field is achieved. Whether you mark this check or not, type in the next field the value to which the item automatically resets, or leave 0 if the item never resets automatically.

> **NOTE:** There is a special use for the option Reset on: when it is active and the value of the next field is 0. In this case, if the counter has a new value which is less than the previous, it will consider that is 0 instead of the new value. This special use is convenient when we have counters that do not behave correctly because they decrease instead of always increase, and we prefer to ignore those counts which are less instead of considering that the new lesser count was entirely produced. The downpart of this use is that if the lesser count is because the counter resetted and produced until the new count, that count will be discarded and considered 0. To avoid all these problems and having to use this special use, please remember that correct counters for edinn are those which never reset or reset at fixed counts.

- **To UDL:** if the signal is sent to the UDL for further processing.
- Without label: dropdown list field to relate the item (normally a Generic Signal) with an Event. If this relation is set up, and in addition the item is also related to a process (field Proc. above in this document), and that process is configured as to Trace to devices, then the value of the signal will be visible from the Registry window of a win32 terminal.
- **Send desc.**: if marked, every time a result is generated, the description of the associated counter will be sent as a comment on the result record. This is useful when we have more than one signal for scrap or rework, and we want the users to be able to distinguish the source of the bad results.
- **Indic.:** useful to complete the Type option:
    - **Justify type:** to indicate the status id to justify.
    - **Scrap / Rework:** to indicate the product id to define as Scrap or Rework.
    - **Reschange:** to indicate the product id to change.
    - **Active:** to indicate the status id to active the signal **Status Active.**
- **Deadb (%):** allows to ignore small changes in this signal. For example, a deadband of 10% will ignore changes in this signal which are less or equal to 10% of the previous value. This means it will process the signal only when Abs(NewData - OldData) / OldData > Deadband, where Abs means the absolute value.
- **Delay (secs):** define the time it takes for products to pass from one process to another, in a continuous workflow. This is useful when it comes to justifying the status of a process that has other dependent processes behind it. For example: if process 1 breaks down at 10:00, and the process 2 has a delay of 5 minutes, the processs 2 will be automatically justified with a "dependent" status at 10:05, instead of at 10:00.
- **User:** the OPC UA user.
- **Password:** the OPC UA password.
- **Refresh button**: refreshes the table below.
- **Configure button**: only available to configure cameras. Please see: Configure Vision.
- **Insert button**: inserts the Monitor configuration.
- **Modify button**: modifies the Monitor configuration.
- **Delete button**: deletes the Monitor configuration.
- **Table:** shows the parameters already configured.

Configure Vision

Only available to configure cameras. The system will consider this signal as 1 when a certain pattern on the image is detected.



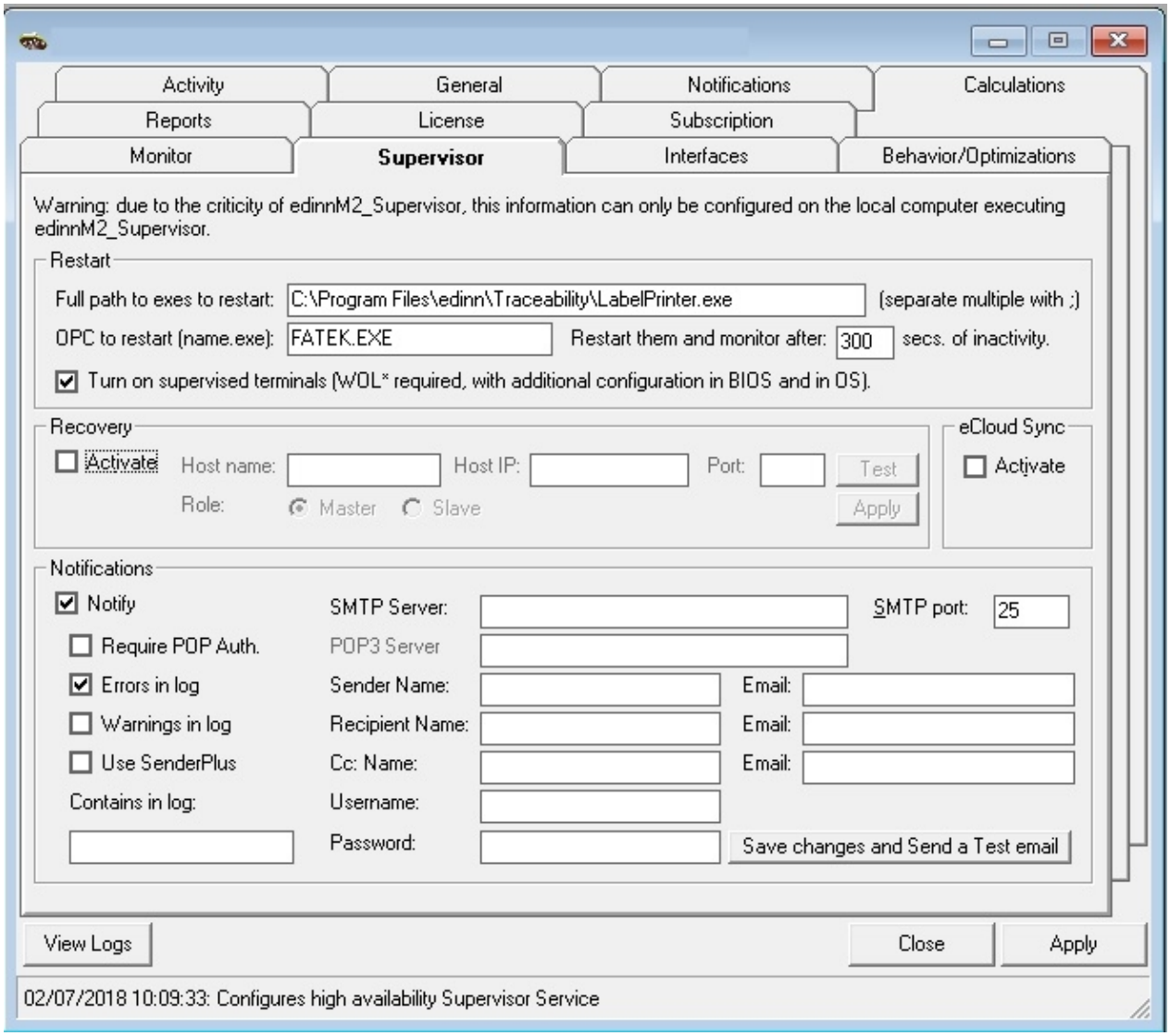
Functionality is as follows:

- **CAPTURE**:
    - **Driver:** choose the desired drivers from the installed on the computer.
    - **Width and height:** of the captured image. The bigger, the less rate of capturing images.
    - **Rate (milliseconds):** milliseconds to wait before capturing a new snapshot.
- **PROCESS**: with what size images will be processed. The bigger, the slower of the detection.
- **PREVIEW**: allows to view in real time what the camera is viewing and to see when the system detects a pulse.
    - **Process:** allows configuration on how to process images in order to detect pulses. You can select as much areas as you need. Every area has a threshold. Whenever all the areas are above or below (configurable) their threshold, and considering the AND / OR conditions between them, the signal will be put as 1 or 0. A 'Pulse' indicator is also present in order to help you to determine if the system is well configured to detect the pulses when you need them.

Supervisor

Once in the console, you will see the Supervisor tab. The functionality of the this subsystem is to supervise that the other Server subsystems are correctly working and respond to

inputs.



Functionality is as follows:

- **RESTART**: configuration to restart processes.
    - **Full path to exes to restart:** indicate here the path of an executable to be executed **always** (even if it is closed, it will be launched again). You can add new paths separating with commas.
    - **OPC Server to restart:** indicate here the name of the OPC Server exe file as seen in the windows task manager, that will be restarted whenever monitoring is timed out.
    - **Restart processes after (secs):** restart any process (task, port, etc., as seen in the Activity tab) that is more than this number of seconds without activity. The last activity of a process is marked in the Alive field of the Activity tab.
    - **Turn on supervised terminals (WOL * required):** if marked, the system will send a WOL (Wake On LAN) TCP/IP packet to the Network Interface Card MAC address

as indicated in the Devices configuration of the edinn® M2 Terminal, and only to those devices that are marked to "Supervise" on the Devices configuration. This functionality requires:

1. To activate WOL in the BIOS of the PC that will be switched on.
2. To activate the "Allow this device to wake up the system" in the operative system (Windows) in System, Device Manager.

- **RECOVERY:** this is the service that copies the contents of the database of one server in the database of a backup server, in order to avoid the loss of data if the main server falls.
    - **Activate:** activates this recovery service.
    - **Host name:** host name of the backup server.
    - **Host IP:** host IP address of the backup server.
    - **Port:** port of the backup server.
    - **Test connection:** to check if the connection with the servers is stable.
    - **Role:** to define the role of the server. If the master server falls, the system will automatically change the role of the slave server to be the master. When the first server is recovered, it has to be changed manually to the master role again. For the slave role, the services "central", "monitor" and "reports" in the Activity tab are disabled.
    - **Apply Recovery:** to apply this configuration.
- **eCLOUD SYNC:**
    - **Activate:** if marked, keeps this system synchronized with the edinn cloud (eCloud) service. This is useful for organizations that want to have their monitoring or server locally on each site, but they want to consolidate the information onto a bigger system for multi site, or multi company supervision.
- **NOTIFICATIONS:**
    - **Notify:** the supervisor will generate errors, warnings and control messages in log:
        - **Require POP Auth.:** if a Server POP autorization is required (POP3 Server: is an application-layer Internet standard protocol used by local e-mail clients to retrieve e-mal from a remote server over a TCP/IP connection. For more information, please see: https://en.wikipedia.org/wiki/Post_Office_Protocol)
        - **Errors in log:** if marked, the supervisor will send emails notifying errors in log.
        - **Warnings in log:** if marked, the supervisor will send emails notifying

warnings in log.

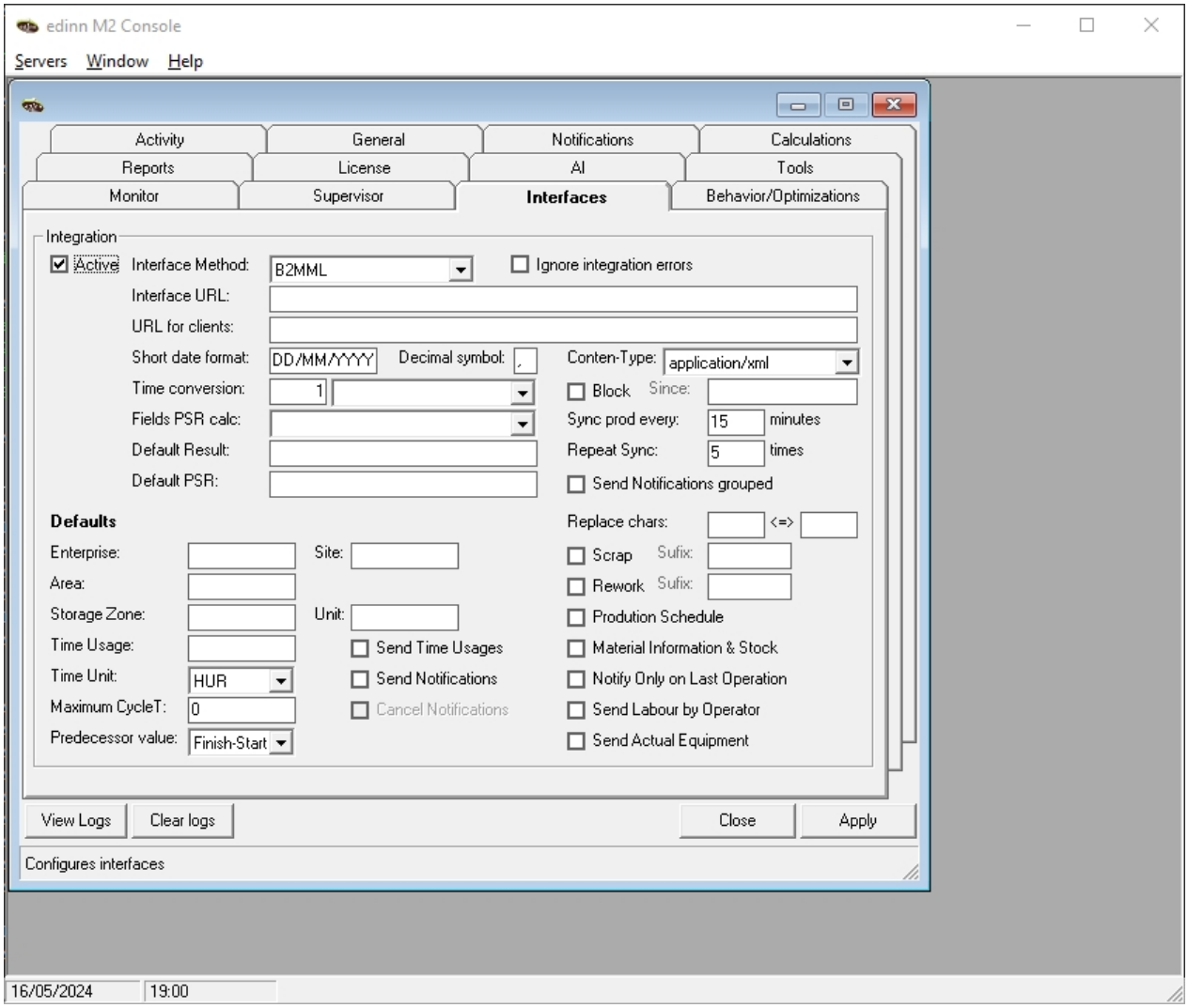
- **Use SenderPlus:** to send emails with SSL (Secure Sockets Layer), cryptographic protocols that provide secure communications over a network.

- **Contains in log:** if you also want to be notified by the supervisor of the logs that contain a certain word.
- **Configuration of the emails:** general parameters of the notifications.

NOTE: although you have not selected any of the 'Errors in log', or 'Warnings in log', you will still be notified if any of the subsystems is not working and the supervisor could not restore it.

Interfaces

Once in the console, you will see the Interfaces tab, which has the configuration of the integration interface with the ERP or another system.

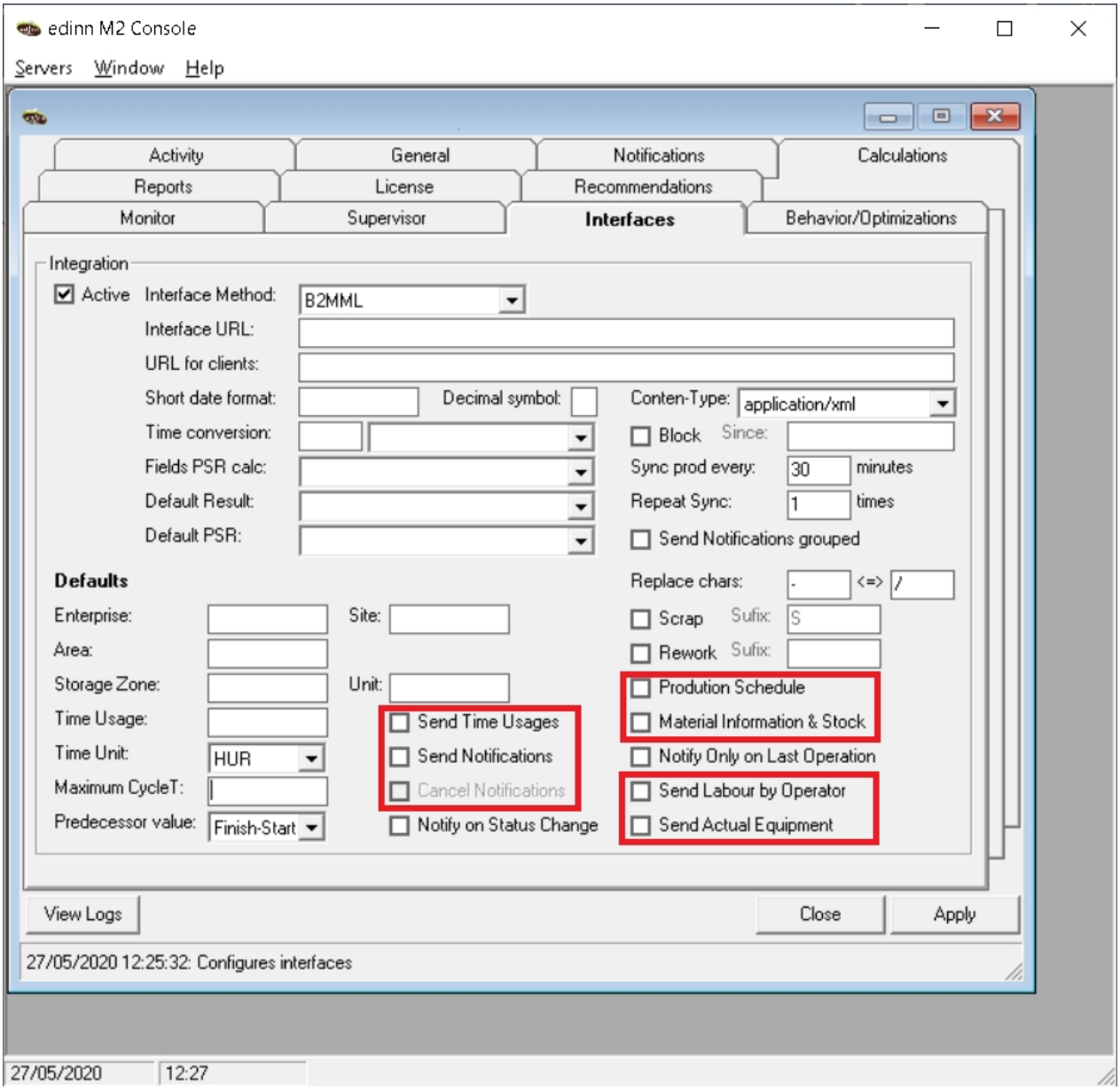


Functionality is as follows:

- **Active Integration:** to active the configuration of the integration with a ERP.
- **Interface method:** there are 2 integration methods, SOAP and B2MML (recommended). There are interchange standard languages between business applications (ERP) and manufacturing applications (MES)
- **Ignore integration errors:** Ignores errors returned by the ERP when results are added manually. In this way, results can be inserted even if they do not enter the ERP.
- **Interface URL:** the URL of the system to send the subscription messages from the edinn server.
- **URL for clients:** the URL to which the terminals are going to connect to send the messages to the server.
- **Short date format:** normally day/month/year.
- **Decimal symbol:** to adapt the messages to the ERP configuration, normally ",".
- **Time conversion:** to adapt the messages to the ERP configuration, edinn works with seconds.
- **Fields PSR calc:** calculation type to convert the productions units.
- **Default Result & Default PSR:** to make a custom configuration of the fields that define the products and the manufacturing orders.
- **Enterprise:** name of the company.
- **Site:** location of the company.
- **Area:** internal parameter that identifies a distribution within the company.
- **Storage Zone:** internal parameter that identifies a distribution within the company. Default value 1.
- **Time Usage:** automatic state of the edinn catalog that will start with the start of a new manufacturing order. If left blank, the previous status will continue, if a "-" is set, whith the start of a new manufacturing order, the system will request the status.
- **Time Unit:** standard measure units.
- **Maximum Cycle T:** determines the cycle time based on the duration of the production order and the desired number of results within this. This is useful for monitoring the machine.
- **Predecessor value:** how the operations within a production are related. There are 5 relationships available:
    - **Finish-Finish:** activity A must finish before activity B can finish.

    - **Finish-Start:** activity A must finish before activity B can begin.
    - **Start-Finish:** activity A must start before activity B finishes.
    - **Start-Start:** activity A must start before activity B can start.
    - **None.**

- **Conten-Type:** to send the XML message as a parameter in a application to the ERP or inside the body of the petition. This is accorded with the other part.
- **Block:** if marked, any message will be sent since a certain date. The messages will remain in edinn until until the blockage is removed.
- **Replace chars:** if the ERP uses characters in its code that conflict with those used in the edinn code, it can be configured to change the character in incoming messages.
- **Scrap & Rework:** every time a production order is sent, a suffix will be associated to the product name that will indicate if it is scrap or rework, for example: if the product is 01, a suffix .S (generally, indicating Scrap) or .R (generally, indicating Rework) will be added.
- **Subscriptions**:



- **Send Time Usages:** Sends a time usage notification to the subscriber and/or cancelation every time a time usage is inserted, modified or removed, and, if active, on order status changes and shift changes. More information here.

- **Send Notifiations:** Sends a quantity notification to the subscriber (both production and consumption) each time a registry is inserted or modified*, and, if active, on order status changes and shift changes. More information here.

Note: Cancel Notificactions must be active in this case as a modification triggers a cancelation and a notification message.

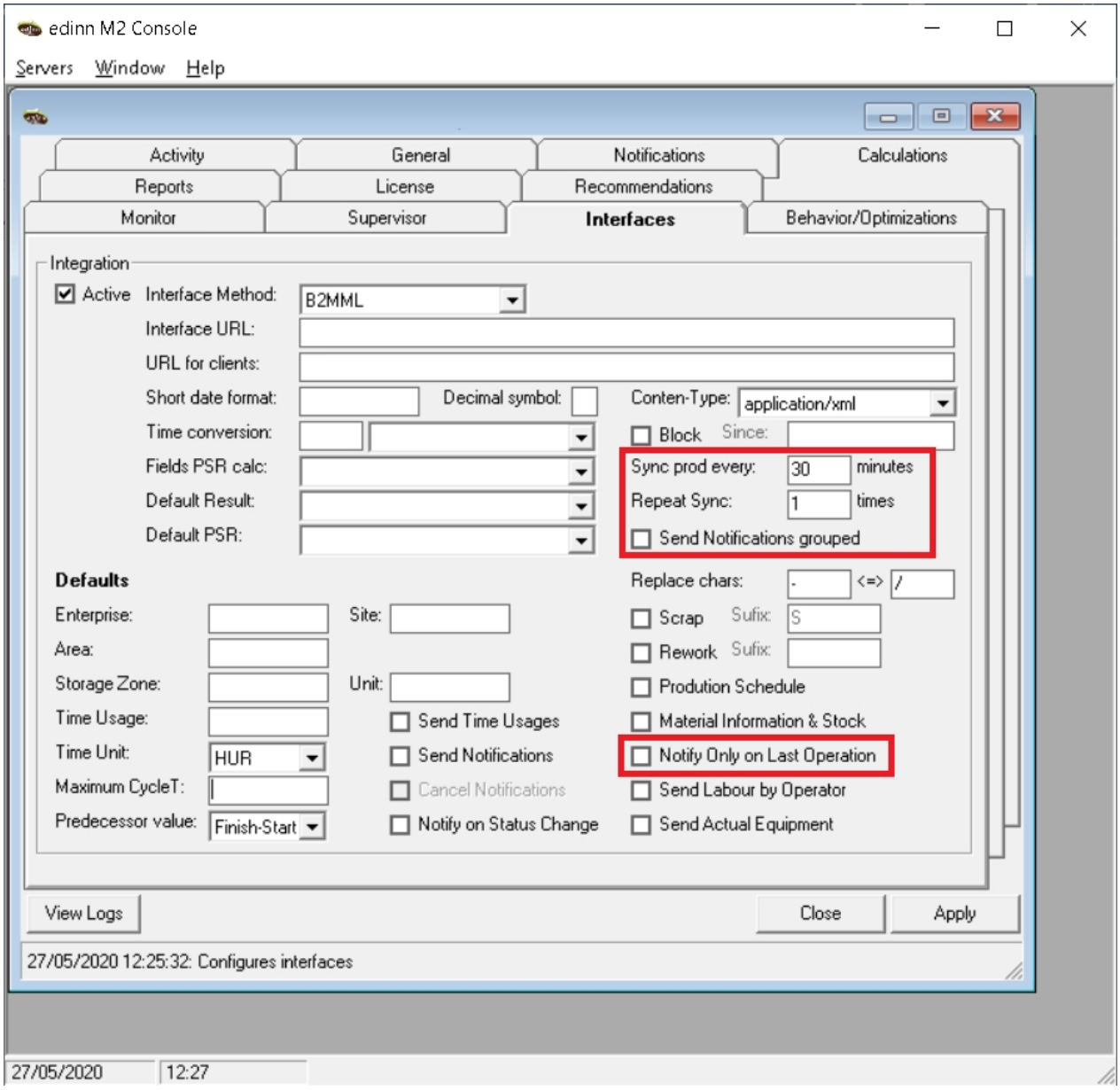
- **Cancel Notifications:** Sends a quantity cancelation to the subscriber (both production and consumption) each time a registry is modified or removed. More information here.

- **Production Schedule:** Send a message to the subscriber each time the status of

an order is changed reporting it. More information here.

- **Material information & Stock:** The system will ask the subscriber how much stock is available for a product. More information here.
- Parameters of the notifications:
  - **Send Labour by resource (person)**: if marked, the labor time will be sent to the subscriber for each resource (person) indepently, if not, the labor time will be sent grouped.
  - **Send Equipment Actual**: if marked, the process that perform the operation will be sent to the subscriber.

- **Frequency** with which subscription notifications are sent.



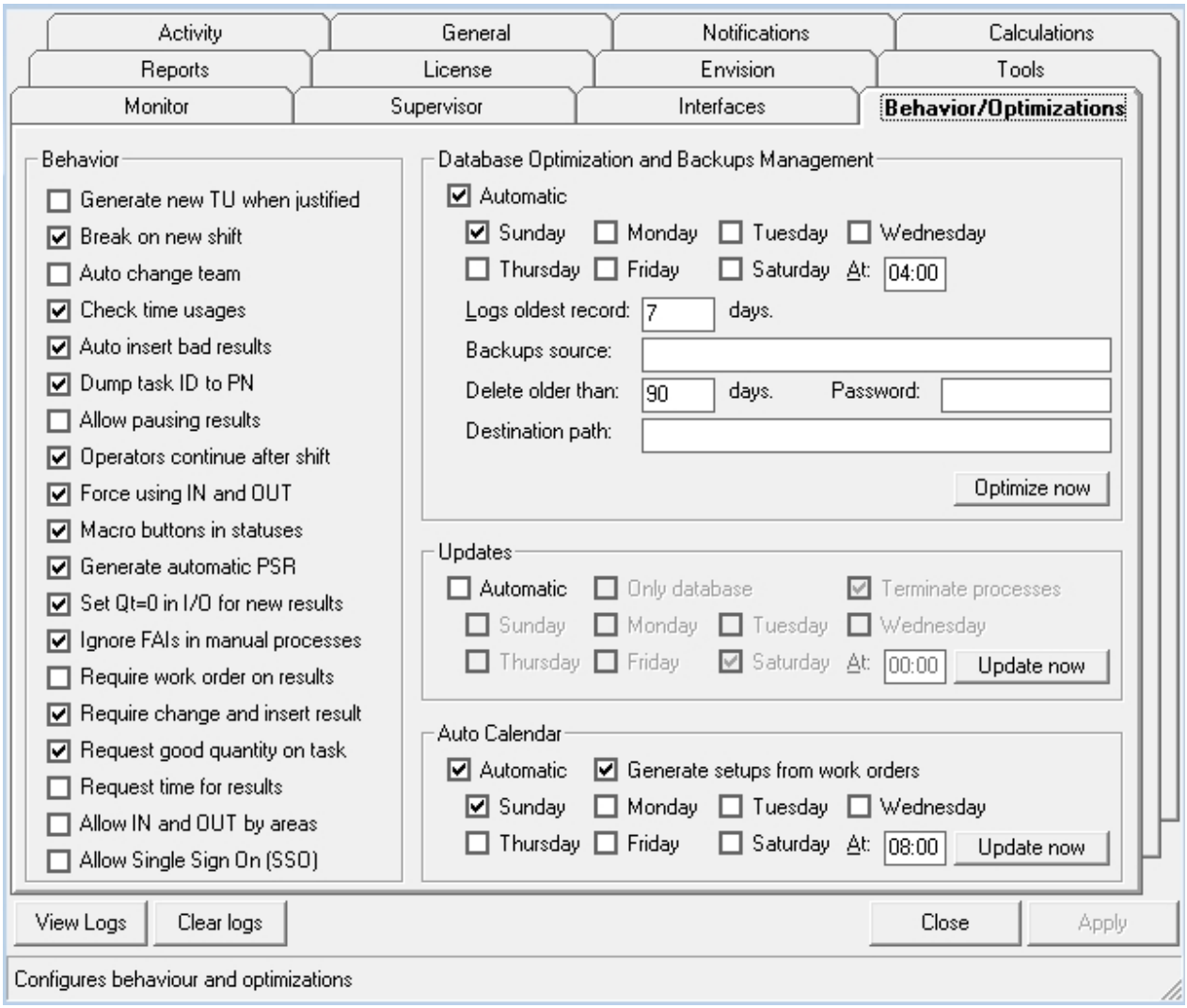
- **Sync prod every (minutes):** if the messages can not be sent to the subscriber

due to an error, this field will configure the period that will pass until it is fordwared.

- **Repeat Sync (times):** if the messages can not be sent to the subscriber due to an error, this field will configure the times that will be fordwared. Once the time has passed, the message will be canceled.
- **Send notifications grouped:** edinn can send messages to the subscriber at the time they are produced (online), at the end of each shift or order, or upon request or from a date (grouped). If not marked, every time a result is generated it will be sent to the subscriber, otherwise, they will be sent grouped.
- **Notify only on last operation:** if not marked, all the information of each operation will be sent to the subscriber, if not, will be sent only the information of the final operation.

Behavior / Optimizations

Once in the console, you will see the Behaviour / Optimizations tab:



Functionality is as follows:

- **BEHAVIOR**: configures certain behavior parameters.
    - **Generate new TU when justified:** if marked, when a status (time usage) is justified (for example, with a cleaning status) and the duration has been determined, once this time has expired, the system will close the cleaning status and create a new state of 'FAI'
    - **Break on new shift:** if marked, when a new shift starts, in all processes, the current status will be closed at the time of the end of the shift and a new identical status will be created starting when the new shift starts. This is useful to not have statuses which start and end in different shifts.
    - **Auto change team:** if marked, the system auto rotates the teams. Please see the General tab for more configuation options.
    - **Check time usages:** if marked, every status (time usage) inserted into the system is validated and checked to see if it is a micro stop or not.
    - **Auto insert bad results:** if marked, when the user changes the result of a

process, also the associated bad results are inserted. This is useful when bad results are automatically monitored, in order for them to be assigned to the correct reference.

- **Dump Task ID to PN:** if marked, when starting a new order, the field Order ID of the order will be copied to the field order (also know as Production Number, PN) of the result.
- **Allow pausing results:** if marked, every time the user indicates that he is going to change the result, the system will ask him if he has finished the previous result. If the users answers "no", the last result will be considered as paused. This option has an impact on the performance of the reports and the monitor, as the system has to search for paused results whenever it is analyzing performance.
- **Operators continue after shift:** by default, all resources (persons) are taken OUT (removed as resources) of processes when the shift ends. If you mark this options, resources (persons) will not be taken OUT of processes, and they will have to mark that they are OUT of the process by themselves.
- **Force using IN and OUT:** if marked, the user must indicate that he is IN or OUT of a process, before being able to change anything from it. This is useful to track activity of resources (persons).
- **Macro buttons in TU (time usages):** if marked, the default status window will be shown, if not, and advanced window will appear.
- **Generate automatic PSR (process-status-result):** if marked, whenever the user assigns a work order to another process, the system will automatically generate the PSR relations for that process, status and result. Therefore, this options will also make visible all results to the user from the terminal and not only those related in PSR to processes which the user can see.
- **Set Qt=0 in I/O for new results:** When new quantities of type production of the current result are produced, the associated inputs and outputs are created with quantity 0. This allows users to manually report inputs and outputs instead of set them edinn automatically.
- **Ignore FAIs in manual processes**: if marked, it will allow to manually put in production a process and to change the status of its orders, even having not justified failures (FAI). Only applies to manual processes.
- **Require Order on Results:** when the production of a new product starts, the system will ask the Production Number or order.
- **Require Change and Insert Result:** if marked, will require the user to generate a new result and then indicate the results it has produced.
- **Request good quantity on task:** when an order is finished or a shift change is made, the system will ask for the amount of good production.
- **Request time for results:** if a scrap or rework production is introduced, the

system will asks the time at which they were produced.

- **Allow IN and OUT by Areas:** instead of doing IN/OUT in each of the processes, allows to enter an entire area.
- **Allow Single Sign ON:** allows to associate in the terminal the Windows login with the edinn login. You can change this configuration in the resources configuration.

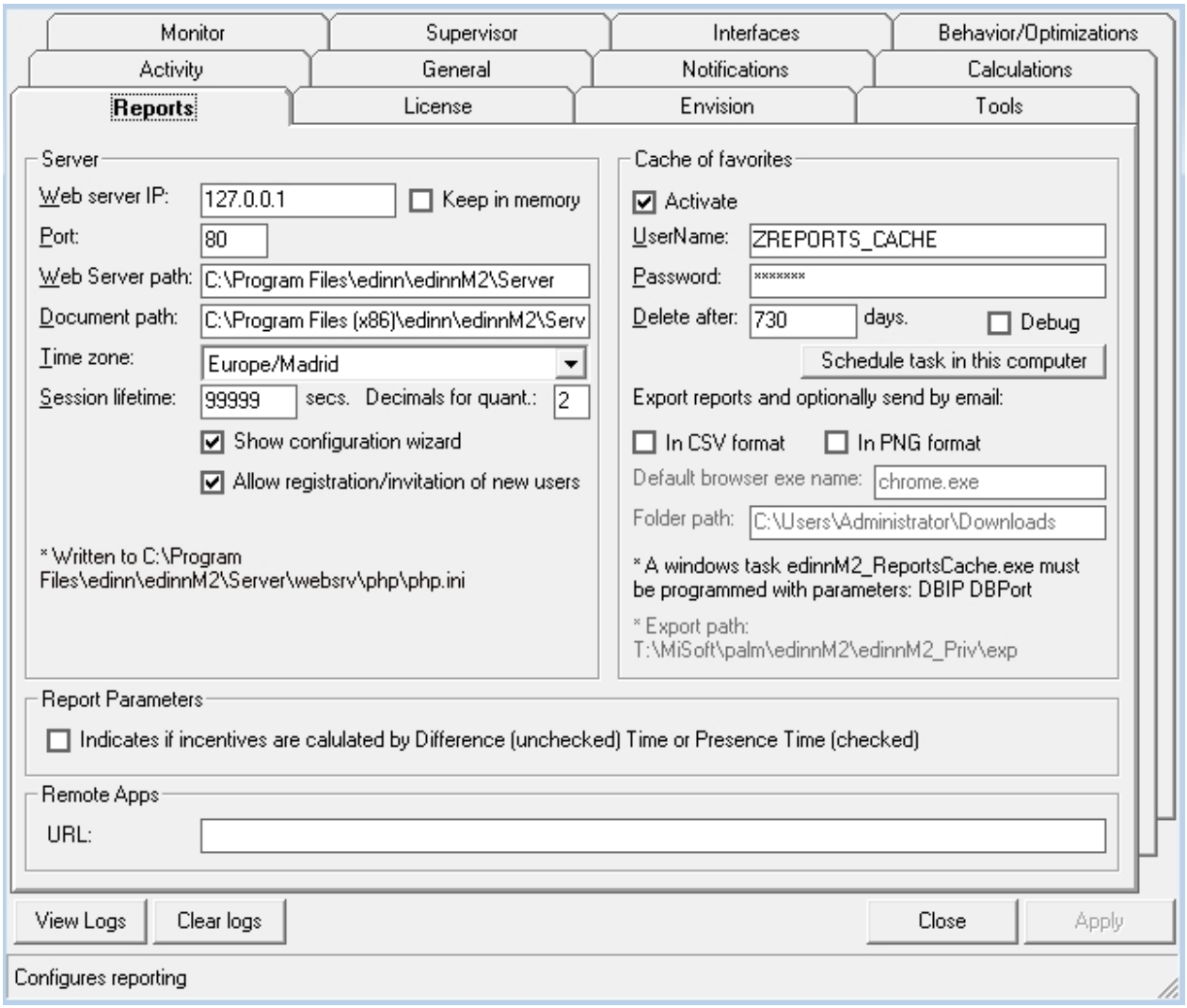
- **DATABASE OPTIMIZATION AND BACKUPS MANAGEMENT**:
  - **Automatic:** if marked the database and other elements, including the backups, will be optimized periodically. This options is very recommended in order to keep the system fast with the pass of the time.
  - **Log oldest record (days):** this field indicates how many days of information will be stored on the server logs. This is critical in order to keep the database at a certain size and therefore response time for users.
  - **Backups source:** complete path to the folder where the data backup files will be automatically generated by the system before optimization. If this field is empty, no backups will be generated. The system will also search for files which names finish with **YYYYMMDDhhmmss.XXX** or **YYYYMMDDhhmm.XXX**, where **YYYY** is the year, **MM** the month, **DD** the day, **hh** the hours, **mm** the minutes and, optionally, **ss** the seconds when the file was generated;and **XXX** is the extension of the file. With the files of this folder matching this format of names, will do the following:
    - **Delete older than**: will delete those files, both from the "source" and the "destination" folders, which are older than the number of days indicated in this field.
    - **Password**: will compress the files with the password indicated in this field.
    - **Destination path**: will copy the compressed files to the folder indicated in this field. This provides higher guarantees to avoid data loss when this folder belongs to a file system which is external to the server (similar, for example, to OneDrive, Google Drive or Dropbox).
  - **Optimize now:** click for an immediate optimization.
- **UPDATES:**
  - **Automatic:** updates the system to the latest available version automaticaly when defined (day/hour).
  - **Only database:** if you want only to update the database.
  - **Terminate processes:** forces the closure of all the services in the Activity tab for updating the system.
  - **Update now:** an update of the system will be performed immediately, if there

are new versions available.

- **AUTO CALENDAR**: automatic update of the programmed statuses system calendar. It is internally used by autocontrol, costs and planning modules. The system will automatically update the calendar, from the current moment (or the date and time indicated manually if updated from the console) to the number of days indicated in *Real time oldest record* indicated in General, every time there is a change in the statuses configuration or, if necessary, in the scheduled work orders. In addition, you can force its update with the button present in this section or periodically according to the schedule indicated in this section.
    - **Generate setups from work orders**: generates in the calendar the statuses of setups based upon the scheduled work orders. Please take into account that you should only have one status for the Setup class.

Reports

Once in the console, you will see the Reports tab: here you can configure the general parameters of the web server and the cache of favorites of the reports.



Functionality is as follows:

- **SERVER**: here you can configure the IP and other parameters of the server that serves the web server.
    - **Web server IP**: IP address of the web server.
    - **Keep in memory**: the reports component will be kept in memory making the reports slightly faster. If you activate this option, you will have to restart the server before updating the version of the edinn platform.
    - **Por**: IP port of the web server.
    - **Web server path**: local path to the web server application.
    - **Document path**: local path to the documents of the web server.
    - **Time zone**: time zone of this system.
    - **Session lifetime (secs):** seconds for the expiration of the inactive web reports user sessions. Type here the highest possible value (999999999) for no expiration.

- **Decimals for quant.:** how many decimals have the quantities of the reports.
- **Show configuration wizard**: the web configuration wizard access will be shown, for administrators, as an icon of a wrench at the top right of every report.
- **Allow registration/invitation of new users**: persons can request registration and become new users from the login page of the reports. Also allows administrators to invite, from the terminals, to other persons to become users.

WARNING: This wizard is designed to configure only new systems, and therefore Administrator users could **delete** detailed configuration that your system already has.

- **CACHE OF FAVORITES**:
  - **Activate**: activates the caching system of favorite reports.
  - **UserName**: id of the user that will be used to launch the favorite reports to cache them. If you remove this user or change his password from the application configuration the cache of favorites will stop working.
  - **Password**: password of the user that will be used to launch the favorite reports to cache them. If you remove this user or change his password from the application configuration the cache of favorites will stop working.
  - **Delete after (days)**: cache of reports will be automatically purged, when it gets older than this number of days.
  - **Debug**: the cache of favorites reports leaves a debug trace in the server root folder.
  - **Export reports and optionally send by email**: allows users to set, by clicking an email icon, their favorites reports to be exported to CSV or PNG and be sent to the user email. Only favorite reports of closed periods, not current, can be exported and sent by email; otherwise, the users would receive an email every minute.

Note: To export to PNG it is mandatory additional configuration on the server. If your company is allocated in ecloud.edinn.com, please contact support and request this change.

  - **Folder path:** indicates the folder where the exported files, both CSV and PNG will be saved.
  - **Default browser exe name:** indicates the browser that will execute the export to PNG image.
  - **Schedule task in this computer:** creates the task in the Windows Task Scheduler which mantains the favorites' cache. This is useful to keep the favorite reports always updated.
- **REPORT PARAMETERS:**

- **Indicates if incentives are calculated by difference (unchecked) Time or Presence Time (checked):** by difference, an incentive will be granted to the resource (person) if it exceeds a certain percentage of activity, if it is done by presence time, the incentive will be given depending on the time of presence.

- **REMOTE APPS:**

  - **URL:** indicates the URL to open remote apps, typically the terminal and console applications, that are actually executed in the server. If you indicate a URL here, a new menu option will be shown in the common options menu of the reports. If you inform this field, the server will automatically create a user in the Windows server to access Remote Apps (console and terminal) and maintain the password synchronized for the (new and modified) users with the Administrator role.

    The user login, to open the Remote Apps and to access the server console, is 1234567890_USER, where:

    - 1234567890 is the company identification number.
    - USER is the code of the resource (must be in upper case).

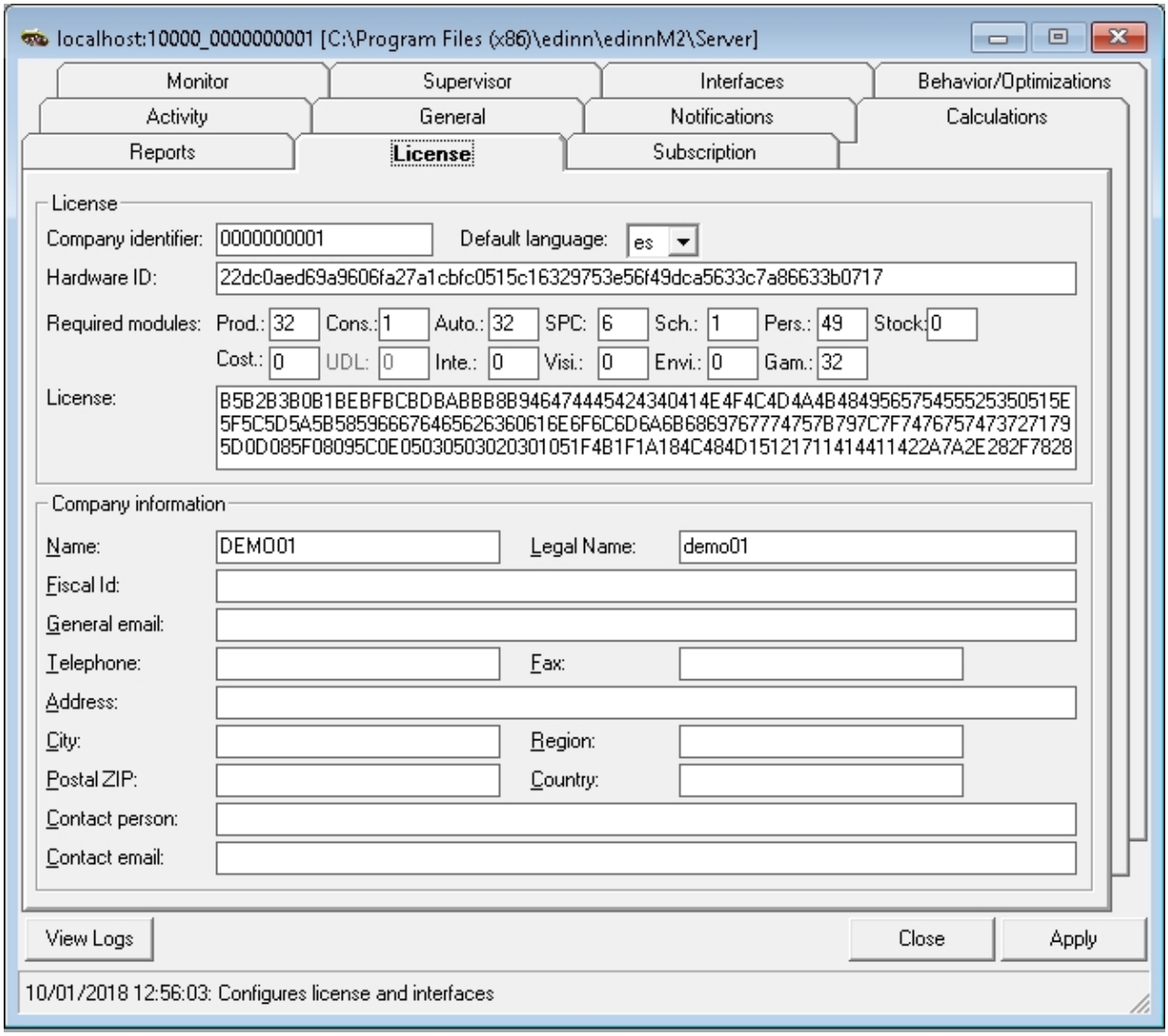
    Executing remote apps can be necessary when:

    1. There is no other way of accessing the Console.
    2. The terminal or the console work slowly.
    3. The terminal or the console do not work in a device (tablet, mobile phone, etc.) but Microsoft Windows Server remote applications do.

WARNING: Remote apps need additional configuration in your server and could imply also additional costs of licenses. It also implies certain restrictions in the resources configuration.

License

Once in the console, you will see the License tab: here you will find the company information and the section to introduce new licenses.



Functionality is as follows:

- **LICENSE:**
    - **Company identifier:** this number is automatically generated when installing the server. Can not be changed.
    - **Hardware id:** this code is automatically generated by the system, **after trying to start the Central service**, according to the present hardware.
    - **Required modules:** number of processes that use each module. These numbers are calculated automatically.
    - **License:** this code is provided by edinn and it will be **updated automatically** if you have an **active maintenance contract** with edinn, or any of its authorized partners, and if your **server has a permanent access to https://ecloud.edinn.com/**

      If your license is about to expire, the system will previously write error messages in the logs with the word "license", and therefore it is recommended to configure

the Supervisor tab so that the system will send you emails if errors, or this particular error, occur.

In case you are updating this field manually, follow these steps:

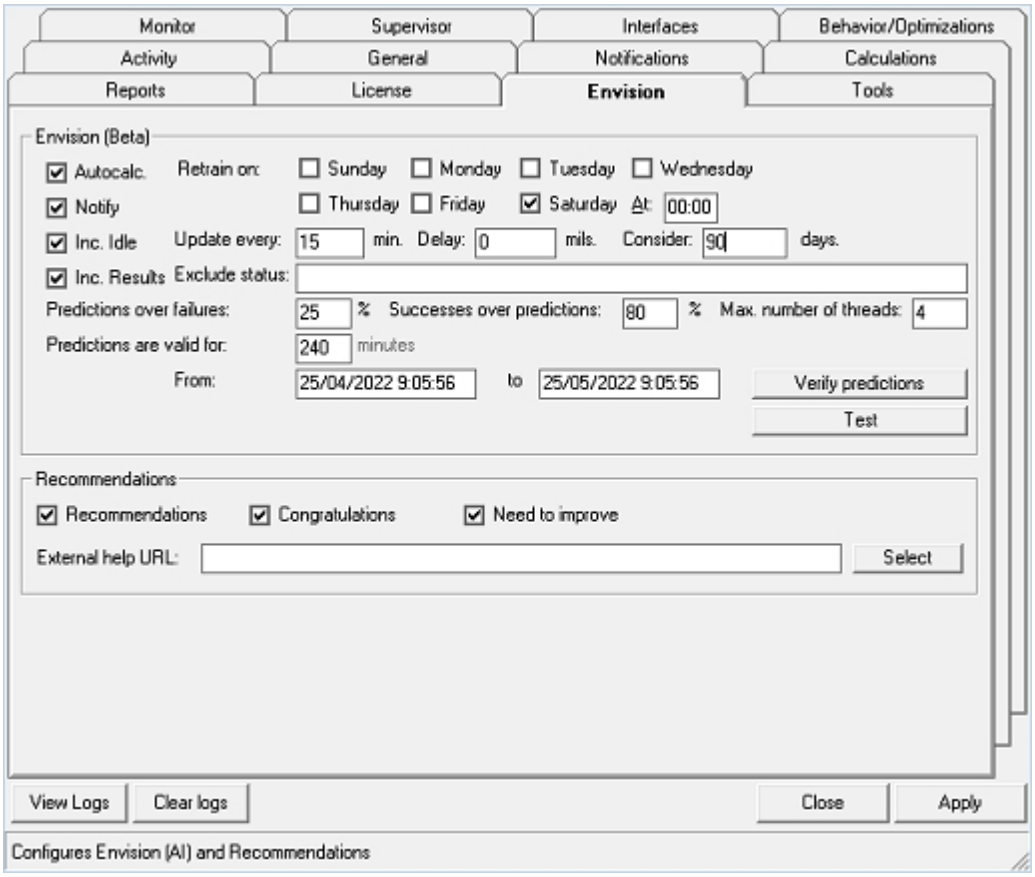
1. **Before updating the license code, backup the current license code** by copying and pasting it to a file. In case the new license code fails, you will be able to restore the current one and therefore the system will not stop while you manage to obtain a valid license.
2. Copy the new license code received from edinn or an authorized edinn partner and paste it into this field.
3. Click on the Apply button (bottom right) to save the changes.
4. From the Activity tab, restart the Central service and verify that it remains executing in green, that it does not stop automatically and that the tasks of the system appear executing in the Activy tab.
5. If you wish, click in View Logs (botton left) to see if there exists any error in the system.

- **COMPANY INFORMATION**: Type here the contact and legal information of the organization using the system. The field NAME is important as it will be the company name that users will have to type to access the system.

IA

> **NOTE:** The Artificial Intelligence (Envision) module is currently in Beta version and subject to changes.

Once in the console, you will see the IA tab, to configure Artificial Intelligence like failure prediction (also know as Envision) and Recommendations.



Functionality is as follows:

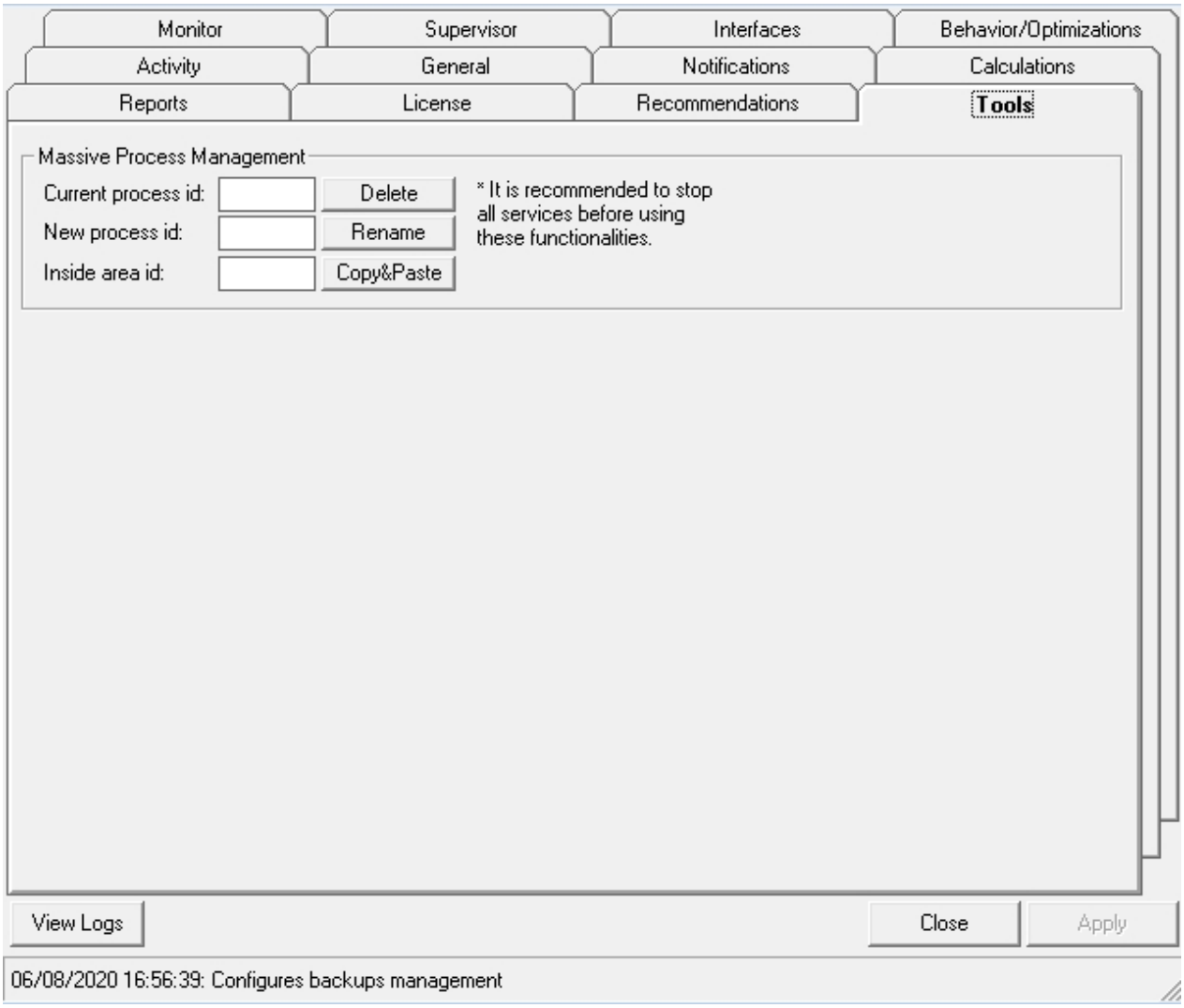
- **ENVISION**: allows to warn devices of a failure in processes **before** it occurs. Envision technology is based on statistical analysis and neural networs. Paramaters are:
    - **Autocalc:** if marked, automatic calculation will be performed.
    - **Retrain on**: the days of the week and the time when retraining of the neural networks will be performed automatically.
    - **Update every (min):** indicate the number of minutes to wait before starting to predict again for all processes.
    - **Delay (miliseconds):** field used to delay the calculations to reduce the server workload.
    - **Consider (days):** in order to search for patterns in failures, a number of past days must be taken into consideration to train neural networks. This parameter allows to configure how many past days should be considered.
    - **Notify:** if marked, devices will be notified. If not, only the calculation and the store into the database will be performed.
    - **Inc. Idle**: if marked, also idle statuses will be considered for the training.
    - **Inc. Results**: if marked, also results will be considered for the training.

- **Exclude status**: list of statuses to be excluded from the predictions. This list will be a list of IDs of failures separated by commas (,).
- **Predictions over failures**: target of predictions over the total number of failures. If the system does not reach this objective, it will not generate predictions.
- **Successes over predictions**: target of successful predictions over the total number of predictions. If the system does not reach this objective, it will not generate predictions.
- **Max. number of threads**: maximum number of execution threads which the system will launch.
- **Predictions are valid for**: minutes during which a prediction will be valid.
- **From / to:** indicate the period that you want to validate the predictions.
- **Verify:** verify the predictions performed, and not notified to users, in the indicated period.
- **Test**: allows launching now the prediction of failures, to test its results in the logs (selecting very high detail) to be able to find the best parameters.

- **RECOMMENDATIONS**:
  - **Recommendations**: activate if you want messages of recommendations to be sent automatically to the users.
  - **Congratulations**: activate if you want messages of congratulations to be sent automatically to the users.
  - **Need to improve**: activate if you want messages of need to improve to be sent automatically to the users.
  - **External help URL**: type the URL (should be accessible from the terminals) which will be showed to the users in the help (?) button.

Tools

Once in the console, you will see the Toos tab, which allows to use certain tools of the server.



- **MASSIVE PROCESS MANAGEMENT:** to delete, rename or modify the id of a process.

WARNING: It is recommended to perform a full backup of the database before using the following functions as deleted or overwritten data cannot be recovered.

  - **Delete**: permanently deletes all the data of the process which id is indicated in the field **current process id**.
  - **Rename**: renames the id of the process indicated in the field **current process id** to the new process id indicated into the field **new process id**.
  - **Copy&Paste:** copies all the data from the process which id is indicated in the field **current process id** into the **new process id**. You can also insert the new process into a specific area by indicating its id inside the field **inside area id**.

IoT (Monitoring)

IoT is the acronym of Internet of Things, also know in edinn as Monitoring, and is the ability

of the edinn system to obtain data directly and automatically from the processes.

The edinn system has a powerful architecture which totally isolates, from the "brain" of the system, the data acquisition. This allows different monitoring methods to be valid and they all generate exactly the same results in terms of information analysis and management. The idea behind is: you provide basic data from your processes, and the edinn system will provide you with all the valuable information.

Please see the different monitoring methods by edinn.

Methods

Monitoring is the ability of the edinn system to obtain data directly and automatically from the processes.

The edinn system has a powerful architecture which totally isolates, from the "brain" of the system, the data acquisition. This allows different monitoring methods to be valid and they all generate exactly the same results in terms of information analysis and management. The idea behind is: you provide basic data from your processes, and the edinn system will provide you with all the valuable information.

The edinn system allows different methods to monitor processes:

- **edinnBOX_1**: this monitoring method connects with edinn monitoring BOXes (eBOXes). Please find more information here.
- URL **(edinnHIP)**: by sending URLs to the server, we are sending and receiving data to and from the system.
- edinn OPC Bridge **(edinnHIP)**: installing an application named edinn OPC Bridge in the server of the organization connects to the OPC Server, the OPC Bridge will automatically send data from the OPC Server installed locally where the edinn OPC Bridge is intalled, to the edinn server.
- **OPC (DA and UA)**: the edinn system is an OPC client which can connect to virtually every OPC Server. This monitoring method is not explained in this manual.
- Manual: is when the user manually indicates results or other situations by using the edinn terminal or other manual methods.
- **Cameras (Vision)**: it allows to detect presence and pass of a product by using IP or USB connected cameras. This monitoring method is not explained in this manual.
- **MQTT:** allows to connect to brokers (or servers) using this protocol.
- **Other**: there are other monitoring methods (for example FaconSvr) and no limitation has been found so far in terms of monitoring.

Need help? 

URL

URL Monitoring is based on sending URL (*Uniform Resource Locator*) requests to the edinn server.

In the monitoring, we must select the source **edinnHIP** to use this method.

Possible URLs that can be sent are:

- **Production**: this URL is to indicate that the process has produced results. To avoid communication overhead, it is recommended (not mandatory) that this URL is sent periodically (for example every 5 seconds), and not everytime a result is produced, so in this case the caller must be able to maintain the counter.

  In case the caller maintains a counter and sends the absolute value, the syntax is as follows:

  http://ecloud.edinn.com/edinnM2/monitor?Id_Company=0123456789&PType=0&Id_Proc=A001&Val=1000&Upd=20130821100000&Pass=password&Response_type=html

  In case de caller just sends the increments, the syntax is as follows:

  http://ecloud.edinn.com/edinnM2/monitor?Id_Company=0123456789&PType=0&Id_Proc=A001&Inc=1&Upd=20130821100000&Pass=password&Response_type=html

  Where:

  - **Id_Company**: is the company identifier.
  - **PType**: fixed value 0.
  - **Id_Proc**: is the Id of the process as configured in the system.
  - **Val or Inc**: Val or Inc must be present, not both, but at least one.
    - **Val**: is the value of the counter of total results produced. This value must:
      - reset to 0 at 32.000. This is 31.999 -> 0. It must never reset under no other circumstance.
      - always increase. If it decreases, the system will interpret that a reset has occured and will assume that 32.000 results were

produced.

- not count empty results: this is the counter of the total results produced, so it must never count without a result being really generated by the process.

- **Inc**: is the value that indicates how many results where obtained. This value must:
  - not count empty results: it must never count without a result being really generated by the process.

- **Upd** (optional): is the date time when the result was generated in format YYYYMMDDhhmmss. This parameter is optional, if not provided, the system will assume the actual date and time.
- **Pass**: password as indicated in the monitoring server console or in the monitoring step in the configuration wizard.
- **Response_type** (optional): indicates in which format the response is needed. It can have the values xml and html. In case this parameter is not present, html will be asumed. In case it is html, and the response is ok, the explorer will close automatically.

- **Scrap**: this URL is to indicate that the process has produced scrap results. This URL should not be sent every time a new scrap result is generated, as this could overload the communication with the system. It is recommended that this URL is sent periodically (for example every 5 seconds) so the caller must be able to accumulate the counter.

  In case the caller maintains a counter and sends the absolute value, the syntax is as follows:

  http://ecloud.edinn.com/edinnM2/monitor?Id_Company=0123456789&PType=1&Id_Proc=A001&Val=1000&Upd=20130821100000&Pass=password&Response_type=html

  In case de caller just sends the increments, the syntax is as follows:

  http://ecloud.edinn.com/edinnM2/monitor?Id_Company=0123456789&PType=1&Id_Proc=A001&Inc=1&Upd=20130821100000&Pass=password&Response_type=html

  Where:

  - **Id_Company**: is the company identifier.
  - **PType**: fixed value 1.
  - **Id_Proc**: is the Id of the process as configured in the system.

- **Val or Inc**: Val or Inc must be present, not both, but at least one.
  - **Val**: is the value of the counter of total results produced. This value must:
    - reset to 0 at 32.000. This is 31.999 -> 0. It must never reset under no other circumstance.
    - always increase. If it decreases, the system will interpret that a reset has occured and will assume that 32.000 results were produced.
    - not count empty results: this is the counter of the total results produced, so it must never count without a result being really generated by the process.
  - **Inc**: is the value that indicates how many results where obtained. This value must:
    - not count empty results: it must never count without a result being really generated by the process.
- **Upd** (optional): is the date time when the result was generated in format YYYYMMDDhhmmss. This parameter is optional, if not provided, the system will assume the actual date and time.
- **Pass**: password as indicated in the monitoring server console or in the monitoring step in the configuration wizard.
- **Response_type** (optional): indicates in which format the response is needed. It can have the values xml and html. In case this parameter is not present, html will be asumed. In case it is html, and the response is ok, the explorer will close automatically.

- **Rework**: this URL is to indicate that the process has produced reworked results. This URL should not be sent every time a new reworked result is generated, as this could overload the communication with the system. It is recommended that this URL is sent periodically (for example every 5 seconds) so the caller must be able to accumulate the counter.

  In case the caller maintains a counter and sends the absolute value, the syntax is as follows:

  http://ecloud.edinn.com/edinnM2/monitor?Id_Company=0123456789&PType=2&Id_Proc=A001&Val=1000&Upd=20130821100000&Pass=password&Response_type=html

  In case de caller just sends the increments, the syntax is as follows:

  http://ecloud.edinn.com/edinnM2/monitor?Id_Company=0123456789&PType=2&Id_Proc=A001&Inc=1&Upd=20130821100000&Pass=pa

ssword&Response_type=html

Where:

- **Id_Company**: is the company identifier.
- **PType**: fixed value 2.
- **Id_Proc**: is the Id of the process as configured in the system.
- **Val or Inc**: Val or Inc must be present, not both, but at least one.
  - **Val**: is the value of the counter of total results produced. This value must:
    - reset to 0 at 32.000. This is 31.999 -> 0. It must never reset under no other circumstance.
    - always increase. If it decreases, the system will interpret that a reset has occured and will assume that 32.000 results were produced.
    - not count empty results: this is the counter of the total results produced, so it must never count without a result being really generated by the process.
  - **Inc**: is the value that indicates how many results where obtained. This value must:
    - not count empty results: so it must never count without a result being really generated by the process.
- **Upd** (optional): is the date time when the result was generated in format YYYYMMDDhhmmss. This parameter is optional, if not provided, the system will assume the actual date and time.
- **Pass**: password as indicated in the monitoring server console or in the monitoring step in the configuration wizard.
- **Response_type** (optional): indicates in which format the response is needed. It can have the values xml and html. In case this parameter is not present, html will be asumed. In case it is html, and the response is ok, the explorer will close automatically.

- **Consumption**: this URL is to indicate that the process has consumed. This URL should not be sent every time a new consumption is generated, as this could overload the communication with the system. It is recommended that this URL is sent periodically (for example every 5 seconds) so the caller must be able to maintain the counter.

  In case the caller maintains a counter and sends the absolute value, the syntax is as follows:

http://ecloud.edinn.com/edinnM2/monitor?Id_Company=0123456789&PType=3&Id_Proc=A001&Id_Coty=ELEC&Val=1000&Upd=20130821100000&Pass=password&Response_type=html

In case de caller just sends the increments, the syntax is as follows:

http://ecloud.edinn.com/edinnM2/monitor?Id_Company=0123456789&PType=3&Id_Proc=A001&Id_Coty=ELEC&Inc=1&Upd=20130821100000&Pass=password&Response_type=html

Where:

- **Id_Company**: is the company identifier.
- **PType**: fixed value 3.
- **Id_Proc**: is the Id of the process as configured in the system.
- **Id_Coty**: is the consumption type (electricity, water, gas, etc.) Id as configured in the system.
- **Val**: is the value of the counter of total consumption. This value must:
  - reset to 0 at 32.000. This is 31.999 -> 0. It must never reset under no other circumstance.
  - always increase. If it decreases, the system will interpret that a reset has occured and will assume that 32.000 units were consumed.
  - not count empty consumption: this is the counter of the total consumed, so it must never count without a consumption being really generated by the process.
- **Upd** (optional): is the date time when the consumption was generated in format YYYYMMDDhhmmss. This parameter is optional, if not provided, the system will assume the actual date and time.
- **Pass**: password as indicated in the monitoring server console or in the monitoring step in the configuration wizard.
- **Response_type** (optional): indicates in which format the response is needed. It can have the values xml and html. In case this parameter is not present, html will be asumed. In case it is html, and the response is ok, the explorer will close automatically.

- **Generic**: this URL is to indicate that a signal has changed. This URL must be sent every time the signal changes.

  The syntax for this URLs is:

  http://ecloud.edinn.com/edinnM2/monitor?

Id_Company=0123456789&PType=4&Id=10&Val=99&Upd=20130821100000&Pass=password&Response_type=html

Where:

- **Id_Company**: is the company identifier.
- **PType**: fixed value 4.
- **Id**: the id of the signal as configured in the system.
- **Val**: is the value of the signal.
- **Upd** (optional): is the date time when the signal changed in format YYYYMMDDhhmmss. This parameter is optional, if not provided, the system will assume the actual date and time.
- **Pass**: password as indicated in the monitoring server console or in the monitoring step in the configuration wizard.
- **Response_type** (optional): indicates in which format the response is needed. It can have the values xml and html. In case this parameter is not present, html will be asumed. In case it is html, and the response is ok, the explorer will close automatically.

- **Available product**: this URL is to indicate that there is available product at the entrance of a process. This URL must be sent every time the signal changes.

  The syntax for this URLs is:

  http://ecloud.edinn.com/edinnM2/monitor?Id_Company=0123456789&PType=5&Id_Proc=A001&Val=1&Upd=20130821100000&Pass=password&Response_type=html

  Where:

  - **Id_Company**: is the company identifier.
  - **PType**: fixed value 5.
  - **Id_Proc**: is the Id of the process as configured in the system.
  - **Val**: must be 0 when there is NO available product and 1 when there is available product.
  - **Upd** (optional): is the date time when the signal changed in format YYYYMMDDhhmmss. This parameter is optional, if not provided, the system will assume the actual date and time.
  - **Pass**: password as indicated in the monitoring server console or in the monitoring step in the configuration wizard.

  - **Response_type** (optional): indicates in which format the response is needed. It can have the values xml and html. In case this parameter is not present, html will be asumed. In case it is html, and the response is ok, the explorer will close automatically.

- **Saturation**: this URL is to indicate that there is saturation of products at the exit of a process. This URL must be sent every time the signal changes.

  The syntax for this URLs is:

  http://ecloud.edinn.com/edinnM2/monitor?Id_Company=0123456789&PType=6&Id_Proc=A001&Val=1&Upd=20130821100000&Pass=password&Response_type=html

  Where:

  - **Id_Company**: is the company identifier.
  - **PType**: fixed value 6.
  - **Id_Proc**: is the Id of the process as configured in the system.
  - **Val**: must be 0 when there is NO saturation and 1 when there is saturation.
  - **Upd** (optional): is the date time when the signal changed in format YYYYMMDDhhmmss. This parameter is optional, if not provided, the system will assume the actual date and time.
  - **Pass**: password as indicated in the monitoring server console or in the monitoring step in the configuration wizard.
  - **Response_type** (optional): indicates in which format the response is needed. It can have the values xml and html. In case this parameter is not present, html will be asumed. In case it is html, and the response is ok, the explorer will close automatically.

- **On/Off**: this URL is to indicate that the process is working or stopped. This URL must be sent every time the signal changes. It is important to understand that the system does not need this signal to know that a process is working, so this signal is optional and is useful for those processes of very long Cycle Time which need an additional On/Off signal for more precision.

  The syntax for this URLs is:

  http://ecloud.edinn.com/edinnM2/monitor?Id_Company=0123456789&PType=10&Id_Proc=A001&Val=1&Upd=20130821100000&Pass=password&Response_type=html

  Where:

- **Id_Company**: is the company identifier.
- **PType**: fixed value 10.
- **Id_Proc**: is the Id of the process as configured in the system.
- **Val**: must be 0 when the process is not working and 1 when there process is working.
- **Upd** (optional): is the date time when the signal changed in format YYYYMMDDhhmmss. This parameter is optional, if not provided, the system will assume the actual date and time.
- **Pass**: password as indicated in the monitoring server console or in the monitoring step in the configuration wizard.
- **Response_type** (optional): indicates in which format the response is needed. It can have the values xml and html. In case this parameter is not present, html will be asumed. In case it is html, and the response is ok, the explorer will close automatically.

edinn OPC Bridge

Monitoring with edinn OPC Bridge consists in installing an application named edinn OPC Bridge in the server which has the OPC Server installed, and then it will automatically send data to the edinn servers.

For more information about the OPC standard, please see this page: https://opcfoundation.org/

To use this monitoring method together with the web configuration wizard, the steps are as follow:

1. Create an organization in edinn and obtain its company identifier. The edinn OPC Bridge will require it from you later.
2. Configure your company using the web configuration wizard. When you will arrive to the monitoring step, you will be required to introduce a monitoring password. Remember it as the edinn OPC Bridge will requiere it from you later. To use the edinn OPC Bridge, select **edinnHIP** as monitoring method.
3. Next, in the computer which has the OPC Server software installed and that is capable of reading your data, also install the edinn OPC Bridge. To do so, download the software edinn OPC Bridge from here: http://edinn.com/en/downloads.html
4. To install the edinn OPC Bridge, uncompress the package and execute setup.exe.

Follow the installation steps.

5. Once the installation is completed, the edinn OPC Bridge will require from you to introduce the company identifier that you obtained in the previous point 1 and the monitoring password that you provided in the previous step 2.

   NOTE: You can change the monitoring password at any time from the monitoring step at the web configuration wizard.

6. Once the installation of the edinn OPC Bridge is finished, it will be already executing and automatically sending data to the edinn server. In the web configuration wizard, at the step of monitoring, now you will be able to select from the drop down HW-Ref, the name of the machine where you installed the edinn OPC Bridge. Next, you will be able to select from the DCOM drop down menu the list of the OPC Servers softwares available in the machine where you installed the edinn OPC Bridge.

7. Once you have selected the HW-Ref and the DCOM, then you can type in the item column the name of the item that you want to monitor. Once the name of the item has been introduced, save the table, and reload it until the state, value and last update, shows you the monitored data.

Need help? 

Manual

Manual monitoring is when the user manually indicates results or other situations by using the edinn terminal or other manual methods.

To use this method, in the configuration web wizard, step of monitoring, we must select as source **Manual** or **edinnHIP**.

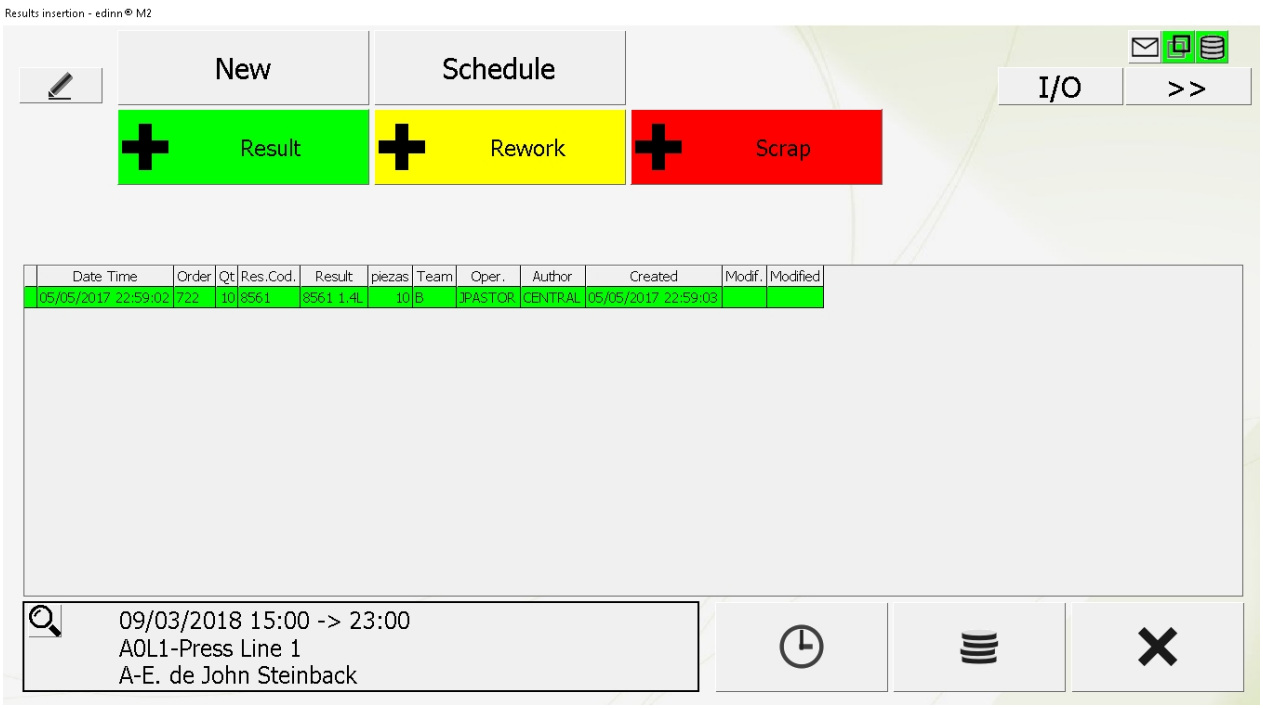
To use the terminal to indicate production

If we want to have buttons in the terminal to indicate production, then we have to:

- Choose **Manual** as the monitoring source, in the configuration web wizard, step of monitoring.

- Select, in the edinn terminal, in the process configuration the type of monitoring, to show the green (production), yellow (rework) or red (scrap) buttons, or any combination of them. This selection must be in accordance with the configured signals as Manual in the monitoring step of the Configuration Wizard.

After having performed the previous steps, you will have the operation buttons (green, yellow and red) as shown below in the results insertion window.



To use other systems to indicate production

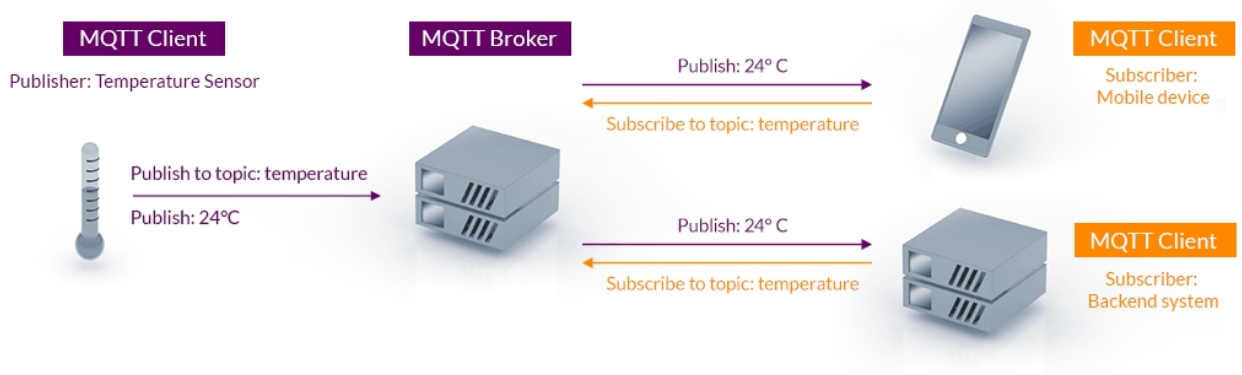
If we want to indicate production from any other system, then we have to send a URL request to the edinn server to indicate if we produced a good result, reworked result or scrap result.

You have information on how these URLs must be on the URL monitoring method.

You can also obtain these URLs directly from the *web configuration wizard*, on the monitoring step, by right clicking with the mouse over the corresponding signal. Copy the URL and paste it where needed.

MQTT

edinn M2 adds an MQTT client (3.1.1 Specification) which means it can connect to MQTT brokers (or servers).



To access the data of a signal, you configure a topic (equivalent to an item in OPC).

The obtained token value can come without any wrapper and would be treated in the same way as an OPC token or included in a string in JSON format. Before you can extract the value, you need to configure the JSON template of the data.

How to configure JSON template.

Data example: {"d":{"value":[**620**]},"ts":"2022-09-22T04:52:32.797408"}

The value in this case is the number 620 in square brackets.

Template for this example: {"d":{"value":[**#value#**]},"ts":"2022-09-22T04:52:32.797408"}

Where we have replaced the value with the **#value#** tag, so that edinn can extract this value from the response.

UDL Guide

Introduction

UDL is acronym of User Defined Logic.

The reason of its existance is that, since all edinn installed platforms have the same core, it was necessary a way to introduce logic defined by the user, with these characteristics:

- **Independent to the arrival of new versios of the edinn platform**: if the version of the platform is updated, there is no risk of deleting the logic of the user.
- **Not compromising the stability of the edinn platform**: if the logic of the user fails, the system keeps working.
- **Provides the user with all the power of the market**: because you can use your own development tools, instead of being limited by an edinn proprietary source code editor.

To achieve these objectives, in the edinn platform you can use source code compiled into DLLs. For more information, you can search on the internet about ***dynamic-link library*** (DLL).

> **UDL and API**: regarding the relation between UDL and API, please consider that UDL is the logic defined by the user and the place where it is located. The edinn API is the way in which programs (for example UDLs) read or write from and to the platform.

> **Add-on or App**: in some documents you will find that UDLs are called **add-on** or **App**. The reason is

that more persons know these concepts rather than UDL.

WARNINGS:

Please remember that any issue related with a UDL is not included in the basic support of edinn, and therefore, its revision and/or reparation could imply additional costs.

Knowledge of programming is required to be able to edit, change, compile, install and debug a UDL.

These are the places where the user can introduce his UDL:

- **Monitor**: it is a UDL called by the monitoring service. It allows the execution of actions in a synchronized manner, and at a high speed, with the monitoring. These are just some examples of what can be done:
    - To define virtual signals and counters, which are calculated based upon a user defined logic.
    - To define complex ways of calculating production, based on multiples counters, signals and fields typed by users or read from other systems.
    - Stop and start machinery and turn on and off acoustic and visual beacons.
    - Integration with other systems, where reading and writting from physical devices is required.
    - In general, any other need which is required and is related to monitoring.
- **Fields**: it is a UDL called by the fields configured by the user which appear in certain windows of the operation terminals. These are just some examples of what can be done:
    - To calculate the expiration dates and complex lot numbers which are later printed on the results (products) of the process.
    - To read available lots from other systems.
    - To print and scan labels.
    - In general, any other need which is required and is related to the user operation.
- **Scheduler**: it is a UDL called by the automatic scheduler. These are just some examples of what can be done:
    - To setup a matrix of changes by families, so that the scheduler considers, for example, that if a result is changed to the same family of the previous result, the setup time for the change is less.

Please note that in this document only basic indications and examples are given, and thus, the user can create thousands of possibilities and additional uses.

Note: If you think that your need cannot be satisfied with any UDL, please consult us through our support, because the calls done by the platform to execute UDLs can be extended.

Note: Some examples of UDL are available for download with its source code in the edinn Academy web site, section **installers and developers, downloads, Add-ons**. These UDLs are typically developed with the Microsoft Visual Studio IDE in VB.NET, as it is easier to learn. There are also older UDLs developed in Microsoft Visual Basic 6.

Monitor

Please check the Introduction to the UDL for more information about this type of UDLs.

You will distinguish the Monitor UDLs because they have these file names:

`edinnM2UDL_1234567890.dll`

Where the number will be the identifier of the company which executes the UDL.

User your favourite editor to modify and compile the DLL.

WARNINGS:

Please remember that any issue related with the modification of a UDL is not included in the basic support of edinn, and therefore, its revision and/or reparation could imply additional costs.

Before modifying a UDL, it is recommended to backup the source code and the DLL file (**`edinnM2UDL_1234567890.dll`**). By this, if you experience any problem, you will be able to come back to the previous version.

Follow these steps to install it, in the server (all the following steps must be done typically on this path **C:\Program Files (x86)\edinn\edinnM2\Server**):

1. From the Console, stop the Monitor service.
2. Unregister the UDL, if it was previously installed, by opening a MS-DOS console (**CMD.EXE**) with **Administrator permissions**, and then launching this command (please remember to change the name of the file to your UDL file, with you company identifier):

   `regsvr32 edinnM2UDL_1234567890.dll -u`

   An on screen message should appear indicating that the unregistration was successful.
3. Paste your file **`edinnM2UDL_1234567890.dll`**, overwritting the previous.

4. Register your new DLL with this command:

   ```
   regsvr32 edinnM2UDL_1234567890.dll
   ```

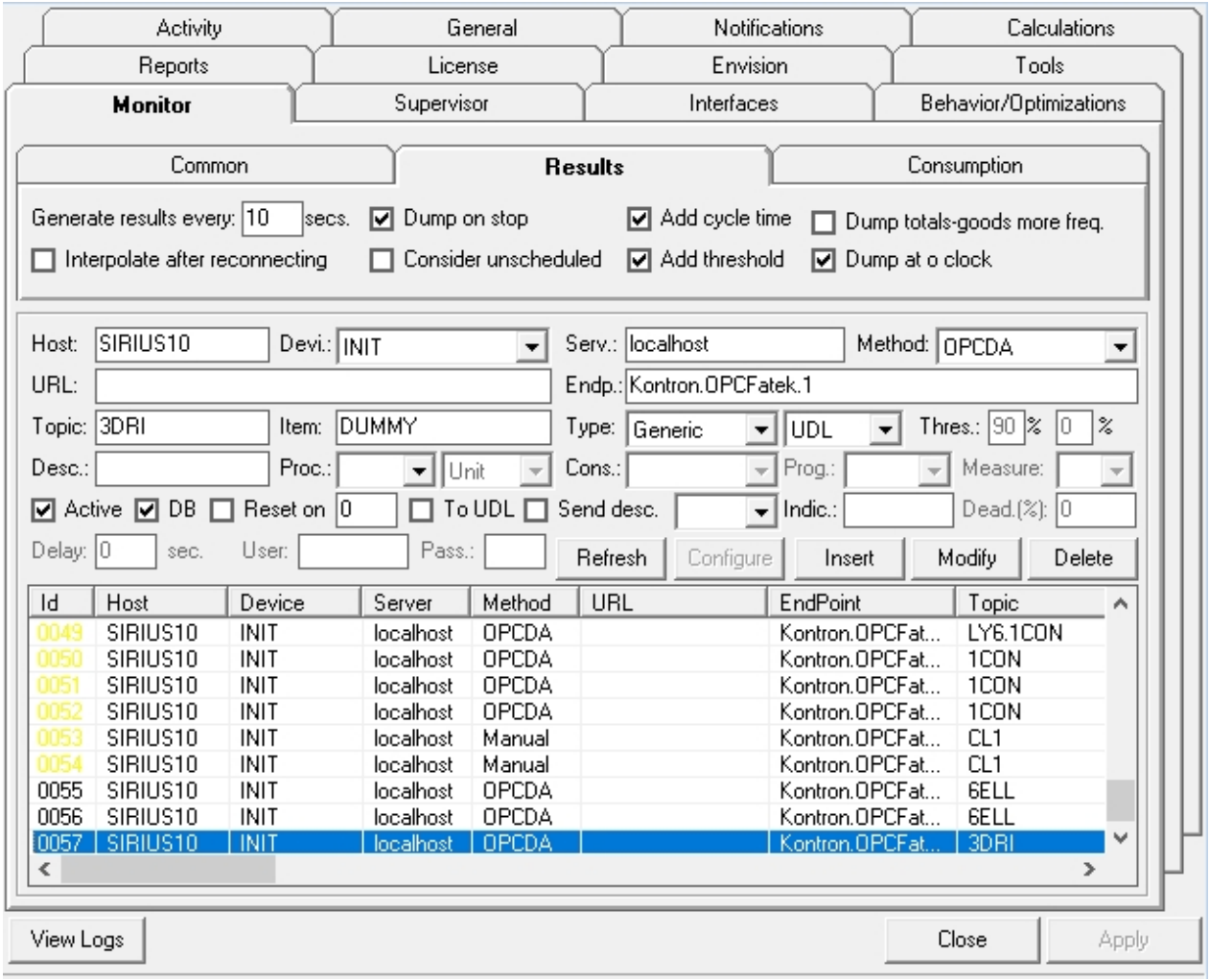
   An on screen message should appear indicating that the registration was successful.

5. Start the Monitor service.

6. Check in the logs of the system that the monitor connects and correctly executes your UDL.

If the UDL is not doing what you expected, introduce trace (writting in the log of the system) inside your UDL to know what is is doing and to debug it until reaching your objective.

In order to activate these UDLs, you need at least one item in your Monitor list in the Console, to be of UDL type. For example, you can create a DUMMY Generic UDL item, like this:



If you want to debug these UDLs, you can use **Microsoft Visual Studio**, start the edinn Monitor service and do *Debug\Attach to Process* from the IDE to the task **edinnM2OPCDA_C.exe**

Fields

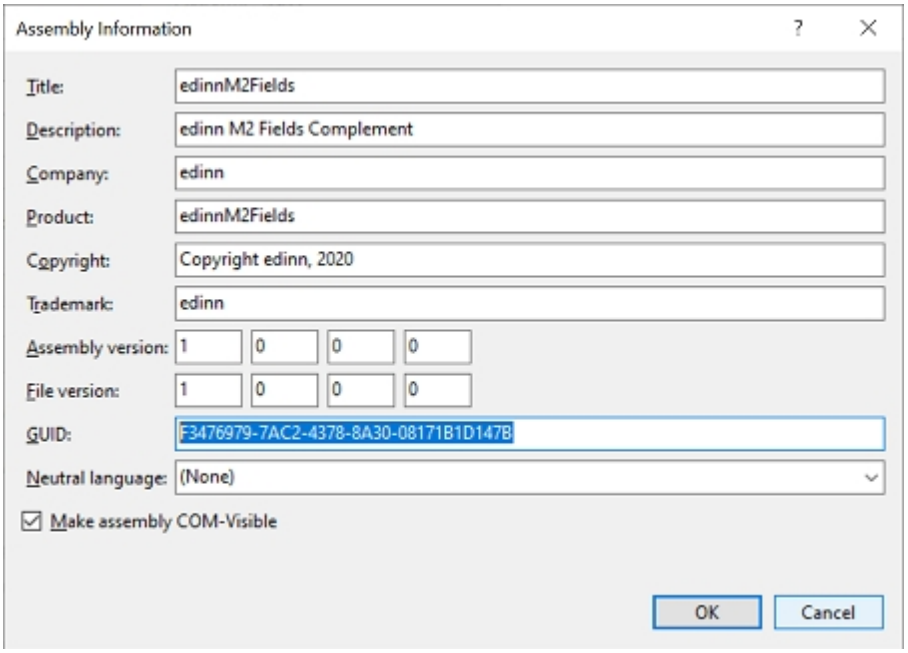
Please check the Introduction to the UDL for more information about this type of UDLs.

You will distinguish the Monitor UDLs because they have these file names:

`edinnM2Fields.dll`

User your favourite editor to modify and compile the DLL.

All fields UDLs must have this GUID: F3476979-7AC2-4378-8A30-08171B1D147B



WARNINGS:

Please remember that any issue related with the modification of a UDL is not included in the basic support of edinn, and therefore, its revision and/or reparation could imply additional costs.

Before modifying a UDL, it is recommended to backup the source code and the DLL file (**`edinnM2UDL_1234567890.dll`**). By this, if you experience any problem, you will be able to come back to the previous version.

Currently the platform does not allow the same server to provide to the terminals different field UDLs for different companies, therefore, these UDLs will require that your server only hosts one company, or that all companies hosted by your server use the same fields UDL.

Follow these steps to install it:

1. Stop the win32 terminal.

2. **Unregister** the UDL, if it was previously installed, by opening a MS-DOS console (**CMD.EXE**) with **Administrator permission**, and launching the next command:

   `regsvr32 edinnM2Fields.dll -u`

   An on screen message should appear indicating that the unregistration was successful.

3. Paste your file **`edinnM2Fields.dll`**, overwritting the previous.

4. Register your new DLL with this command:

   `regsvr32 edinnM2Fields.dll`

   An on screen message should appear indicating that the registration was successful.

5. Start the win32 terminal.

6. Be sure to have created calculated fields where the default value is the name of the class of your UDL.

7. Check, on the terminal, that your UDL is behaving as you need.

If the UDL is not doing what you expected, introduce trace (on screen messages) inside your UDL to know what is is doing and to debug it until reaching your objective.

Once your UDL is behaving exactly as you wish, the distribution and registration of the fields UDL can consume a lot of time and effort. If you want to do that process automatically, you can copy your UDL file into the server in the folder of the distribution of the terminal files, typically here:

`C:\Program Files (x86)\edinn\edinnM2\Server\Client\`**`edinnM2Fields.dll`**

Then rename the files M2UPD_EDINNM2_YYYYMMDD.txt, where YYYYMMDD is an inverted date (year, month and day), by changing the date to a newer date.

Now you can restart the terminals which will receive and register your UDL.

Scheduler

Please check the Introduction to the UDL for more information about this type of UDLs.

You will distinguish the Scheduler UDLs because they have these file names:

`edinnM2ProjectUDL_6497308612.dll`

Where the number will be the identifier of the company which executes the UDL.

User your favourite editor to modify and compile the DLL.

WARNINGS:

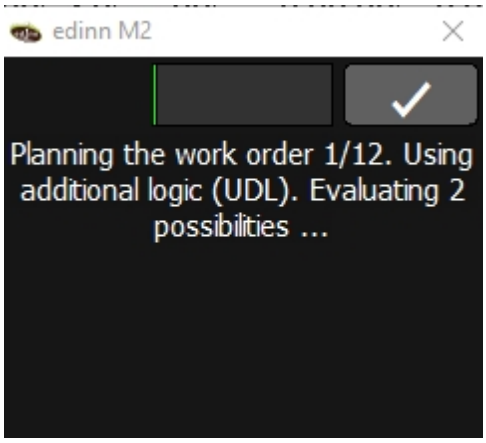
Please remember that any issue related with the modification of a UDL is not included in the basic support of edinn, and therefore, its revision and/or reparation could imply additional costs.

Before modifying a UDL, it is recommended to backup the source code and the DLL file (`edinnM2UDL_1234567890.dll`). By this, if you experience any problem, you will be able to come back to the previous version.

To install it, copy and paste the file `edinnM2ProjectUDL_6497308612.dll` where the executable of your win32 terminal is, normally in **C:\Program Files (x86)\edinn\edinnM2\Client**

You do not need any other registration operation, as the scheduler will link with the UDL located where the win32 terminal executable file is.

When the terminal finds and uses your UDL, it will indicate it in the progress window of the automatic scheduler:



If you wish to debug your UDL, it is recommended to put code which dumps trace to a text file.

API guide

API GUIDE

Below you will find the API guide of the 4.0 Industry Open Platform Edinn.

Make your own developments, improve and adapt any functionality, configure the visualization of the user interface or even integrate any external application.

Introduction

This section covers the edinn API, which provides a complete collection of resources to develop customized solutions using the platform. The edinn API is a REST API.

There are these section in this guide:

- **Functions**: calls to the edinn system to obtain or introduce data or configuration.
- **Subscriptions**: calls from the edinn system to obtain or introduce data or configuration.

WARNINGS:

Please remember that any issue related with something developed with the API is not included in the basic support of edinn, and therefore, its revision and/or reparation could imply additional costs.

Knowledge of programming is required to be able to edit, change, compile, install and debug a UDL.

Methods from the API which are indicated to be ISA-95 *Compliant* require the Integration Module.

As any other API REST, you can try or use the edinn API through different methods. Let us see examples through:

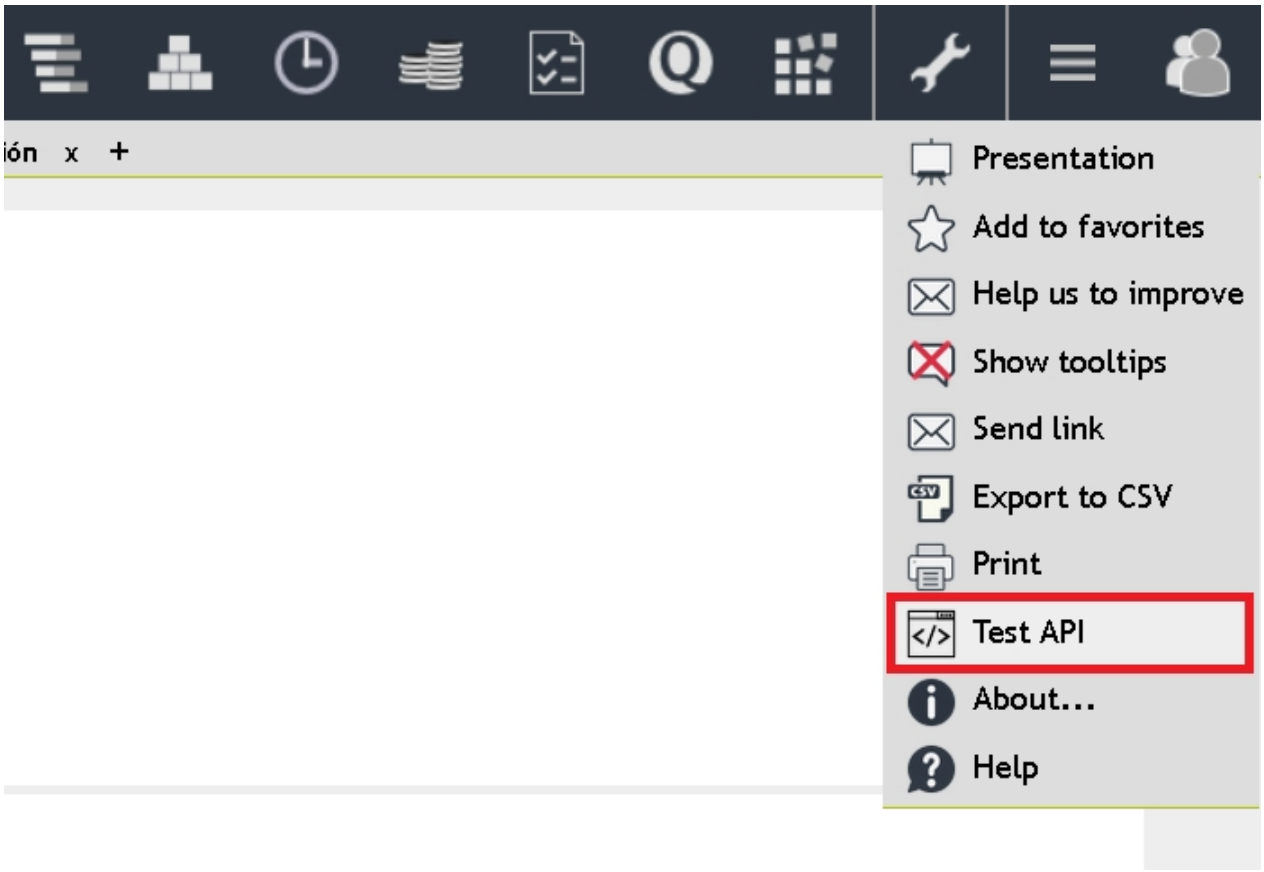
- edinn Reports
- Microsoft .NET
- Microsoft Excel

How to test or use?

edinn Reports

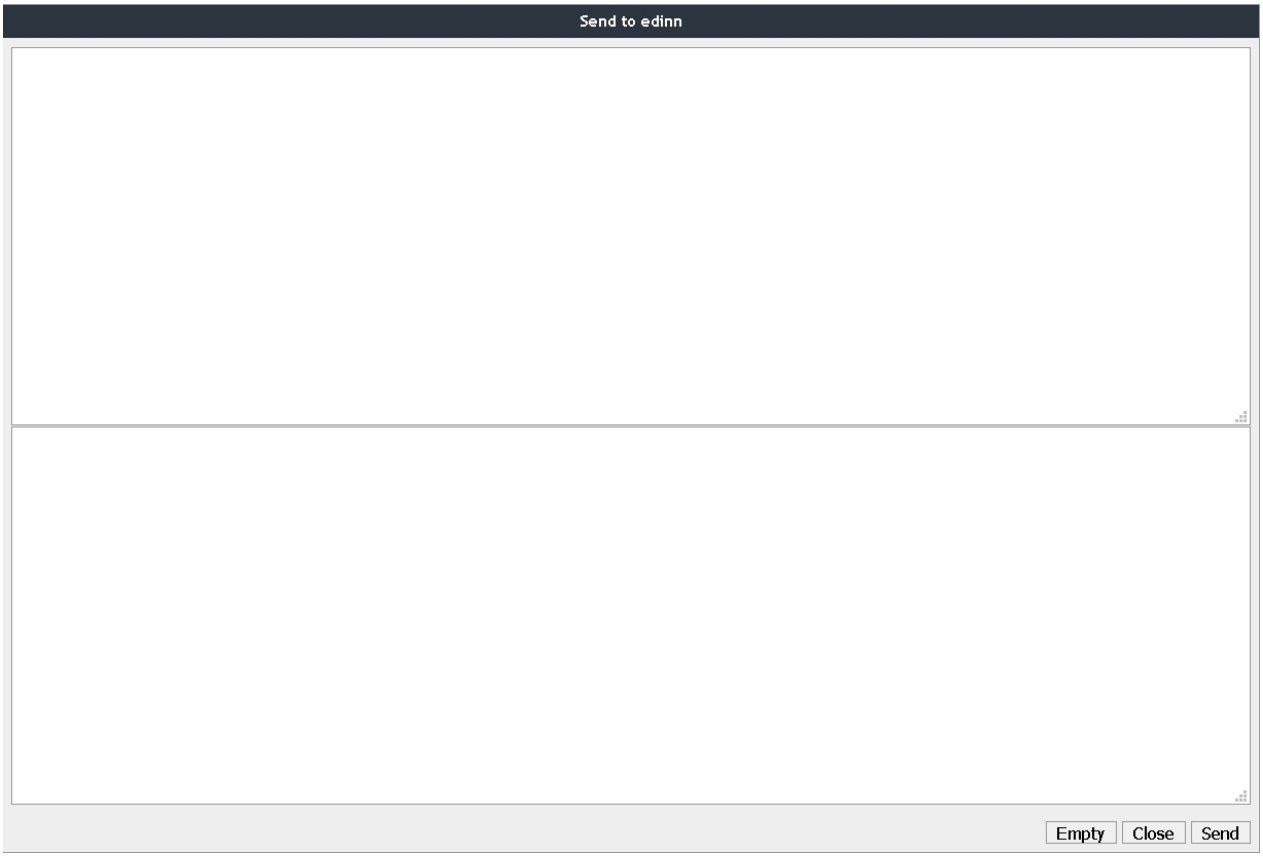
Edinn API functions can be tested without programming anything through the reporting tool.

First, we access the reports. In the menu, we select the option “Test API”:



A window will appear in which we will put in the upper text box, the call to the API that we want to test. When you hit the "send" button, the answer will appear in the lower text box.

The window has a button to empty the text boxes and another to close the window and return to reports.



In the help of each API call there is an example to test it using this system.

MS .NET

To integrate your applications using Microsoft .NET with edinn through the API, you will need a working environment **in your preferred programming language**.

In this example we will use VB.NET to connect to the edinn platform, obtain the tree of areas and processes and disconnect from the platform.

This example needs the NuGet package Newtonsoft.Json package, which you can install from the Microsoft Visual Studio IDE.

We can explain the example in three steps:

1st Create a UWP Application:

Open Visual Studio .Net.

Select "Create a new project" in Get Started section.

Type "uwp basic" in the search box.

Select Blank App (Universal Windows). Note that there must have the labels "Visual Basic" and "UWP".

Choose the project name, location and solution name you prefer.

2nd Paste the example code:

In the solution explorer, make double click on MainPage.xaml.

We can see now the preview of thr window and the corresponding xml code of the interface.

Paste the following code inside the grid section.

```
<StackPanel x:Name="frmLogin" HorizontalAlignment="Stretch">
<TextBlock Text="edinn API tester" FontSize="14"
FontWeight="Bold" />
<TextBox x:Name="txtCompany" PlaceholderText="company" />
<TextBox x:Name="txtUser" PlaceholderText="user" />
<TextBox x:Name="txtPass" PlaceholderText="password" />
<Button x:Name="cmdLogin" Content="Login" Click="cmdLogin_Click"
HorizontalAlignment="Stretch" />
<TextBox x:Name="txtSession" Text="" TextWrapping="Wrap"
PlaceholderText="session"/>
<Button x:Name="cmdTree" Content="Get Tree"
HorizontalAlignment="Stretch" Click="cmdTree_Click"/>
<Button x:Name="cmdLogout" Content="Logout"
HorizontalAlignment="Stretch" Click="cmdLogout_Click"/>
<TextBox x:Name="txtResponse" Text="" TextWrapping="Wrap"
IsReadOnly="True" Height="180" PlaceholderText="response"/>
</StackPanel>
```

Open de VB code belongin to the inteface in MainPage.xaml.vb

```
' The Blank Page item template is documented at https://go.microsoft.com/fwlink/?
LinkId=402352&clcid=0x409

''' <summary>
''' An empty page that can be used on its own or navigated to within a Frame.
''' </summary>
Public NotInheritable Class MainPage
    Inherits Page

End Class
```

Import the Http library at the top of the code.

```
' The Blank Page item template is documented at https://go.microsoft.com/fwlink/?
LinkId=402352&clcid=0x409

Imports System.Net.Http

''' <summary>
''' An empty page that can be used on its own or navigated to within a Frame.
''' </summary>
Public NotInheritable Class MainPage
    Inherits Page

End Class
```

Paste the following code after the class declaration.

```
' The Blank Page item template is documented at https://go.microsoft.com/fwlink/?
LinkId=402352&clcid=0x409

Imports System.Net.Http

''' <summary>
''' An empty page that can be used on its own or navigated to within a Frame.
''' </summary>
Public NotInheritable Class MainPage
    Inherits Page
```

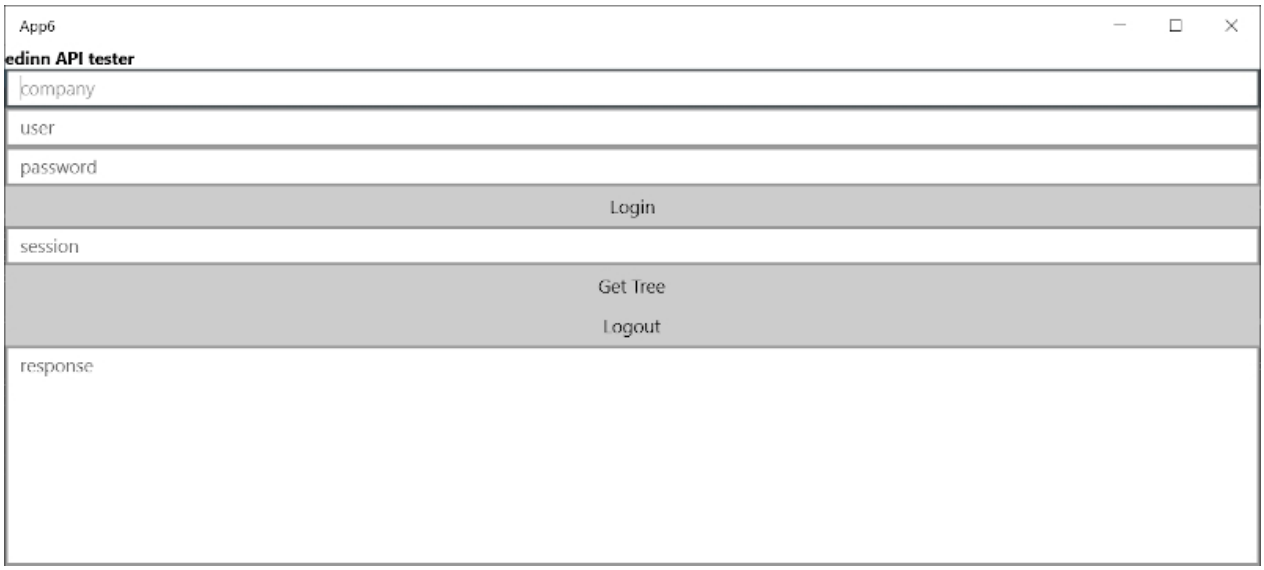
```
  Private Sub cmdLogin_Click(sender As Object, e As RoutedEventArgs) Handles
cmdLogin.Click
    txtResponse.Text = ""
    Dim lStr = session_post()
    txtResponse.Text = lStr
  End Sub

  Private Sub cmdTree_Click(sender As Object, e As RoutedEventArgs)
    txtResponse.Text = ""
    Dim lStr = tree_get()
    txtResponse.Text = lStr
  End Sub

  Private Sub cmdLogout_Click(sender As Object, e As RoutedEventArgs)
    txtResponse.Text = ""
    txtResponse.Text = session_delete()
  End Sub

  Private Function session_post() As String
    Dim client As New HttpClient
    ''' IMPORTANT! Change the url according to your server IP ad PORT '''
    Dim url As String = "http://localhost:8080/sessions"
    Dim values As New Dictionary(Of String, String)

    values("company") = txtCompany.Text
    values("user") = txtUser.Text
    values("password") = txtPass.Text

    Try
      Dim content As New FormUrlEncodedContent(values)
      Dim response As New HttpResponseMessage
      response = client.PostAsync(url, content).Result
      Return response.Content.ReadAsStringAsync().Result

    Catch ex As Exception
      Return ex.Message

    End Try
  End Function

  Private Function session_delete() As String
    Dim client As New HttpClient
    ''' IMPORTANT! Change the url according to your server IP ad PORT '''
    Dim url As String = "http://localhost:8080/sessions/" & txtSession.Text & "?
company=" & txtCompany.Text

    Try
      Dim request As New HttpRequestMessage(HttpMethod.Delete, New Uri(url))
      Dim response As New HttpResponseMessage
      response = client.SendAsync(request).Result
      Return response.Content.ReadAsStringAsync().Result
```

```
    Catch ex As Exception
      Return ex.Message

    End Try

  End Function

  Private Function tree_get() As String
    Dim client As New HttpClient
    ''' IMPORTANT! Change the url according to your server IP ad PORT '''
    Dim url As String = "http://localhost:8080/tree?company=" & txtCompany.Text &
"&session=" & txtSession.Text

    Try
      Dim response As New HttpResponseMessage
      response = client.GetAsync(New Uri(url)).Result
      Return response.Content.ReadAsStringAsync().Result

    Catch ex As Exception
      Return ex.Message

    End Try
  End Function

End Class
```

3rd Customize and test it

If we have our own server, a company and a user in edinn, we can test the application inmediatly.

We should only change the url adress that in the example code begins wthit "http://localhost:8080/" with the adress of our own server.



How to use the test application

To do login: Fill in the "company", "user" and "password" field and click "Login". In the “response” box you will see the session code.

Get the tree of areas and processes: To get the tree of areas and processes, fill in the field

"company", paste the session code in the "session" box and click "Get Tree". In the “response” box you will see the tree of areas and processes.

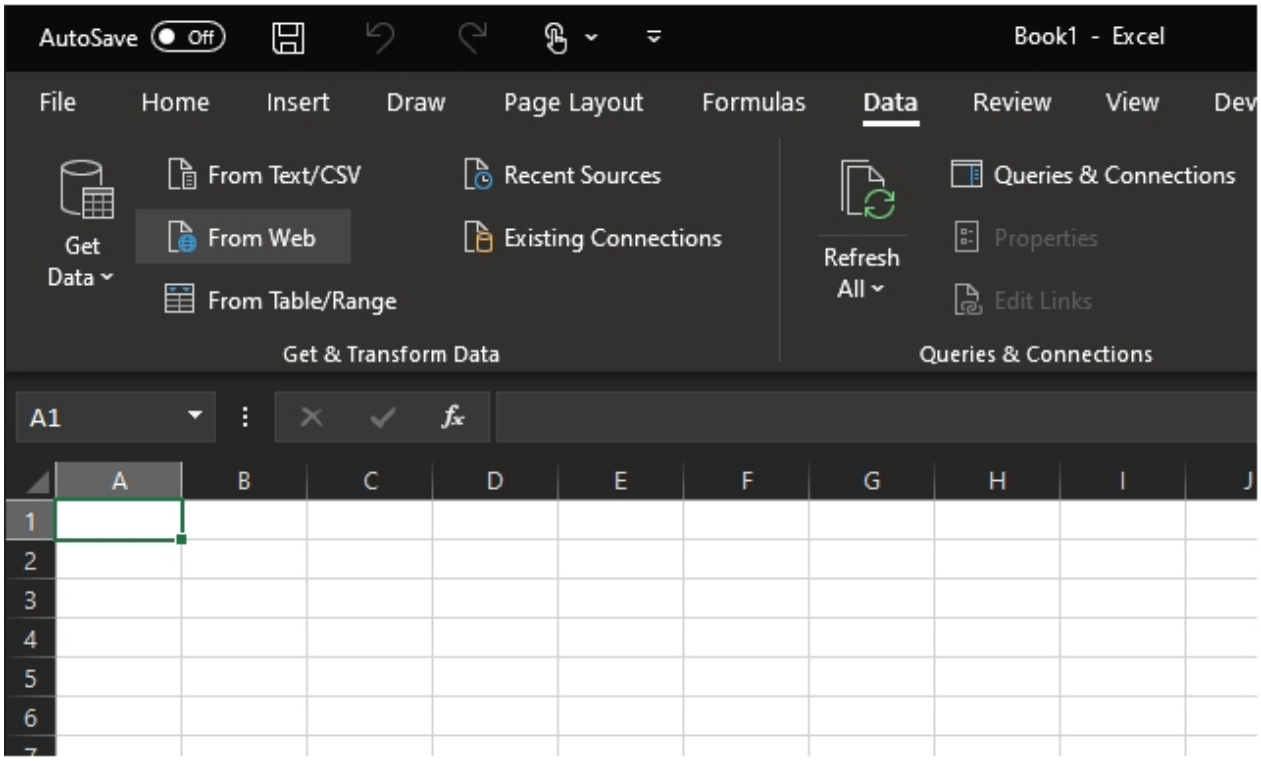
To logout: To end a session, fill in the "company" field, paste the session code in the "session" box and press Logout. In the “response” box you will see the response.

From this point on, you can implement the calls to the available functions, by reading the API help topics.

MS Excel

Edinn API functions can be tested or used using Microsoft Excel. I this document we will see an example.

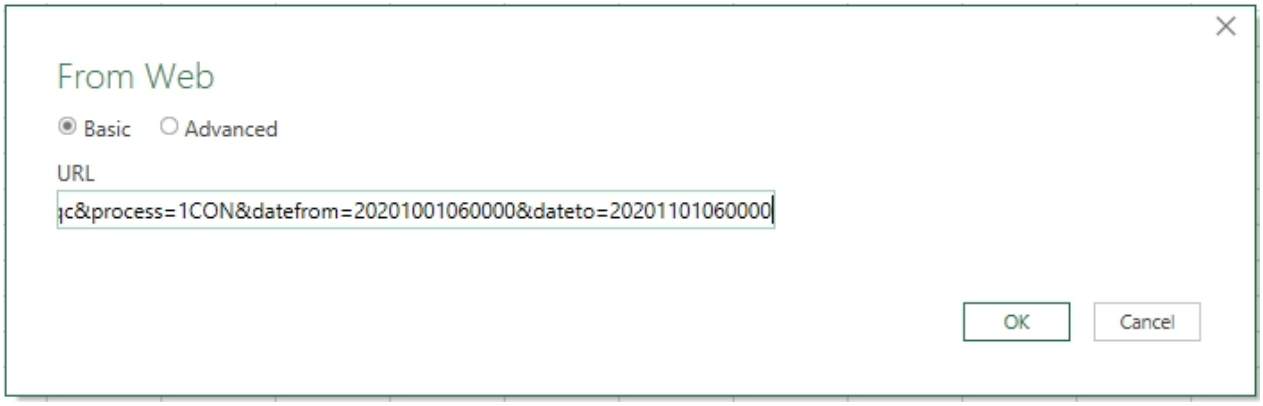
From MS Excel, in the top menu **Data**, select **From Web**:



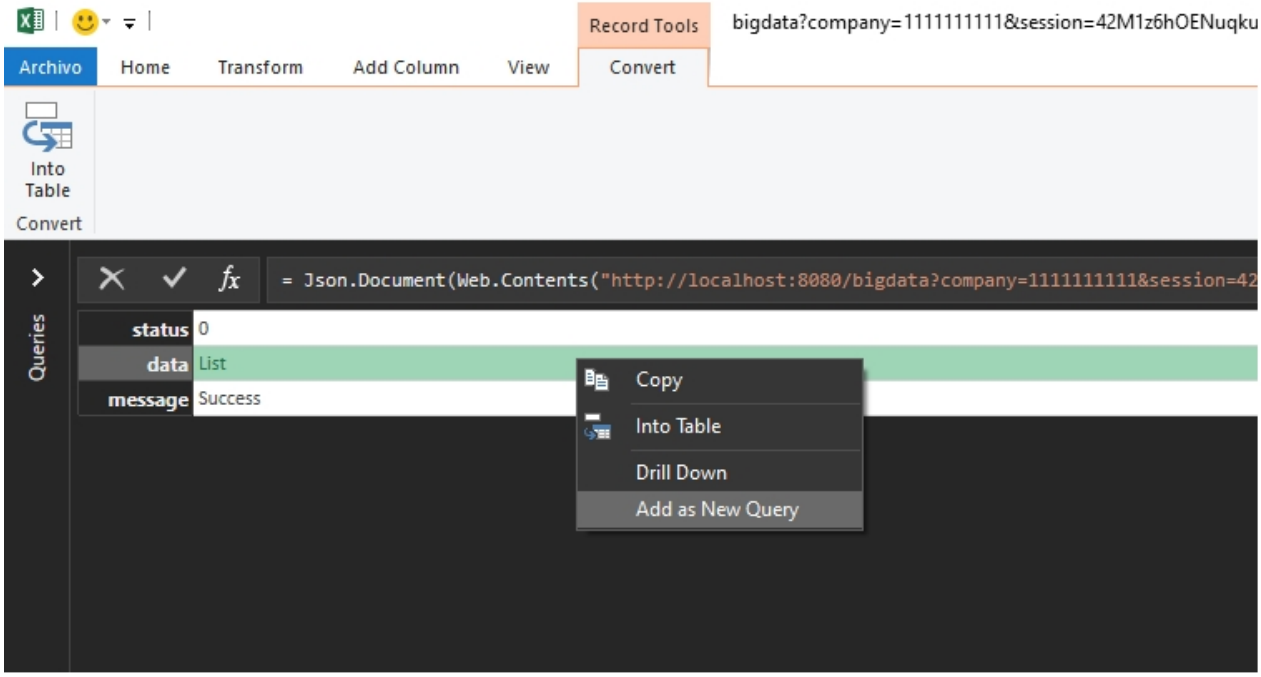
Type, into the URL, what is indicated in the edinn API for Big Data, for example:
(IMPORTANT! Change the url according to your server IP ad PORT)

```
http://localhost:8080/bigdata?
company=1111111111&session=42M1z6hOENuqkuuro4CJVHv6UAtsih4X9TUA5TKT1S
N13LkYLdv7AdpYxZqc&process=1CON&datefrom=20201001060000&dateto=202011
01060000
```

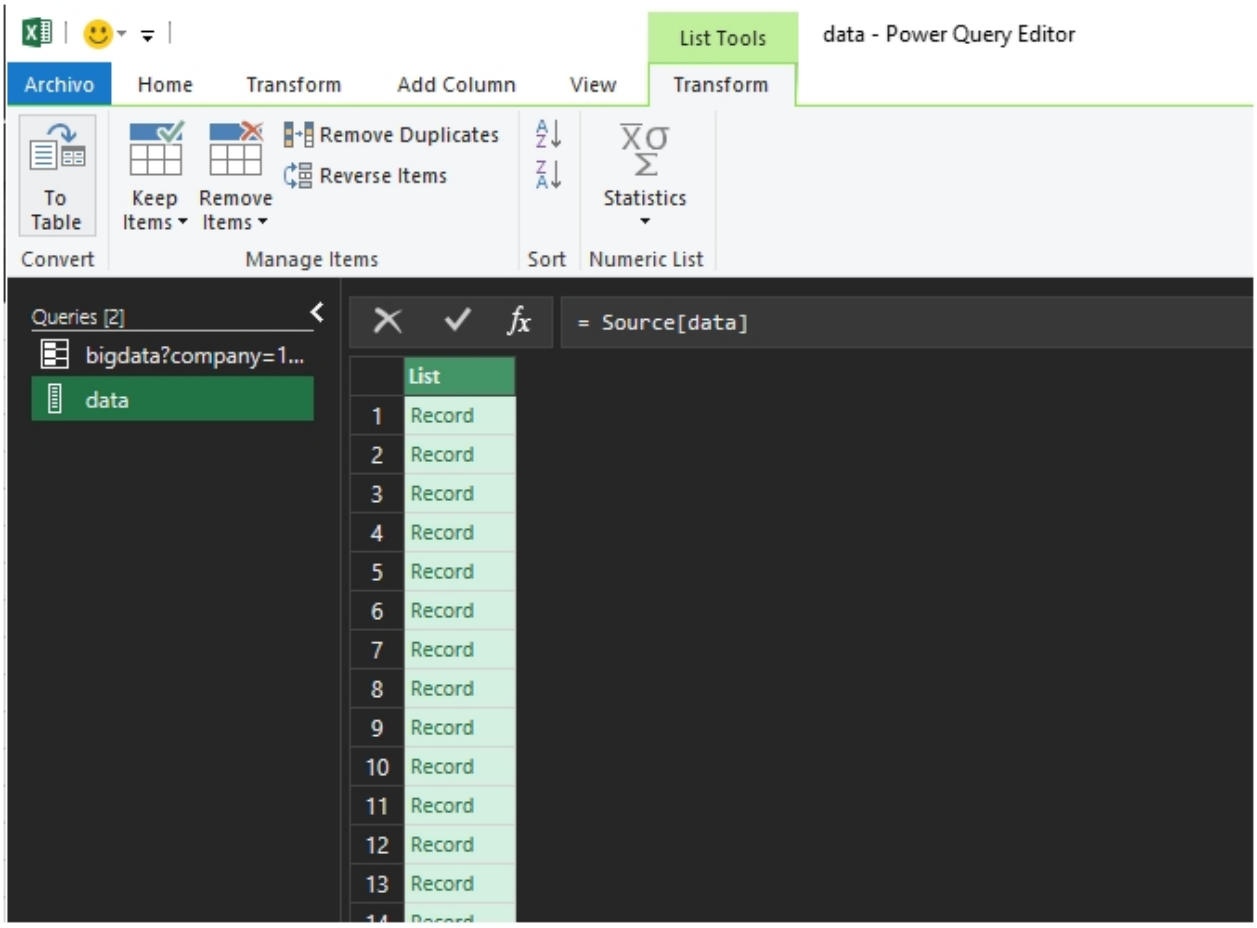
And click on **Ok**:



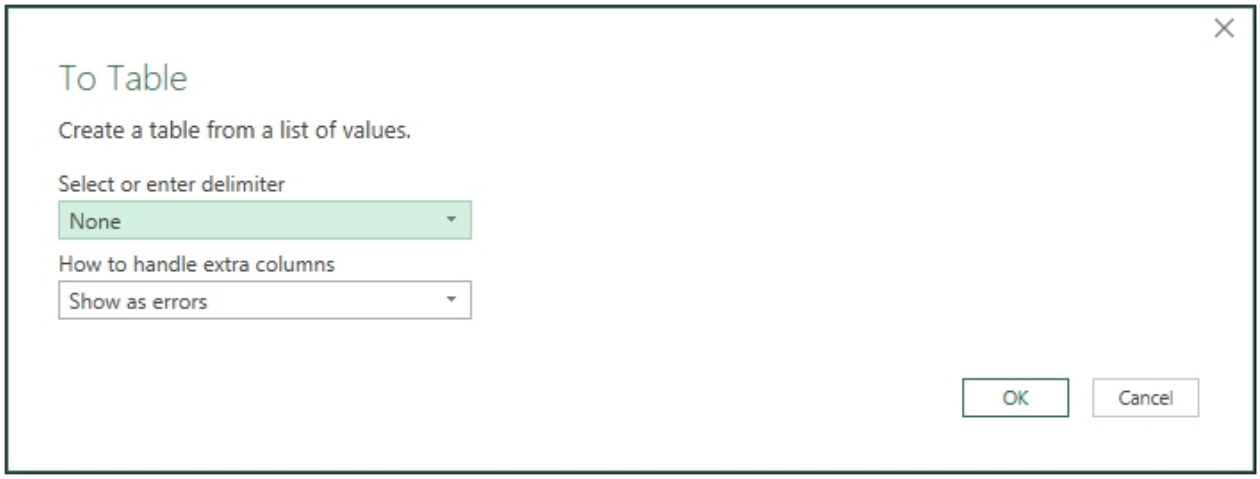
When the response appears, select **data**, put the **mouse pointer on the right side**, click **right mouse button** and choose "**Add as a new query**":



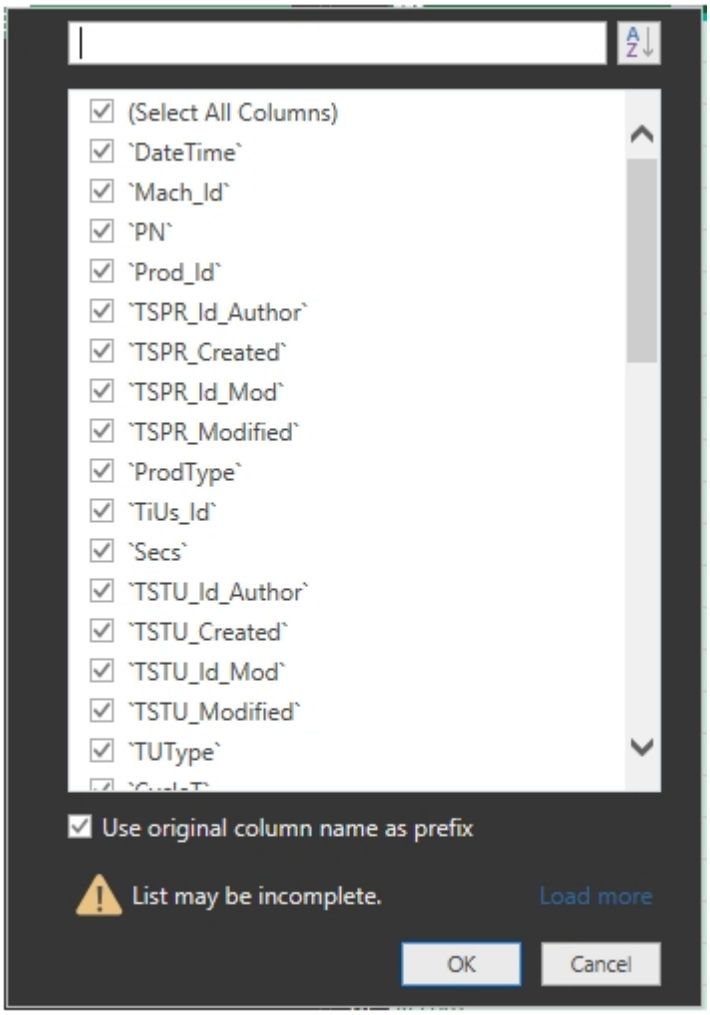
When the records appear, click "**To Table**" on the top left corner:



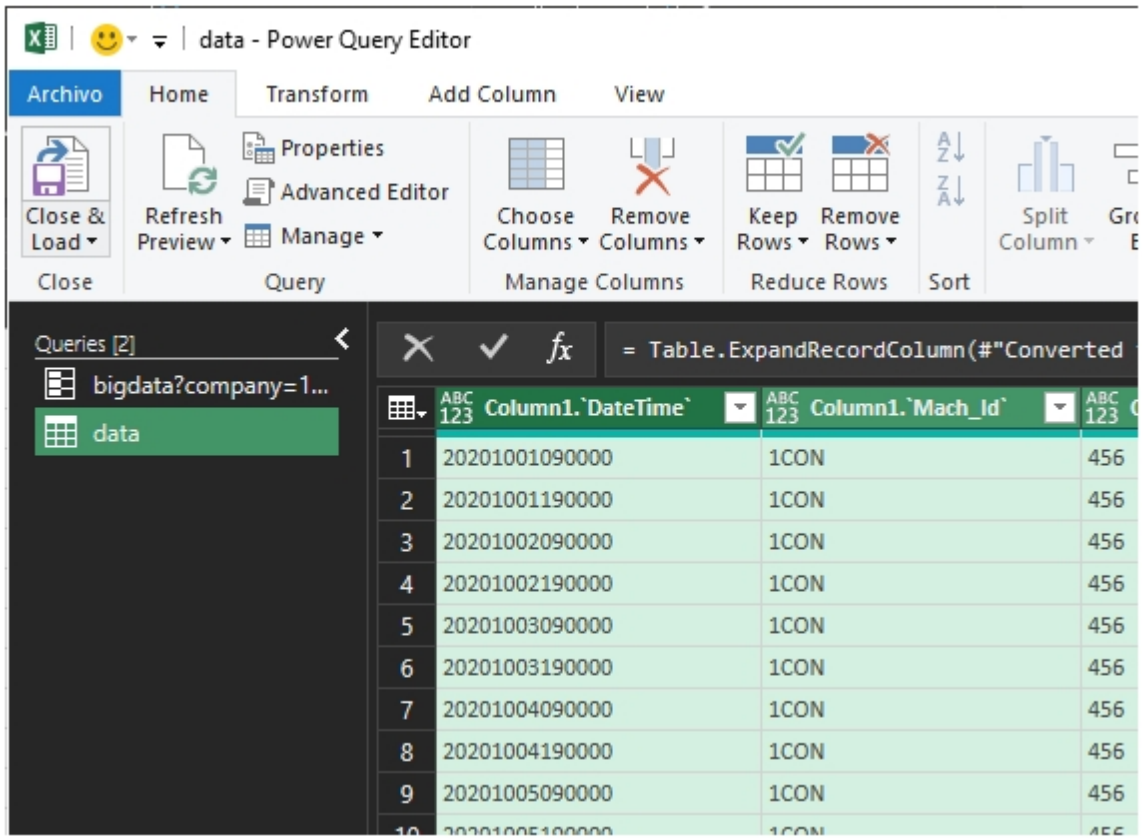
Then, click **Ok** without changing anything:



Click on the **button**  which is on the top of the column, and click Ok on the following window:



When data appears, **click** on **Close & Load** on the top left corner:



And data will be already in our excel sheet.

For more information about the meaning of this data, please check the Big Data call from the edinn API.

Python

You can test or use the edinn API functions with Python. This document will guide you through an example.

1º Install Python (if not already installed):

- Check if Python is installed by opening a terminal window and running the command `python --version`
- If Python is not installed, download and install it from https://www.python.org/downloads/

2º Create a Python file:

- Go to your development environment.
- Create a new Python file with the extension ".py" (e.g., example.py).

3º Copy and paste the following code:

```python
import requests
import json

# --- Edinn API Configuration ---
EDINN_API_URL = "http://localhost:8080"  # Replace with your edinn API URL
EDINN_COMPANY = "1231231234"  # Replace with your company name
EDINN_USER = "user"  # Replace with your username
EDINN_PASSWORD = "123asd"  # Replace with your password
EDINN_PROCESS = "A0L1"
EDINN_DATEFROM = "202407180000000"
EDINN_DATETO = "202407190000000"

# --- Function to create an edinn session ---
def create_edinn_session():
    url = f"{EDINN_API_URL}/sessions"
    data = {
        "company": EDINN_COMPANY,
        "user": EDINN_USER,
        "password": EDINN_PASSWORD
    }
    try:
        response = requests.post(url, data=data)
        response.raise_for_status()
        session_id = response.json()["data"]
        print(f"Created edinn session: {session_id}")
        return session_id
    except requests.exceptions.RequestException as e:
        print(f"Error creating edinn session: {e}")
        return None

# --- Function to delete an edinn session ---
def delete_edinn_session(session_id):
    url = f"{EDINN_API_URL}/sessions/{session_id}?company={EDINN_COMPANY}"
    try:
        response = requests.delete(url)
        response.raise_for_status()
        print(f"Deleted edinn session: {session_id}")
    except requests.exceptions.RequestException as e:
        print(f"Error deleting edinn session: {e}")

def get_results():
    url = f"{EDINN_API_URL}/results?company={EDINN_COMPANY}
&session={session_id}&process={EDINN_PROCESS}&datefrom={EDINN_DATEFROM}
&dateto={EDINN_DATETO}"

    try:
        response = requests.get(url)
        response.raise_for_status()  # Check for HTTP errors
        data = response.json()  # Parse the JSON response
        print(f"Data from edinn:")
        print(json.dumps(data, indent=4))  # Pretty-print the JSON data
    except requests.exceptions.RequestException as e:
        print(f"Error finding data: {e}")
```

```
# --- Create edinn session ---
session_id = create_edinn_session()
get_results()
delete_edinn_session(session_id)
```

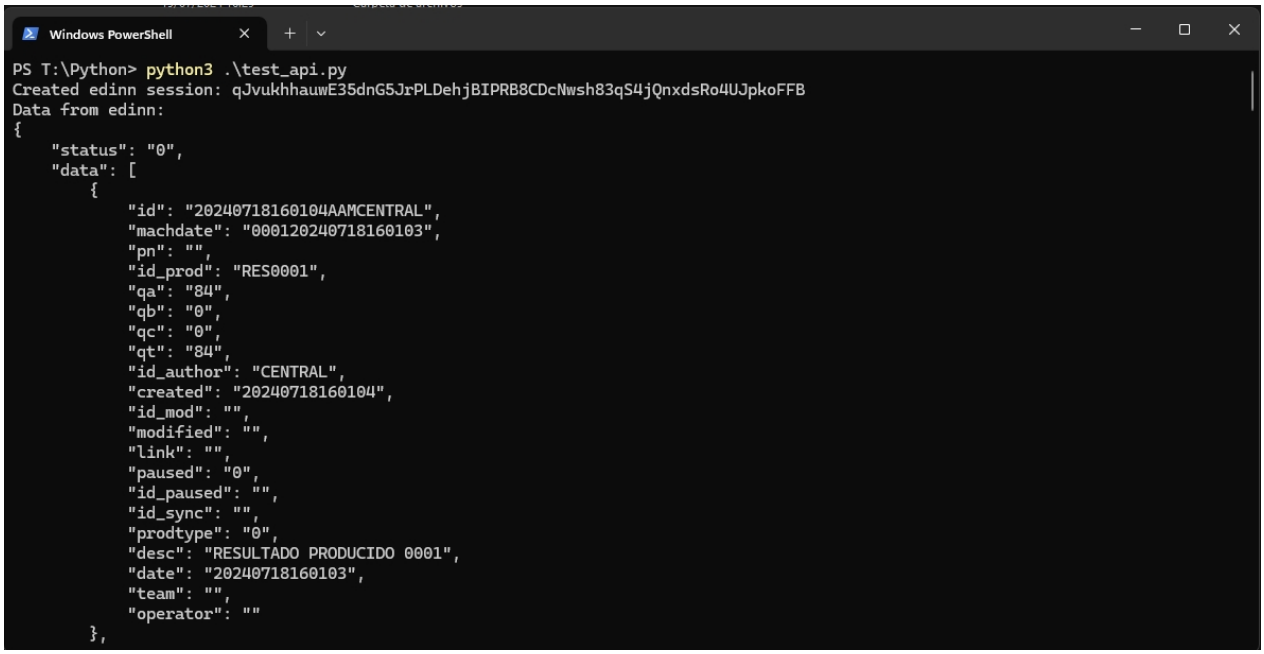
4º Configure your edinn API credentials:

- Replace the placeholder values in the `Edinn API Configuration` section with your actual edinn API URL, company name, username, and password.

5º Run the script:

- Open a terminal window.
- Navigate to the directory where you saved your Python script.
- Execute the script using the command `python nombre_de_app.py` (replace your_script_name.py with the actual name of your file).

This script will connect to the edinn API, create a session, retrieve data, and then delete the session. The retrieved data will be printed to the console in a formatted JSON structure.



Jupyter

Jupyter

Edinn API functions can be tested or used using Jupyter. This document will guide you through an example.

1º Install Python and Jupyter (if not already installed):

- Verify if Python is installed by opening a terminal window and running the command `python --version`
- If Python is not installed, download and install it from https://www.python.org/downloads/.
- Verify if Jupyter is installed by opening a terminal window and running the command `jupyter --version`
- If Jupyter is not installed, run the command `pip install jupyter` to install it.

2º Create a Jupyter file:

- Open or create any empty repository.
- Open this folder from the terminal.
- Run the command `jupyter notebook`
- Create a new file by clicking the "New" button.

3º Copy and paste the following code:

Cell 1: Install Required Libraries

```
# Install the requests library if not already installed
!pip install  requests
```

Cell 2: Configuration

```
# --- Edinn API Configuration ---
EDINN_API_URL = "http://localhost:8080"  # Replace with your edinn API URL
EDINN_COMPANY = "1231231234"  # Replace with your company name
EDINN_USER = "user"  # Replace with your username
EDINN_PASSWORD = "123asd"  # Replace with your password
EDINN_PROCESS = "A0L1"
EDINN_DATEFROM = "202407180000000"
EDINN_DATETO = "202407190000000"
```

Cell 3: Session Management Functions

```
import requests
import json
# --- Function to create an edinn session ---
def create_edinn_session():
    url = f"{EDINN_API_URL}/sessions"
    data = {
        "company": EDINN_COMPANY,
        "user": EDINN_USER,
        "password": EDINN_PASSWORD
    }
    try:
        response = requests.post(url, data=data)
        response.raise_for_status()
```

```
        session_id = response.json()["data"]
        print(f"Created edinn session: {session_id}")
        return session_id
    except requests.exceptions.RequestException as e:
        print(f"Error creating edinn session: {e}")
        return None

# --- Function to delete an edinn session ---
def delete_edinn_session(session_id):
    url = f"{EDINN_API_URL}/sessions/{session_id}?company={EDINN_COMPANY}"
    try:
        response = requests.delete(url)
        response.raise_for_status()
        print(f"Deleted edinn session: {session_id}")
    except requests.exceptions.RequestException as e:
        print(f"Error deleting edinn session: {e}")

def get_results():
    url = f"{EDINN_API_URL}/results?company={EDINN_COMPANY}
&session={session_id}&process={EDINN_PROCESS}&datefrom={EDINN_DATEFROM}
&dateto={EDINN_DATETO}"

    try:
        response = requests.get(url)
        response.raise_for_status()  # Check for HTTP errors
        data = response.json()  # Parse the JSON response
        print(f"Data from edinn:")
        print(json.dumps(data, indent=4))  # Pretty-print the JSON data
    except requests.exceptions.RequestException as e:
        print(f"Error finding data: {e}")
```

Cell 5: Execution python

```
# --- Create edinn session ---
session_id = create_edinn_session()
get_results()
delete_edinn_session(session_id)
```

4º Configure your edinn API credentials:

- Replace the placeholder values in the Edinn API Configuration section with your actual edinn API URL, company name, username, and password.

5º Run the script:

This script will connect to the edinn API, create a session, retrieve data, and then delete the session. The retrieved data will be printed to the console in a formatted JSON structure.

First steps

In edinn, in order to capture data, turn this data into useful information and then take actions, you first need to configure (POST):

1. A process: all acquired data must be associated to a final entity called process. For more information about this very important concept, please read this.
2. Some statuses: during time, the process sometimes will be working and sometimes will be stopped. You need different statuses to reflect the different time usages of the process.
3. Some results: a process produces results as its output. You need results to reflect the different outputs of the process.
4. Some relations: you need to indicate certain configuration data when a process is in a certain status and producing a certain result.
5. Some IoT: you need to indicate to the edinn platform where it should read data from. In order to monitor the status of a process, you need, **at least**, 2 items of IoT: one for the variable number 1 and one for the variable number 0, the total, which will use the calculation type to calculate the total production based on the previous IoT item.

After completing these steps, you now have a monitored process. Now you could:

6. Read a value from IoT (GET) to obtain the value of the item and to know when it changed for the last time.
7. Insert a result (POST) to indicate to edinn the type of results that will be obtained through IoT.
8. Justify (PATCH) the different failure (FAI) statuses that will be generated when the process is not working, in order to obtain later the paretto of the losses and a correct OEE.

You could **optionally**:

9. Create an area to visually group your processes.
10. Set quality assurement parameters.
11. Create schedule orders.

Your objective is to create a virtual model of your organization, and then obtain the multiple benefits of the edinn platform, for example:

- Calculate OEE in real time or from whatever the period.
- Obtain its productive losses.
- Create dashboards to monitor its performance.
- Perform Quality SPC control of it.

- Manage its preventive maintenance.
- Etc.

If you do not find the API function that you need to create and manage the virtual model of your organization, please contact edinn technical support.

Functions

This section covers the available functions in the API. If you do not find the function that you need, please request it at the edinn support.

Test

This section describes the test function.

GET /test

This method performs a test of the operation of the main components of the API (client and reports).

Request parameters:

- **company [required]:** Company name or ID.

Response:

- **status:** returns "0" on success and an error code otherwise.
- **data:** Returns the status of the client and reports components. The status can be true (working correctly) or false (failed).
- **message:** "Success" in case of success or a descriptive information of the error in any other case.

Example:

Request:

{"type":"get",

"url":"test",

"data":"company=testco"}

Reponse (success):

{"status":"0",

"data":{"client":true,"reports":true},

"message":"Success"}

Reponse (error):

{"status":"1000",

"data":"<a href='http:\/\/mayen19\/edinnM2\/help\/en\/API_Test.html'>for additional information follow this link<\/a>",

"message":"Some parameters are missing"}

Sessions

This section handles authentication.

GET /sessions/{session}

This method gets the name of the user to whom the provided session belongs or error in case the session does not exist.

Request parameters:

- **company [obligatorio]:** The company name or id.
- **session [obligatorio]:** the session code.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the code of the user to whom the session belongs in case of success or the link to the request help in case of failure.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Request:

{"type":"get",
"url":"sessions/neQXoKWYLALJ1yVjr5d2ANOhZrwp52p5qeyM7WdzAi96rJ5QKmGHsIyF5tzV",
"data":"company=testco"}

Reponse (success):
{"status":"0",
"data":"ADMIN",
"message":"Success"}

Reponse (error):
{"status":"1007",
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Sessions.html'>for additional information follow this link<\/a>",
"message":"Session not stablished"}

POST /sessions

This method creates a new session, it allows you to log in the API. You have to provide your credentials and the method return you a session Id. You need this session Id in order to use all the other resources of the API.

If your company is not on that server, the correct server will be specified in the response message.

Request parameters:

- **company [mandatory]**: The company name or id.
- **user [mandatory]**: The user identifier.
- **password [mandatory]**: The user password.
- **device**: The device id.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the session code on success or the link to the request help on error.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"post",
"url":"sessions",
"data":"company=testco&user=admin&password=1234&device=init"}

Reponse (success):
{"status":"0",
"data":"neQXoKWYLALJ1yVjr5d2ANOhZrwp52p5qeyM7WdzAi96rJ5QKmGHsIyF5tzV",
"message":"Success"}

Response (error):
{"status":"1004",
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Sessions.html'>for additional information follow this link<\/a>",
"message":"Incorrect credentials"}

DELETE /sessions/{session}

This method deletes a specific session, it allows you to log out the API.

Request attribute:

- **session**: The Id of the session to delete.

Request parameters:

- **company**: The company name or id.

Response:

- **status**: Returns "0" on success, and an error code in any other case.
- **data**: true if the session has been successfully deleted.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"delete",
"url":"sessions/neQXoKWYLALJ1yVjr5d2ANOhZrwp52p5qeyM7WdzAi96rJ5QKmGHsIyF5tzV",
"data":"company=testco"}

Reponse:
{"status":"0",
"data":true,
"message":"Success"}

Devices

This section handles devices.

Configuration

This section handles the devices configuration.

GET /devices

This method returns the list of devices visibles for the user.

Request parameters:

- **company [mandatory]**: the company name or id.

- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of processes and its descriptions.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Request:

{

"session": "QhuFhXB9MXprVKEfakhwaZxgA5osZg1o2NDaZeDunjRBX7cETnq6fehDH0K1",

"company": "demo04"

}

Response:

"id": "GRUPOA",

"ki": "",

"supervise": "X",

"hidehelp": "",

"adminexits": "X",

"demo": "",

"nokeyb": "1",

"area": "L1",

"process": "A0L1,A0L2",

"autologout": "480",

"logdetail": "0",

"defratio": "0",

"fdow": "0",

"datatype": "1",

"datacust": "ProdTheor; Produccin Teorica ProdBad; Scrap ProdGood; Produccion Real TimeNotProdMin; Tiempo no Prod.(min) TimeProdMin; Tiempo Productivo(min) TimeMin; Tiempo Total (min) OEE; % OEE OEEE; % Eficiencia SpeedT; Velocidad Terica SpeedAvg; Velocidad Med.",

"allowmin": "",

```
"resize": "0",
"login": "1",
"mac": "",
"url": "",
"vistype": "0",
"scrwait": "30",
"scrpresent": "10",
"preselectteam": "",
"theme": "0",
"periodtoshow": "1"
},
"message": "Success"
}
```


POST /devices

This method creates a device. The user must have the administrator role.

Request parameters (for additional information please review the devices configuration):

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **id [mandatory]**: the id of the device.
- **ki**: indicates with 1 (deprecated) if the device works off-line or on-line (0, recommended).
- **supervise**: indicates if the device must be checked by the supervisor service.
- **nokeyb**: indicates with 1 if the device has a touch screen and therefore a soft keyboard will be shown.
- **adminexits**: indicates with 1 if only the administrators can exit the application.
- **autologout**: indicates after how many minutes will the terminal auto log-out when there is no activity by the user. Set 0 to ignore.
- **defratio**: indicates the default ratio to be shown on the main form of the terminal.

- **hidehelp**: indicates with 1 if the contextual help icon must be hidden in the terminal.
- **fdow**: indicates at what day the week starts.
- **logdetail**: (not in use).
- **datatype**: type of data seen at "Data" at the main window. Options: standard, particular. At the particular option, particularization sequence should be introduced in this field. To get to know the particularization options please consult the *Particularization of data grid* on the Advanced Guide of this manual.
- **datacust**: contains the tags to select the data to be shown in the custom data panel.
- **demo**: (not in use).
- **allowmin**: indicates with 1 if the terminal can be minimized to the Windows bar.
- **resize**: indicates the size of the application on the Windows screen.
- **login**: indicates the type of login for the operation terminal. Put 0 for the user and password login form, and 1 for the easy login (select user and type your PIN).
- **mac**: indicate the MAC address of the device.
- **url**: a URL, following the URL syntax of the system, that can be launched by the user from the login window.
- **vistype**: indicates the default panel that will be shown in the main window.
- **scrwait**: indicates the time in seconds of inactivity of the user that the operation terminal should wait, in the login screen, to activate the edinn screen saver.
- **scrpresent**: indicates the time in seconds for the edinn screen saver to show the next process of the list of processes managed by that device.
- **preselecteam**: indicates if the team of the user should be preselected in the team drop down lists.
- **theme**: indicates the color theme.
- **area**: working areas to which the device (terminal) is associated. The terminal will only be able to see the areas indicated here.
- **process**: processes to which the device (terminal) is associated. The terminal will only be able to see the processes indicated here.
- **period:** indicates the period of time to show for the device.

Example:

```
Request:
{
 "session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
 "company": "demo01",
 "id": "DeviceId"
}
Response:
{
"status": 0,
"data": null,
"message": "Success"
}
```

DELETE/devices/{id}

This method deletes a device. The user must have the administrator role.

Request attribute:

- **id**: the Id of the device to delete.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of processes and its descriptions.
- **message**: "Success" or a descriptive information of the error in any other case.

```
Request:
{
/devices/D0001
}
```

```
{
 "session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
 "company": "demo01"
}

Response:
{
"status": 0,
"data": null,
"message": "Success"
}
```


Recourses

This section handles resources. Please notice that we use the word 'recourses' instead of 'resources' as this last is a reserved word in URLs and it might generate confusion.

Configuration

This section handles the resources configuration. Please notice that this was changed to 'recourses' instead of 'resources' as this last is a reserved word in URLs.

GET /recourses/config/{id}

This method returns the list of resources for the user.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **id:** The Id of the recourse.

Response:

- **status**: returns "0" on success, and an error code in any other case.

- **data**: returns the list of resources and its descriptions.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

```
Request:
{
    "session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
    "company": "demo01"
}

Response:
{
    "status": "0",
    "data": {
        "id": "USER",
        "name": "",
        "lastname": "",
        "language": "en",
        "email01": "user@edinn.com",
        "email02": "",
        "code": "",
        "tooltips": "0",
        "y0": "0",
        "g0": "0",
        "areas": "A0001",
        "processes": "P0001",
        "process": "",
        "roles": "548682072063",
        "password": "03ac674216f3e15c761ee1a5e255f067953623c8b388b4459e13f978d7c846f4",
        "fields_production": "",
        "fields_autocontrol": "",
        "user": "",
        "expire": "",
        "scheduleatstart": "0",
        "alertsweekdays": "0",
        "alertstype": "0",
        "alertsat": "",
        "alertssent": "",
        "sso": "",
        "maxunits": "0,00",
        "decimalsymbol": ",",
        "schedulefields": "0",
        "orderschefields": "0",
        "notify":"1",
        "maximun_time":"0"
    },
    "message": "Success"
}
```

POST /recourses/config

This method creates a resource. The user must have the administrator role.

Request parameters (for additional information please review the resources configuration):

- **company [mandatory]**: the company name or id.

- **session [mandatory]**: the Id of the active session.
- **id [mandatory]**: the id of the resource.
- **pass [mandatory]**: the password of the resource.
- **name**: name of the person.
- **lastname**: last name of the person
- **language**: to configure the system when the user connects to the system.
- **email01**: used to:
    - receive notifications or comments that are sent through the comments system.
    - list of the autocontrol tasks that are about to expire.
    - indications of what messages, in the communication flow between edinn and other systems (ERP, etc.), have failed and require manual intervention.
- **email02**: used to:
    - receive notifications of not done autocontrol tasks.
    - receive notifications of processes that are stopped more than a certain time.
- **code**: code for quick access from the login window. This is useful for card readers, so that the user just passes the card and automatically logs in the system.
- **tooltips**: if 1, help tags will appear on screen to the user next time he logs in.
- **y0**: yellow target of activity.
- **g0**: green target of activity.
- **areas**: if the user is restricted to access only the process of certain areas. Indicate them separated by commas: AREA1,AREA2. The user will only be able to see the areas indicated here. Leave blank to see all areas.
- **processes**: if the user is restricted to access only these processes. Indicate them separated by commas: PROC1,PROC2. The user will only be able to see the processes indicated here. Leave blank to see al processes.
- **associated process**: this is useful when a process is a person. With this field the selected person will be associated with that particular process.
- **roles (**permissions or security): please see the available roles in resources configuration. You should check if the bit of the corresponding role number

(substracting 1 as they start in the bit zero) is active in this number.

- **expire:** after this date, the person will not be able to access the system, will not be shown in the resources (persons) list at the login page nor will be able to access the reports tool.
- **scheduleatstart**: what should happen when a new work order is started, to pause, to prefinish or to finish the work order which is being processed in that moment.
- **fields_production:** indicates the customized product fields that will be visible for the user.
- **fields_autocontrol:** indicates the customized autocontrol tasks fields that will be visible for the user.
- **alertsweekdays**: the days that the user will receive alerts.
- **alertstype**: the type alerts the user will receive.
- **alertsat**:the time at that the user will receive alerts.
- **alertssent**: indicates if the user will receive alerts.
- **sso**: single sing on, enables the user to autoload when starting the session in the computer.
- **maxunits**: the maximun number of recourses that can be used at the same time.
- **decimalsymbol**: indicates which symbol will be used for this resource to indicate decimals.
- **schedulefields**: is a combination of bits to indicate which columns of the data grid in the Schedule window are visible for this resource.
- **orderschefields**: if different of zero the system will allow this resource (user) to sort the data table in the Schedule window by any visible column. Otherwise, the table will be automatically sorted by the WBS column.
- **notify**: if 0 the user will not be notified about changes in the status of work orders.
- **template_proc:** template process. If indicated, it makes a copy of the indicated process and establishes it as the process associated with the person. It is useful in work management.
- **maximun_time:** Maximum uninterrupted time in seconds that the resource can be IN in a process. After this time, the system will automatically OUT. If left at zero, it will not be applied.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "success" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.

```
Request:
{
    "session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
    "company": "demo01",
    "id": "USER",
    "pass": "1234"
}

Response:
{
    "status": 0,
    "data": null,
    "message": "Success"
}
```

POST /recourses/config (ISA-95 compliant, for more information please read the B2MML document)

This method creates a new resources using json encoded B2MML data. The user must have the administrator role.

Request parameters (for additional information please review the resources configuration):

- **company [mandatory]**: The company name or id.
- **session [mandatory]**: The Id of the active session.
- **data [mandatory]**: The json encoded B2MML data defining the resource.

Response:

- **status**: Returns "0" on success, and an error code in any other case.
- **data**: the json encoded B2MML data with the response.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

```
{
   "SyncPerson":{
      "ApplicationArea":{
         "Sender":{
            "ComponentID":"ERP",
            "ConfirmationCode":"Always"
         },
         "CreationDateTime":"2012-10-23T16:30:32"
      },
      "DataArea":{
         "Sync":{
```

```
            "ActionCriteria":{"ActionExpression":{"@actionCode":"Add"}}
         },
         "Person":{
            "ID":"PERS1",
            "PersonName":"Name",
            "PersonProperty":[
               {"ID":"Roles","Value":
               {"ValueString":"11325423","DataType":"integer","UnitOfMeasure"
               :""}},
               {"ID":"Pass","Value":
               {"ValueString":"1234","DataType":"string","UnitOfMeasure":""}}
               ,
               {"ID":"Lang","Value":
               {"ValueString":"ES","DataType":"string","UnitOfMeasure":""}},
               {"ID":"Email01","Value":
               {"ValueString":"pers01@enterprise.com","DataType":"string","Un
               itOfMeasure":""}},
               {"ID":"Email02","Value":
               {"ValueString":"pers02@enterprise.com","DataType":"string","Un
               itOfMeasure":""}},
               {"ID":"Tooltips","Value":
               {"ValueString":"0","DataType":"integer","UnitOfMeasure":""}},
               {"ID":"Lines","Value":
               {"ValueString":"L1","DataType":"string","UnitOfMeasure":""}},
               {"ID":"LastName","Value":
               {"ValueString":"Doe","DataType":"string","UnitOfMeasure":""}},
               {"ID":"Code","Value":
               {"ValueString":"12345678","DataType":"string","UnitOfMeasure":
               ""}},
               {"ID":"Id_Proc","Value":
               {"ValueString":"WK1","DataType":"string","UnitOfMeasure":""}},
               {"ID":"Expire","Value":
               {"ValueString":"20130131235959","DataType":"string","UnitOfMea
               sure":""}},
               {"ID":"Processes","Value":
               {"ValueString":"WK1,WK2","DataType":"string","UnitOfMeasure":"
               "}},
               {"ID":"Name","Value":
               {"ValueString":"John","DataType":"string","UnitOfMeasure":""}}
            ],
            "Location":{
               "EquipmentID":"COMP1",
               "EquipmentElementLevel":"Enterprise",
               "Location":{
                  "EquipmentID":"P1",
                  "EquipmentElementLevel":"Site",
                  "Location":{
                     "EquipmentID":"PL1",
                     "EquipmentElementLevel":"ProductionLine"
                  }
               }
            }
         }
      }
   }
}
```


Response example:

```
{
   "status":"0",
   "data":"{
      \"ConfirmBOD\":{
      \"ApplicationArea\":{
         \"Sender\":{\"ComponentID\":\"EDINN\",\"ConfirmationCode\":
         \"Always\"},\"CreationDateTime\":\"2019-11-12T13:16:27\"
      },
      \"DataArea\":{
         \"Confirm\":{\"ResponseCriteria\":{\"ResponseExpression\":
         {\"@actionCode\":\"Accepted\"}}},
         \"BOD\":{\"Description\":\"SyncPerson\",\"Note\":\"Person PERS1
         inserted successfully\"}}
      }
   }",
   "message":"Success"
}
```


DELETE /recourses/config/{id}

This method deletes a resource. The user must have the administrator role.

Request attribute:

- **id**: the Id of the device to delete.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "success" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.


```
Request:
{
    "session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
    "company": "demo01"
}

Response:
{
    "status": 0,
    "data": null,
    "message": "Success"
}
```

Operación

This section addresses operations on resources.

GET /recourses/activity

This method provides the activity report.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **datefrom [mandatory]:** The starting date of the selection.
- **dateto [mandatory]:** The ending date of the selection.
- **recourse:** Id of the resource you want to get the report from.
- **area:** Id of the area for which you want to obtain the report.
- **process:** Id of the process for which you want to obtain the report.
- **ignoreteam:** Excludes information about the team to which the resource belongs.
- **inhours:** Returns times in hours instead of minutes, which is the default unit.

Response:

- Returns the activity report in json format.

Example:


```
Request:
{
"type":"get",
"url":"recourses/activity",
"data":"company=demo01&session=BldLi14J6RaUidpt15VNDEbjufjrertuhgrutiSHTHMXnO8bbcM4hNtGAHzii7&datefrom=20240311060000&dateto=20240312060000&area=1100"
}

Response:
{
"status":"0",
"data":
[
{"Person":"0111-SAM","Area":"1100","Process":"1110-MIZ","Result":"AF1400CM","Quantity":"0","Target":"0","TargetActivityPercentage":"51.3","ActivityPercentage":"0","Start":"11\/03\/2024 06:00:00","End":"12\/03\/2024 06:00:00","RealTime":"1440","TotalTime":"1440","AttendanceTime":"1440","ActivityTime":"0","MandatoryTime":"0","Difference":"0","Incentive":"0","TotalIncentive":"0","Currency":""},{"Person":" ","Area":" ","Process":" ","Result":"TOTAL","Quantity":"0","Target":"0","TargetActivityPercentage":"51.3","ActivityPercentage":"0","Start":"11\/03\/2024 06:00:00","End":"12\/03\/2024
```

06:00:00","RealTime":"1440","TotalTime":"1440","AttendanceTime":"1440","ActivityTime":"0","MandatoryTime":"0","Difference":" ","Incentive":"0","TotalIncentive":"0","Currency":""}
],
"message":"Success"
}

Units

This section handles measure units.

Configuration

This section handles the measure units configuration.

GET /units

This method returns the list of measures units.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of measures units.
- **message**: "Success" or a descriptive information of the error in any other case.

Request:

{
"session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
"company": "demo01"
}

Response:

```
{
"status": "0",
"data": [
        {
                "order": "00000",
                "name": "piezas"
        }
"message": "Success"
}
```


POST /units

This method creates a unit. The user must have the administrator role.

Request parameters (for additional information please review the meausre units configuration):

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **name [mandatory]**: the name of the measure unit.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "success" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.


```
Request:
{
 "session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
 "company": "demo01",
 "id": "UnitId"
}

Response:
```

```
{
"status": 0,
"data": null,
"message": "Success"
}
```

DELETE /units/{id}

This method deletes a unit. The user must have the administrator role.

Request attribute:

- **id [mandatory]**: the name of the unit to delete.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "success" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.

```
Request:
{
/units/UnitId
}

{
 "session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
 "company": "demo01"
}

Response:
{
```

```
"status": 0,
"data": null,
"message": "Success"
}
```


Calculations

This section handles calculations.

Configuration

This section handles the calculations configuration.

GET /calculations

This method returns the list of calculation types.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **process [mandatory]:** The id of the process.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of calculation types.
- **message**: "Success" or a descriptive information of the error in any other case.


```
Request:
{
 "session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
 "company": "demo01",
 "process": "P0001"
```

```
}


Response:

{

"status": "0",

"data": [

        {

                "name": "Euros",

                "variables": "Euros",

                "units": "Euros",

                "formula": ""Euros""

        }

"message": "Success"

}
```

POST /calculations

This method creates a calculation type. The user must have the administrator role.

Request parameters (for additional information please review the calculation types configuration):

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **name [mandatory]**: the name of the calculation type.
- **variables**: variables that will be requested in order to calculate the quantity. There are predefined variables in configuration.
- **units:** measurement units that will be used to express the result of the calculation.
- **formula:**mathematic function used to calculate the result. Variables must be enclosed with quotation marks, for example: "Width".

```
Request:
```

```
{
 "session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
 "company": "demo01",
 "name": "Calc"
}
Response:
{
"status": 0,
"data": true,
"message": "Success"
}
```

DELETE /calculations/{id}

This method deletes a calculation type. The user must have the administrator role.

Request attribute:

- **id [mandatory]**: the Id of the calculation type to delete.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "success" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.

```
Request:
{
/calculations/Calc
}
```

```
{
 "session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
 "company": "demo01"
}
```


Response:


```
{
"status": 0,
"data": null,
"message": "Success"
}
```


Statuses

This section handles statuses.

Configuration

This section handles the statuses configuration.

GET /statuses/config

This method returns the list of statuses.

Request parameters:

- **company [mandatory]**: The company name or id.
- **session [mandatory]**: The Id of the active session.
- **process [mandatory]**: The id of the process.
- **id**: The Id of the parent status. Default null.
- **level**: The level of status list. Default 1.

Response:

- **status**: Returns "0" on success, and an error code in any other case.
- **data**: Returns the list of statuses and its properties.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Content-Type: application/json

Request: /statuses/config?process=A0L1&session=gWCRY6y8mEZNCIloJyKGXPpttRJyrBU34CgRcPfuTeK1teJnHT2TbDrZ981i&company=DEMO01&id_tu=&tu_level=1

Reponse (success):


```
{
  "status": "0",
  "data": [
    {
      "Id": "0",
      "Order": "00000",
      "Level": "1",
      "Desc_de": "Produktion",
      "Desc_en": "Production",
      "Desc_es": "Producción",
      "Desc_it": "Produzione",
      "Desc_pt": "Produção",
      "Desc_sk": "Výroba",
      "Desc_zh": "生产",
      "MaxTime": "0",
      "Type": "0",
      "Next_Status": "",
      "ActPct": "0",
      "URL": "",
      "PStart": "0",
      "Week_Days": "0",
```

```
    "ScheTime": "",
    "Blocks": "0",
    "TypeDef": "0",
    "Class": "0",
    "ReqCom": "0",
    "ReqRes": "0"
  },
  ...
 ],
 "message": "Success"
}
```


POST /statuses/config

This method creates a new status. The user must have the administrator role.

Request parameters:

- **company [mandatory]:** The company name or id.
- **session [mandatory]**: The Id of the active session.
- **id [mandatory]**: The Id of the status.
- **order**: the relative order to be shown.
- **level**: statuses are organized in 3 levels. These levels allow to show to the user the status in up to 3 levels. The user selects one option, and enters into the second level, selects another option, and goes to the 3rd level where he has to finally select a status.
- **desc_de:** description of the status (in german).
- **desc_en:** description of the status (in english).
- **desc_es:** description of the status (in spanish).
- **desc_it:** description of the status (in italian).
- **desc_pt:** description of the status (in portuguese).
- **desc_sk:** description of the status (in slovak).
- **desc_zh:** description of the status (in chinese).

- **maxtime**: maximum time in minutes that a process can be in this status. If any value different from 0 is specified, the system will automatically close this status when it is open and the indicated minutes are surpassed.
- **type**: production, failure (the working user is waiting for the process), idle time (the process is waiting for the working user), dependence (the process is on hold because it depends on another process) and not scheduled (no work is scheduled because of legal stops, holidays, etc.). Please see "What is a status?" for more information.
- **next_status**: the id of the next status to automatically pass when the maximum minutes are surpassed.
- **actpct**: percentage of activity that is required from a working user when a process is in this status. This is a requirement configuration for the activity report.
- **url**: document or application that will be opened when the user clicks on the that appears after having clicked on the status, on the status window. Please see how to configure URLs.
- **pstart**: this mark indicates to the system to automatically generate a new status to indicate that a new work has started, every time a new result with 0 quantity is inserted, if at the process configuration the option Gen W. Start has been marked for the process. Only one status should be marked with this option.
- **weel_days**: status can be automatically scheduled by days of the week and time. This allows, for example, to close days or weeks with END OF PRODUCTION and start them up when needed.
- **schedtime**: time at which the status will automatically be inserted.
- **blocks**: if the status should block any other statuses. Options are: none (no blocking), stops (this status will block all not productive statuses coming later), production (this status will block all productive statuses coming later), all (this status will block all status coming later).
- **typedef**: indicates to the system that this status is the default for the type. Please see "What is a status?" for more information.
- **class**: allows to identify the type or class of the status, according to a general classification. This is useful to compare statuses from different plants with different configurations of statuse.
- **reqcom**: to use the status a mandatory comment will be required.
- **reqres**: to use the status it will check that no more time than Max Cycle Time of the process has passed since the last result of advance or since the current working order was started. This is useful to guarantee that users inform of results advance before, for example, indicating that they have finished the day.

DELETE /statuses/config/{id}

This method deletes a configured status. The user must have the administrator role.

Request attribute:

- **id [mandatory]**: the Id of the calculation type to delete.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "Success" on success, and empty in case of error.
- **message**: "Success" or a descriptive information of the error in any other case.

Operation

This section handles the statuses operation.

GET /statuses

Provides you a list of statuses. You have to provide a specific process and the start and end time in order to filter the statuses. The user must have the statuses role.

Request:

- **session [mandatory]**: Your session Id.
- **company [mandatory]**: The Id of the company.
- **process [mandatory]**: The Id of the process of which you want the statuses.
- **datefrom [mandatory]**: The starting date of the selection.
- **dateto [mandatory]**: The ending date of the selection.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "Success" on success, and empty in case of error.
  - id: Record identifier in the database.

- machdate: Indicates the process where the status was registered along with the start date and time of the period in the format yyyymmddhhmmss.
- id_tu: Status code.
- endtime: End date and time of the period in the format yyyymmddhhmmss. This field remains blank for the current status.
- secs: Duration of the period in seconds. It remains at 0 for the current status.
- id_author: Identifier of the resource that created the period. The value CENTRAL indicates that it was automatically created by Edinn.
- created: Indicates the creation date and time of the record in the database in the format yyyymmddhhmmss.
- id_mod: If the record was modified, it indicates the identifier of the resource that last modified it.
- modified: Indicates the date and time of the last modification of the record in the database in the format yyyymmddhhmmss.
- link: Identifier of the comment table record associated with this status record.
- notified: Date and time when an email notification was sent about this status in the format yyyymmddhhmmss.
- tu_desc: Status description in the language of the resource that made the request.
- tu_type: Type of status. The status types are described here.

- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"get",
"url":"statuses",
"data":"company=testco&session=LzRlv4vjbyHWPhO2B2ulNjmSZFNmvpW4aqbLaoz1n6KJF91RJPCsZcmVLzpy&process=7101&datefrom=20211008060000&dateto=20211008090000"}

Response (success):
{"status":"0",
"data":
[{"id":"20211008093900AARCENTRAL","machdate":"710120211008060000","id_tu":"SNP","endtime":"20211008085530","secs":"10530","id_author":"CENTRAL","created":"20211008093900","id_mod":"ADMIN","modified":"20211008101419","link":"","notified":"","tu_desc":"SIN PRODUCCION","tu_type":"4"},
{"id":"20211008101419AAGINIT","machdate":"710120211008085530","id_tu":"6105","endtime":"20211008095512","secs":"3582","id_author":"ADMIN","created":"20211008101419","id_mod":"ADMIN","modified":"20211008101432","link":"","notified":"","tu_desc":"Cambio de pedido o modelo","tu_type":"1"}],

"message":"Success"}

POST /statuses

Create a new status. You have to provide a specific process and the start and end time in order to filter the statuses. The user must have the statuses role.

Request:

- **company [mandatory]**: The Id of the company.
- **session [mandatory]**: Your session Id.
- **process [mandatory]**: The Id of the process of which you want the statuses.
- **device [mandatory]:** The Id of the device that the status pertains to.
- **status [mandatory]:** The name of the new status.
- **datefrom [mandatory]**: The date of the new status.
- **dateend:** If the status is planned, indicates the finish date of it.
- **calendar:** Indicates if the status is planned or not.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: status identifier code on success, and empty in case of error.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"post",
"url":"statuses",
"data":"company=testco&session=0bvqC8zNdyTkm5BrqX7M0C80IC3O2MraAz0RxKyhnenmNfNVTL0aH5dpMJO6&process=7102&datefrom20211011160000&dateto=20211011163000&status=6105&calendar=1&device=init"}

Response (success):
{"status":"0","data":"20211011153821AABINIT","message":"Success"}

DELETE /statuses/{id}

This method deletes a status. View this document to know the restrictions that apply.

Request attribute:

- **id [mandatory]**: the Id of the status.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "success" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"delete",
"url":"statuses/20211011153821AABINIT",
"data":"company=testco&session=0bvqC8zNdyTkm5BrqX7M0C80IC3O2MraAz0RxKyhnenmNfNVTL0aH5dpMJO6&calendar=1"}

Response (success):
{"status":"0","data":true,"message":"Success"}

DELETE /statuses

This method deletes a status. View this document to know the restrictions that apply.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **datefrom [mandatory]**: the start date of the period
- **dateto [mandatory]:** the finish date of the period.
- **calendar [mandatory]:** Indicates if the status is planned or not.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "success" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"delete",
"url":"statuses",
"data":"company=testco&session=0bvqC8zNdyTkm5BrqX7M0C80IC3O2MraAz0RxKyhnenmNfNVTL0aH5dpMJO6&process=7102&datefrom=20211011160000&dateto=20211011163000&calendar=1"}

Response (success):

{"status":"0","data":true,"message":"Success"}

PATCH /statuses/{id}

This method justifies a status.

Request attribute:

- **id [mandatory]**: the Id of the status.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **date [mandatory]**: The starting date of the selection.
- **status [mandatory]**: the id of the status to justify.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "success" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"patch",
"url":"statuses/20211011141555ACSCENTRAL",
"data":"company=testco&session=0bvqC8zNdyTkm5BrqX7M0C80IC3O2MraAz0RxKyhnenmNfNVTL0aH5dpMJO6&date=20211011141500&status=6104"}

Response (success):
{"status":"0","data":true,"message":"Success"}

PUT /statuses/{id}

This method justifies or divides a status based on the **type** of the parameter passed. (**0** to **Justify, 1** to **Divide**)

Request attribute:

- **id [mandatory]**: the Id of the status.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **type[mandatory]**: Type of PUT operation (**0** to **Justify, 1** to **Divide.**)

- **status**: the id of the status to justify. (**mandatory** in order to **Justify** a status**).**
- **date**: The starting date of the selection (**mandatory** in order to **Divide** a status**).**
- **device:** The Id of the device (**mandatory** in order to **Divide** a status**).**

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "true" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.

Example (**Divide**):

Request:
{"type":"put",
"url":"statuses/20090301070004TBBCENTRAL",
"data":"company=testco&session=0bvqC8zNdyTkm5BrqX7M0C80IC3O2MraAz0RxKyhnenmNfNVTL0aH5dpMJO6&date20211011134506=FAI&type=1&device=init"}

Response (success):
{"status":"0","data":true,"message":"Success"}

Example (**Justify**):

Request:
{"type":"put",
"url":"statuses/20090301070004TBBCENTRAL",
"data":"company=testco&session=0bvqC8zNdyTkm5BrqX7M0C80IC3O2MraAz0RxKyhnenmNfNVTL0aH5dpMJO6&status=FAI&type=0"}

Response (success):
{"status":"0","data":true,"message":"Success"}

GET /statuses/fai

This method returns information related to the failure(s) of a process.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **process [mandatory]:** the id of the process

Response:

- **status**: returns "0" on success, "01" in the case of there being no failures, and an error code in any other case.
- **data**: Returns the data pertaining to the failures of the selected process, or in the case of no failures, a link to help info.

- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"get",
"url":"statuses/fai",
"data":"company=testco&session=0bvqC8zNdyTkm5BrqX7M0C80IC3O2MraAz0RxKyhnenmNfNVTL0aH5dpMJO6&process=7102"}

Response (success):
{"status":"0","data":
[{"id":"20211011134701AAKCENTRAL","machdate":"710220211011060000","id_tu":"FAI","end":"","secs":"0","Author":"CENTRAL","Created":"20211011134701","Modifier":"","Modified":"","Link":"3","Notified":"","type":"PENDING"},
{"id":"20211011134659AAICENTRAL","machdate":"710220211011134500","id_tu":"6531","end":"","secs":"0","Author":"CENTRAL","Created":"20211011134659","Modifier":"","Modified":"","Link":"4","Notified":"","type":"Final turno ma\u00f1ana"}],"message":"Success"}

GET /statuses/summary

Provides you a list of statuses. You have to provide a specific process and the stant and end time in order to filter the statuses.

Request:

- **session [mandatory]**: Your session Id.
- **company [mandatory]**: The Id of the company.
- **process [mandatory]**: The Id of the process of which you want the statuses.
- **datefrom [mandatory]**: The starting date of the selection.
- **dateto [mandatory]**: The ending date of the selection.
- **top**: The maximum number of statuses shown. Default value is 10.
- **Type**: The type of status.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "Success" on success, and empty in case of error.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"get",
"url":"statuses/summary",
"data":"company=testco&session=0bvqC8zNdyTkm5BrqX7M0C80IC3O2MraAz0RxKyhnenmNfNVTL0aH5dpMJO6&process=7101&datefrom=20211008060000&dateto=20211009060000&top=3"}

Response (success):
{"status":"0","data":[{"id":"6102","start":"","time":"51375","quantity":"1","desc_de":"Cambio de bobina","desc_en":"Cambio de bobina","desc_es":"Cambio de bobina","desc_it":"Cambio de bobina","desc_pt":"Cambio de bobina","desc_sk":"","desc_zh":""},{"id":"0","start":"","time":"12759","quantity":"4","desc_de":"PRODUCCION","desc_en":"PRODUCCION","desc_es":"PRODUCCION","desc_it":"PRODUCCION","desc_pt":"PRODUCCION","desc_sk":"","desc_zh":""},{"id":"SNP","start":"","time":"11094","quantity":"2","desc_de":"SIN PRODUCCION","desc_en":"SIN PRODUCCION","desc_es":"SIN PRODUCCION","desc_it":"SIN PRODUCCION","desc_pt":"SIN PRODUCCION","desc_sk":"","desc_zh":""}],"message":"Success"}

GET /statuses/top

This method returns the Id and description of the top status of the selected process.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **Process [mandatory]:** the id of the process

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: Id and description of the status
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"get",
"url":"statuses/top",
"data":"company=testco&session=0bvqC8zNdyTkm5BrqX7M0C80IC3O2MraAz0RxKyhnenmNfNVTL0aH5dpMJO6&process=7101"}

Response (success):
{"status":"0",
"data":[{"Id":"6105","Desc":"Cambio de pedido o modelo"},{"Id":"6102","Desc":"Cambio de bobina"},{"Id":"6301","Desc":"Limpieza"}],
"message":"Success"}

Results

This section handles the results produced by a process.

Configuration

This section handles the results configuration.

GET /results/config

GET /results/config/{result}

This method returns the list of results.

Request attribute:

- **result:** Id of the result from which we want to obtain the information. If not indicated, all are shown.

Request parameters:

- **company [mandatory]**: The company name or id.
- **session [mandatory]**: The Id of the active session.
- **type:** result type to filter by (0 good, 1 rework, 2 scrap, 255 all). The default values is 255.
- **showfields**: indicates if the associated fields and their default values are shown (0 not shown, 1 if shown). The dafault value is 0.

Response:

- **status**: Returns "0" on success, and an error code in any other case.
- **data**: Returns the list of results and its properties.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

```
Content-Type: application/json
Request: /results/config?
company=demo01&session=LebEWTNf8GEhxfQeeGbBmVUlxmUIDLVdEoeVYrnWNzkdRXKxmgCNc9
2wnSjA
Reponse (success):
{
  "status": "0",
  "data": [
    {
      "id": "8561",
      "desc_de": "8561 1.4L",
      "desc_en": "8561 1.4L",
      "desc_es": "8561 1.4L",
      "desc_it": "8561 1.4L",
      "desc_pt": "8561 1.4L",
      "desc_sk": "8561 1.4L",
      "desc_zh": "8561 1.4L",
      "type": "0",
      "equiv": "1",
      "weight": "1",
      "url": "\\Servervlc01\SCalidad\Instruc\Instr.
```

```
Inspeccion\8561\8561ii040-r07.doc = Inst. Inspección",
      "group": "0",
      "y0": "0",
      "g0": "0",
      "y1": "0",
      "g1": "0",
      "y2": "0",
      "g2": "0",
      "y3": "0",
      "g3": "0",
      "y4": "0",
      "g4": "0",
      "y5": "0",
      "g5": "0",
      "y6": "0",
      "g6": "0",
      "y7": "0",
      "g7": "0",
      "in_percentage": "0",
      "ignore_integration": "0",
      "replacement": "30",
      "expiration": "90",
      "minimum_stock": "300",
      "recurring": "0"
    },
    ...
  ],
  "message": "Success"
}
```


POST /results/config

This method creates a new result. The user must have the administrator role.

Request parameters (for additional information please review the result configuration):

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **id [mandatory]**: the id of the result.
- **desc_de:** description of the result (in german).
- **desc_en:** description of the result (in english).
- **desc_es:** description of the result (in spanish).
- **desc_it:** description of the result (in italian).
- **desc_pt:** description of the result (in portuguese).
- **desc_sk:** description of the result (in slovak).
- **desc_zh:** description of the result (in chinese).

- **type:** production, rework or scrap.
- **equiv:** every production counting is equivalent to how many pieces.
- **weight:** weight of each piece (equiv.).
- **url:** document or application that will be opened when the user clicks on the folder icon that appears after having clicked on the result, on the results window. Please see how to configure URLs.
- **group:** if this result will be used to group results.
- **y0**: yellow objective for availability.
- **g0**: green objective for availability.
- **y1**: yellow objective for speed.
- **g1**: green objective for speed.
- **y2**: yellow objective for quality.
- **g2**: green objective for quality
- **y3**: yellow objective for productive availability.
- **g3**: green objective for productive availability.
- **y4**: yellow objective for the quantity of MTBF.
- **g4**: green objective for the quantity of MTBF.
- **y5**: yellow objective for the MTBF.
- **g5**: green objective for the MTBF.
- **y6**: yellow objective for the MTTR.
- **g6**: green objective for the MTTR.
- **y7**: yellow objective for the OCE.
- **g7**: green objective for the OCE.
- **inpct:** if production is measured in %. Useful for results with a large cycle time (days or months) and to allow the working user to provide periodical advance of his work.
- **interign:** marks the result as to be ignored when transferring from and to information to another system, like ERPs. Please see integration with other systems.
- **replacement:** time it takes for the material to arrive since the order is placed (in days).

- **expiration:** time in which a material expires since it is manufactured (in days).
- **minstock:** minimum stock below which a new order must be placed.
- **recurring**: indicates if this result will be recurring. Task or work orders which are marked as recurring can only generate recurring results.
- **replace**: indicates if the result with the same id is going to be replaced with the new one. Defaul value is zero (no replacement).

POST /results/config (ISA-95 compliant, for more information please read the B2MML document)

This method creates a new result using json encoded B2MML data. The user must have the administrator role.

Request parameters (for additional information please review the result configuration):

- **company [mandatory]**: The company name or id.
- **session [mandatory]**: The Id of the active session.
- **data [mandatory]**: The json encoded B2MML data defining the result.

Response:

- **status**: Returns "0" on success, and an error code in any other case.
- **data**: the json encoded B2MML data with the response.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

```
{
"SyncProductDefinition": {
    "ApplicationArea": {
      "Sender": {
        "ComponentID": "ERP",
        "ConfirmationCode": "Always"
      },
      "CreationDateTime": "2017-02-10T07:30:03"
    },
    "DataArea": {
      "Sync": {
        "ActionCriteria": {
          "ActionExpression": { "@actionCode": "Add" }
        }
      },
      "ProductDefinition": [
        {
          "ID": "PROD001",
          "Description": "Product Description",
          "Location": {
            "EquipmentID": "COMPANY",
            "EquipmentElementLevel": "Enterprise"
```

```
        },
        "ProductSegment": {
          "EquipmentSpecification": {
            "EquipmentID": "PROC001",
            "EquipmentSpecificationProperty": [
              {
                "ID": "Equivalence",
                "Quantity": {
                  "QuantityString": "1",
                  "DataType": "FLOAT",
                  "UnitOfMeasure": "UN"
                }
              },
              {
                "ID": "CycleT",
                "Quantity": {
                  "QuantityString": "10",
                  "DataType": "FLOAT",
                  "UnitOfMeasure": "SEC"
                }
              },
              {
                "ID": "CycleQ",
                "Quantity": {
                  "QuantityString": "1",
                  "DataType": "FLOAT",
                  "UnitOfMeasure": "UN"
                }
              }
            ]
          }
        },
        "Any": {
          "ProductType": "0",
          "Equivalence": {
            "QuantityString": "1",
            "DataType": "FLOAT",
            "UnitOfMeasure": "UN"
          },
          "Weight": {
            "QuantityString": "1",
            "DataType": "FLOAT",
            "UnitOfMeasure": "KG"
          },
          "DocumentURL": "\\\\DOCS_FOLDER\\DOC1.INF",
          "Fields": {
            "Field": {
              "ID": "FIELD",
              "Description": "Field_Description",
              "DataType": "STRING",
              "Mandatory": "Ifempty",
              "Context": "Input",
              "StandardClass": "Lot",
              "MaxLength": "20",
              "Calculated": "0",
              "Locked": "0",
              "Inherits": "1",
              "Stock": "1"
            }
```

```
        }
      }
    },
    {
      "ID": "PROD001.S1",
      "Description": "Scrap_Description",
      "Location": {
        "EquipmentID": "COMPANY",
        "EquipmentElementLevel": "Enterprise"
      },
      "ProductSegment": {
        "EquipmentSpecification": {
          "EquipmentID": "PROC001",
          "EquipmentSpecificationProperty": [
            {
              "ID": "Equivalence",
              "Quantity": {
                "QuantityString": "1",
                "DataType": "FLOAT",
                "UnitOfMeasure": "UN"
              }
            },
            {
              "ID": "CycleT",
              "Quantity": {
                "QuantityString": "10",
                "DataType": "FLOAT",
                "UnitOfMeasure": "SEC"
              }
            },
            {
              "ID": "CycleQ",
              "Quantity": {
                "QuantityString": "1",
                "DataType": "FLOAT",
                "UnitOfMeasure": "UN"
              }
            }
          ]
        },
        "MaterialSpecification": {
          "MaterialDefinitionID": "PROD001",
          "MaterialUse": "Consumed",
          "Quantity": {
            "QuantityString": "1",
            "DataType": "FLOAT",
            "UnitOfMeasure": "UN"
          }
        }
      },
      "Any": {
        "ProductType": "0",
        "Equivalence": {
          "QuantityString": "1",
          "DataType": "FLOAT",
          "UnitOfMeasure": "UN"
        },
        "Weight": {
          "QuantityString": "1",
```

```
            "DataType": "FLOAT",
            "UnitOfMeasure": "KG"
          },
          "DocumentURL": "\\\\DOCS_FOLDER\\DOC2.INF",
          "Fields": {
            "Field": {
              "ID": "FIELD",
              "Description": "Field_Description",
              "DataType": "STRING",
              "Mandatory": "Ifempty",
              "Context": "Input",
              "StandardClass": "Lot",
              "MaxLength": "20",
              "Calculated": "0",
              "Locked": "0",
              "Inherits": "1",
              "Stock": "1"
            }
          }
        }
      }
    ]
  }
 }
}
```

Notes:

B2MML tags not present in this document will be ignored by the system, althought could be considered for future versions.

Any: Refers to result properties not found in the standard:

- ProductType: production, rework or scrap.
- Equivalence: every production counting is equivalent to how many pieces.
- Weight: weight of each piece (equiv.).
- Fields: refers to the list of custom properties added by the user:
  - ID, Description, DataType, Mandatory, Context, StandardClass, MaxLength, Calculated, Locked, Inherits and Stock are described in fields configuration section.

Response exmaple:

```
{
   "status":"0",
   "data":"{
      \"ConfirmBOD\":{
         \"ApplicationArea\":{
         \"Sender\":{\"ComponentID\":\"EDINN\",\"ConfirmationCode\":
         \"Always\"},
         \"CreationDateTime\":\"2019-11-12T11:16:57\"
         },
         \"DataArea\":{
            \"Confirm\":{
```

```
                \"ResponseCriteria\":{\"ResponseExpression\":{\"@actionCode\":
                \"Accepted\"}}
             },
             \"BOD\":[
                {\"Description\":\"Accepted PROD001\",\"Note\":\"Product
                PROD001 updated successfully\"},
                {\"Description\":\"Accepted PROD001.S1\",\"Note\":\"Product
                PROD001.S1 updated successfully\"}
             ]
          }
       }
    }",
    "message":"Success"
}
```


DELETE /results/config/{id}

This method deletes a result. The user must have the administrator role.

Request attribute:

- **id [mandatory]**: the Id of the result to delete.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "success" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.

GET /results/config/{result}/relatedresults

This method returns the relationship between the result and its rework or scrap.

Request attribute:

- **result [mandatory]**: Id of the result whose related results we want to get.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **result [mandatory]**: the Id of the active result.
- **type [mandatory]**: the index of product type.
  - type = 0: Production

- type = 1: Scrap
- type = 2: Rework

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of results.
- **message**: "success" or a descriptive information of the error in any other case.

Request:
{"type":"get",
"url":"results/config/res0001/relatedresults",
"data":"company=testco&session=neQXoKWYLALJ1yVjr5d2ANOhZrwp52p5qeyM7WdzAi96rJ5QKmGHslyF5tzV&type=1"}

Reponse (success):
{
"status":"0",
"data":[

{"Id_prod":"res001.S1","Desc":"Setup"},
{"Id_prod":"res001.S2","Desc":"Operator"},
{"Id_prod":"res001.S3","Desc":"Raw material"}
],
"message":"Success"
}

Reponse (error):
{"status":"1005",
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Results_Config_Fields.html'>for additional information follow this link<\/a>",
"message":"No data available"}

POST /results/config/{result}/relatedresults

This method allows you to create the relationship between two results. If the relationship already exists and is of the same type, it updates the data. If it does not exist or is of a different type, it adds it.

Request attribute:

- **result [mandatory]:** Id of the result to which we want to add a related result.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **relatedresult [mandatory]**: Result we want to relate.
- **relationtype [mandatory]**: Relationship type.

        - 1: Input. Each time a result occurs, an amount of the related result will be consumed.
        - 2: Output. Each time a result is produced, an amount of the related result will be produced.
    - **quantity**: Amount of the related result to be consumed or produced each time a result is produced. Default is 1.
    - **confirm**: If set to 1, it asks for confirmation when the related result is to be consumed or produced, and allows the related result to be changed.
    - **produced**: If it is set to 1, it indicates that the result has been produced internally and has not been acquired.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns "true" on success or the link to the request help on error.
- **message**: "success" or a descriptive information of the error in any other case.

Request:
{"type":"post",
"url":"results/config/res0001/relatedresults",
"data":"company=testco&session=neQXoKWYLALJ1yVjr5d2ANOhZrwp52p5qeyM7WdzAi96rJ5QKmGHsIyF5tzV&relatedresult=res0002&relationtype=1&quantity=20"}

Reponse (success):
{
"status":"0",
"data":"true",
"message":"Success"
}

Reponse (error):
{"status":1074,
"data":"<a href='http:\/\/mayen19\/edinnM2\/help\/en\/API_Results_Config_Relations.html'>for additional information follow this link<\/a>",
"message":"Result not found"}

GET /results/config/{result}/fields

The method returns the field definition related with a result.

Request attribute:

- **result [mandatory]**: Id of the result whose fields we want to get.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the the list of fields belongin to the result on success or the link to the request help on error.
- **message**: "Success" or a descriptive information of the error in any other case.

Request:

{"type":"get",
"url":"results/config/res0001/fields",
"data":"company=testco&session=neQXoKWYLALJ1yVjr5d2ANOhZrwp52p5qeyM7WdzAi96rJ5QKmGHslyF5tzV"}

Reponse (success):


```
{
    "status":"0",
    "data":[
        {
            "id":"20200520113706AACCENTRAL",
            "parentid":"res0001",
            "order":"0",
            "name":"COLOR",
            "description":"NumerodeColores",
            "default":"4",
            "datatype":"4",
            "mandatory":"2",
            "fielddomainid":"",
            "context":"0",
            "calculation":"",
            "standardclass":"0",
            "maxlength":"0",
            "calculated":"0",
            "locked":"0",
            "inherits":"0",
            "stock":"0",
            "recurring":"0"
        }
    ],
    "message":"Success"
}
```


Reponse (error):

{"status":"1005",
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Results_Config_Fields.html'>for additional information follow this link<\/a>",
"message":"No data available"}

Operation

This section handles the results operation.

GET /results

This method returns a list of results. You have to provide a specific process and the stant and end time in order to filter the results. The user must have the results role.

Request parameters:

- **company [mandatory]**: The company name or id.
- **session [mandatory]**: The Id of the active session.
- **process [mandatory]**: The Id of the process of which you want the results.
- **dateFrom [mandatory]**: The starting date of the selection.
- **dateTo [mandatory]**: The ending date of the selection.

Response:

- Returns the list of the results in json format.

Example:

Request:
/results
{
"session": LzRlv4vjbyHWPhO2B2ulNjmSZFNmvpW4aqbLaoz1n6KJF91RJPCsZcmVLzpy
"company": DEMO04
“Process”: A0L1
“Datefrom”: 20181010000000
“Dateto”: 20191110000000
}
Response:
{
"status": 0,
Data: {
"Id": "20181115173804AABINIT",
"MachDate": "A0L120181115173804",
"PN": "",
"id_prod": "8561",
"QA": "1",
"QB": "0",
"QC": "0",
"QT": "100%",
"Id_Author": "MIGUEL",
"Created": "20181115173804",
"Id_Mod": "",
"Modified": "",
"Link": "",
"Paused": "0",
"Id_Paused": "",
"Id_Sync": "",
"ProdType": "0",
"Desc": "8561 1.4L"

"message":" Success"
}

POST /results

Allows you the possibility of adding a new result. The user must have the results role.

Request:

- **company [mandatory]**: The company name or id.
- **session [mandatory]**: The Id of the active session.
- **process [mandatory]**: The Id of the process.
- **result [mandatory]**: The Id of the result.
- **date**: The date on which you want to insert the result. Uses current date and time by default.
- **order**: The Id of the order.
- **quantityA:** quantity for the 1st variable.
- **quantityB**: quantity for the 1st variable.
- **quantityC**: quantity for the 1st variable.
- **device**: The Id of the device.

Response:

- Returns TRUE if the result has been successfully inserted.

Example:

Request:
/results
{
"session": LzRlv4vjbyHWPhO2B2ulNjmSZFNmvpW4aqbLaoz1n6KJF91RJPCsZcmVLzpy
"company": DEMO04
“Process”: A0L1
"result": 123456
}
Response:
{
"status": 0,
"data": true,
"message": "Success"
}

DELETE /results/{id}

This method deletes a results. The user must have the administrator or supervisor role.

Request attribute:

- **id [mandatory]**: the Id of the status.

Request parameters:

- **company [mandatory]**: the company name or id.

- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "true" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

```
Request:
/results/123456
{
 "session": 01qgmfcZmZ4nqg2eNGpxY4JlpmhJuKw1H31p5rtD1a112SSK8S8odROFIKgn
 "company": DEMO04
}
Response:
{
"status": 0,
"data": true,
"message": "Success"
}
```

GET /results/summary

This method returns a list of results. You have to provide a specific process and the start and end time in order to filter the results.

Request parameters:

- **company [mandatory]**: The company name or id.
- **session [mandatory]**: The Id of the active session.
- **process [mandatory]**: The Id of the process of which you want the results.
- **area [mandatory]**: The Id of the area.
- **dateFrom [mandatory]**: The starting date of the selection.
- **dateTo [mandatory]**: The ending date of the selection.
- **showio**: Shows the inputs and outputs of the selection.
- **type**: The index of result type.
  - type = 0: Production
  - type = 1: Scrap
  - type = 2: Rework

Response:

- **status**: returns "0" on success, and an error code in any other case.

- **data**: returns the amount of results produced grouped by result and filtered by the criteria in the request, on success or the link to the request help on error.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"get",
"url":"results/summary",
"data":"company=testco&session=1FrqhchVXQoAgpxUbdKHWQkILZjMMAV0cesiSMmaRjwFvaWLwrpLyxJ6O9Zo&area=7200&process=7202&dateFrom=20200910060000&dateTo=20200910140000&showio=0"}

Reponse (success):
{"status":"0",
"data":[{"Area":"7200-NAVE2","Process":"7202-RS 7202","Result":"A01-Botella100mlCiclopiroxOlamineGerda*","Type":"Results","Quantity":"718852","Units":"Ml","ActualKg":"718852","Target":"89.85","TargetKg":"89.85","ProductionOrder":"","Team":""},
{"Area":"","Process":"","Result":"","Type":"TOTALES","Quantity":"718852","Units":"Ml","ActualKg":"718852","Target":"89.85","TargetKg":"89.85","ProductionOrder":"","Team":""}],
"message":"Success"}

Reponse (error):
{"status":"1012",
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_.html'>for additional information follow this link<\/a>",
"message":"Process not found"}

Processes

This section handles processes.

Configuration

This section handles the processes configuration.

GET /processes

This method returns the list of processes accessible for the user.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **id:** The id of the process, to show only one.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of processes and its properties.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:

{"type":"get",
"url":"processes",
"data":"company=demo04&session=QhuFhXB9MXprVKEfakhwaZxgA5osZg1o2NDaZeDunjRBX7cETnq6fehDH0K1&id=A0L1"}

Reponse (success):

"status": "0",
"data": [
{
"order": "00001",
"id": "A0L1",
"desc": "Robot Line 1",
"calculation": "piezas",
"cycleot": "0",
"cyclet": "11",
"maxcyclet": "0",
"cycleq": "1",
"npcauto": "0",
"unit": "piezas",
"minopers": "0",
"opers": "3",
"y0": "77,47",
"g0": "87,15",
"y1": "100",
"g1": "100",
"y2": "99,8",
"g2": "99,8",
"y3": "0",
"g3": "0",
"y4": "0",
"g4": "0",
"y5": "0",
"g5": "0",
"y6": "0",
"g6": "0",
"y7": "0",
"g7": "0",
"trace": "1",
"next_processes": "",
"resps": "",
"stshort": "50",
"prshort": "1",
"microstop": "",
"url": "",
"lockrec": "525600",
"man": "72",
"genpstart": "0",
"interign": "0",
"limprodtosche": "0",

"allowconf": "0",
"type": "0",
"requirenotify": "0",
"wbs": "",
"notifyratios": "0",
"represent": "0",
"excessordefect": "0",
"schedulefields": "0",
"img": "",
"imgurl": "http://localhost/edinnm2/img/4444444444/Mach_A0L1_",
"operationtime": "0"
}
],
"message": "Success"}

Response (error):
{"status":"1007",
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Processes_Config.html'>for additional information follow this link<\/a>",
"message":"Session not stablished"}

GET /processes/{process}/reta

Returns the Id of the favorite tabs assigned to that process

Request attribute:

Process [mandatory]: the id of the process

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: Returns the Id of the favorite tabs assigned to that process, or true if there is no data to be returned.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:
/processes/A0L1/reta
{
"session": 01qgmfcZmZ4nqg2eNGpxY4JlpmhJuKw1H31p5rtD1a112SSK8S8odROFIKgn
"company": DEMO04
}
Response:
{
"status": 0,
"Data":
{
true,

```
}
"message": "Success"
}
```

POST /processes

This method creates a process. The user must have the administrator role.

Request parameters (for additional information please review the process configuration):

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **id [mandatory]**: the id of the process.
- **order**: the relative order to be shown.
- **desc**: the description of the process.
- **calculation**: the calculation type of the process.
- **cycleot**: operator time in seconds. How many seconds from the Cycle Time (see next) correspond to the resource (person) work.
- **cyclet**: time in seconds per cycle of production. Please see Maximum speed.
- **maxcyclet**: this is the template cycle time to apply when creating new orders.
- **cycleq**: number of units expected per cycle of production. Please see Maximum speed.
- **npcauto**: indicates the number of units of results with a maintained speed for the system to automatically update the maximum speed of the process. Automatic update of maximum speed is disconnected if this value is left to 0.
- **unit**: the unit id of what the process produces.
- **minopers**: minimum number of working persons that can work at the process at a certain moment.
- **opers**: maximum number of working persons that can work at the process at a certain moment.
- **y0**: yellow objective for availability.
- **g0**: green objective for availability.
- **y1**: yellow objective for speed.
- **g1**: green objective for speed.
- **y2**: yellow objective for quality.
- **g2**: green objective for quality

- **y3**: yellow objective for productive availability.
- **g3**: green objective for productive availability.
- **y4**: yellow objective for the quantity of MTBF.
- **g4**: green objective for the quantity of MTBF.
- **y5**: yellow objective for the MTBF.
- **g5**: green objective for the MTBF.
- **y6**: yellow objective for the MTTR.
- **g6**: green objective for the MTTR.
- **y7**: yellow objective for the OCE.
- **g7**: green objective for the OCE.
- **trace**: indicates if the signals should be sent to the registry (0 for inactive, or 2 for active).
- **next_processes**: the list (separated by ,) of the next processes in the line.
- **resps**: the list of the responsibles (ids from Resources).
- **stshort**: minimum stop time in seconds in order to be registered as failure (FAI). Stops with less time will be registered as Microstops.
- **prshort**: minimum production time in seconds in order to be registered as production. This cannot be 0 and should be correctly configured considering server parameters. Contact your technical support and request to check this value according to option "Check Status Every" at the Monitor window of the server.
- **microstop**: status that will be automatically used to automatically justify microstops.
- **url**: document or application that will be opened when the user clicks on the  that appears after having clicked on the process, on the main window. Please see how to configure URLs.
- **lockrec**: indicates after how many minutes the time and production registers will be blocked for users without supervisor permission. This allows to force justifications before a certain time.
- **man**: combination of bits to indicate how the process is monitored manually.
- **genpstart**: this mark, when 1, indicates to the system to automatically generate a new status to indicate that a new work has started, every time a new result with 0 quantity is inserted, if at the status configuration the option Work start

has been marked for one, and no more than one, status.

- **interign**: marks, when 1, the process as to be ignored when transferring from and to information to another system, like ERPs. Please see *integration with other systems*.
- **limprodtosche**: if marked, with 1, the process is restricted to follow the orders scheduled for it, and that can be viewed at the view schedule window.
- **allowconf**: if marked with 1, allows the user to configure the process and to add more statuses and more results.
- **type**: indicates, when 1, that the process is dedicated to attend services. This mark will, for example, show the working time of these processes as services in the FTE report.
- **requirenotify**: this selector, when 1, forces the user to indicate quantities when closing an order in the selected items.
- **wbs**: reserved for advanced scheduling of tasks.
- **notifyratios**: if selected, the system will notify by email (1) when the selected ratio (OEE, Availability, Speed or Quality) is below the yellow target, in any of the defined periods.
- **represent**: indicates, when 1, if the process is the representant of an area.
- **excessordefect**: in case notifyratios is set to Scheduling, then this will include the deviation % of excess or defect in production of orders to be notified.
- **img:**an image to upload that represents the process visually.
- **schedulefields:** (to be implemented) is a combination of bits to indicate which columns of the data grid in the Schedule window are visible for this process.
- **operationtime:** the time the process has been in operation.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of processes and its descriptions.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

```
Content-Type: application/json
Request:
{
   "company": "demo01",
   "session": "LebEWTNf8GEhxfQeeGbBmVUlxmUIDLVdEoeVYrnWNzkdRXKxmgCNc92wnSjA",
   "order": "00001",
```

```
   "id": "1CON",
   "desc": "CONVEYOR 1",
   "calculation": "PIECES",
   "cyclet": "10",
   "maxcyclet": "0",
   "cycleq": "1",
   "npcauto": "0",
   "unit": "PIECES",
   "minopers": "0",
   "opers": "10",
   "y0": "60",
   "g0": "60",
   "y1": "90",
   "g1": "95",
   "y2": "95",
   "g2": "98",
   "y3": "60",
   "g3": "70",
   "y4": "0",
   "g4": "0",
   "y5": "0",
   "g5": "0",
   "y6": "0",
   "g6": "0",
   "y7": "60",
   "g7": "60",
   "trace": "0",
   "machnext": "2ROT",
   "resps": "",
   "stshort": "180",
   "prshort": "10",
   "microstop": "MIC",
   "url": "",
   "lockrec": "120",
   "man": "9",
   "genpstart": "0",
   "interign": "0",
   "limprodtosche": "0",
   "allowconf": "0",
   "type": "0",
   "requirenotify": "0",
   "wbs": "",
   "notifyratios": "0",
   "represent": "0",
   "img": "",
   "operationtime": "22",
   "schedulefields": "2"
}
Reponse (success):
{
   "status": "0",
   "data": true,
   "message": "Success"
}
```


POST /processes (ISA-95 compliant, for more information please read the B2MML document)

This method creates a process using json encoded B2MML data. The user must have the administrator role.

Request parameters (for additional information please review the process configuration):

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **data [mandatory]**: the json encoded B2MML data defining the process.

Example:

```
{
   "SyncEquipment":
   {
      "ApplicationArea":
      {
         "Sender": {"ComponentID": "ERP","ConfirmationCode": "Always"},
         "CreationDateTime": "2019-04-10T16:30:32"
      },
      "DataArea":
      {
         "Sync": {"ActionCriteria": {"ActionExpression": { "@actionCode":
         "Add" }}},
         "Equipment":
         {
            "ID": "WC1",
            "Description": "WorkCell 1",
            "Location":
            {
               "EquipmentID": "DEMO01",
               "EquipmentElementLevel": "Enterprise",
               "Location":
               {
                  "EquipmentID": "P1",
                  "EquipmentElementLevel": "Site",
                  "Location":
                  {
                     "EquipmentID": "PL1",
                     "EquipmentElementLevel": "ProductionLine"
                  }
               }
            },
            "EquipmentProperty":
            [
               {"ID": "Id_Calc","Value": {"ValueString": "CycleT","DataType":
               "string","UnitOfMeasure":""}},
               {"ID": "CycleQ","Value": {"ValueString": "4","DataType":
               "int","UnitOfMeasure":""}},
               {"ID": "CycleT","Value": {"ValueString": "3600","DataType":
               "int","UnitOfMeasure":""}},
               {"ID": "NPCAuto","Value": {"ValueString": "0","DataType":
               "byte","UnitOfMeasure":""}},
               {"ID": "Id_Unit","Value": {"ValueString": "M3","DataType":
               "string","UnitOfMeasure":""}},
               {"ID": "Opers","Value": {"ValueString": "3","DataType":
```

```
                "int","UnitOfMeasure":""}},
                {"ID": "StShort","Value": {"ValueString": "60","DataType":
                "int","UnitOfMeasure":""}},
                {"ID": "PrShort","Value": {"ValueString": "60","DataType":
                "int","UnitOfMeasure":""}},
                {"ID": "MachNext","Value": {"ValueString":
                "WC2,WC3","DataType": "string","UnitOfMeasure":""}},
                {"ID": "Trace","Value": {"ValueString": "1","DataType":
                "integer","UnitOfMeasure":""}},
                {"ID": "Resps","Value": {"ValueString": "PERS1","DataType":
                "string","UnitOfMeasure":""}},
                {"ID": "URL","Value": {"ValueString":
                "nombre_del_archivo","DataType":
                "string","UnitOfMeasure":""}},
                {"ID": "Id_StTU","Value": {"ValueString": "FAI","DataType":
                "string","UnitOfMeasure":""}},
                {"ID": "LockRec","Value": {"ValueString": "10080","DataType":
                "integer","UnitOfMeasure":""}},
                {"ID": "Man","Value": {"ValueString": "7","DataType":
                "integer","UnitOfMeasure":""}},
                {"ID": "Type","Value": {"ValueString": "7","DataType":
                "integer","UnitOfMeasure":""}},
                {"ID": "GenPStart","Value": {"ValueString": "0","DataType":
                "integer","UnitOfMeasure":""}},
                {"ID": "InterIgn","Value": {"ValueString": "7","DataType":
                "integer","UnitOfMeasure":""}},
                {"ID": "LimProdToSche","Value": {"ValueString":
                "7","DataType": "integer","UnitOfMeasure":""}},
                {"ID": "AllowConf","Value": {"ValueString": "7","DataType":
                "integer","UnitOfMeasure":""}}
            ]
        }
    }
}
```

Response example:

```
{
   "status":"0",
   "data":"{
      \"ConfirmBOD\":{
         \"ApplicationArea\":{
            \"Sender\":{\"ComponentID\":\"EDINN\",\"ConfirmationCode\":
            \"Always\"},
            \"CreationDateTime\":\"2019-11-12T12:16:18\"
         },
         \"DataArea\":{
            \"Confirm\":{\"ResponseCriteria\":{\"ResponseExpression\":
            {\"@actionCode\":\"Accepted\"}}},
            \"BOD\":{\"Description\":\"Accepted WC1\",\"Note\":[]}
         }
      }
   }",
   "message":"Success"
}
```

DELETE /processes/{id}

This method deletes a process. The user must have the administrator role.

Request attribute:

- **id [mandatory]**: the Id of the session to delete.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of processes and its descriptions.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Content-Type: application/json

Request:

```
{
  "company": "demo01",
  "session":
"LebEWTNf8GEhxfQeeGbBmVUlxmUlDLVdEoeVYrnWNzkdRXKxmgCNc92wnSjA"
}
```

Response (success):

```
{
    "status": 0,
    "data": null,
    "message": "Success"
}
```

Operation

This section handles the processes operation.

GET /processes/{process}/activeresult

This method returns the active result.

Request attribute:

- **Process [mandatory]:** the id of the process

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of results.
- **message**: "success" or a descriptive information of the error in any other case.

Content-Type: application/json

Request: /processes/A0L1/activeresult?session=rB9RmQypCzdbiJjomQktYH2Uh8Rdljtal0Cn1ga1qjtRMKWyNNP405lyBs2h&company=DEMO01

Reponse (success):

```
{
    "status": "0",
    "data": [
     {
            "Id_prod": "8563",
            "Desc": "8563 1.8L Lynx"
     }
    ]
 "message": "Success"
}
```

GET /processes/{process}/cpk

This method returns the cpk information of a process.
Request attribute:

- **Process [mandatory]:** the id of the process

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **datefrom [mandatory]**: start date of the period.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the CPK of the selected process.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:
/processes/A0L1/cpk
{
"session": 01qgmfcZmZ4nqg2eNGpxY4JlpmhJuKw1H31p5rtD1a112SSK8S8odROFIKgn
"company": DEMO04
“Datefrom”: 20171001000000
}
Response:
{
"status": 0,
"Data":
{
"Cpk": "0"
}
"message": "Success"
}

GET /processes/{process}/cycleTime

This method returns the cycle time information of a process

Request attribute:

- **Process [mandatory]:** the id of the process

Request parameters:

- **company [mandatory]:** The company name or id.
- **session [mandatory]:** The Id of the active session.

Response:

- **status**: Returns "0" on success, and an error code in any other case.
- **data**: Returns current cycle time information of the process.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Request:

```
/processes/A0L1/cycleTime
{
"session": 01qgmfcZmZ4nqg2eNGpxY4JlpmhJuKw1H31p5rtD1a112SSK8S8odROFlKgn
 "company": DEMO04
}
Response:
{
        "status":0,
        "data":
{
        "Val": "3470",
        "DateTime": "20200902134938",
        "CycleOT": "0",
        "CycleT": "11"
}
        "message":"Success"
}
```

GET /processes/{process}/kpicalculated

This method returns the calculated KPI information of a process, given an algorithm, a set of one or more variables, and a description.

Request attribute:

- **Process [mandatory]:** the id of the process

Request parameters:

- **company [mandatory]:** The company name or id.
- **session [mandatory]:** The Id of the active session.
- **datefrom [mandatory]**: start date of the period.
- **dateto [mandatory]**: end date of the period.
- **algorithm [mandatory]:**mathematic function used to calculate the result. Variables must be enclosed with quotation marks, for example: "Width".
- **variables [mandatory]:**variables that will be requested in order to calculate the quantity. There are predefined variables in configuration.
- **description [mandatory]**: A description of the result to be obtained.

Response:

- **status**: Returns "0" on success, and an error code in any other case.
- **data**: Returns calculated KPI information of the process.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

```
Request:
/processes/A0L1/kpicalculated
{
```

```
"session": 01qgmfcZmZ4nqg2eNGpxY4JlpmhJuKw1H31p5rtD1a112SSK8S8odROFIKgn
 "company": DEMO04
 "datefrom":20170401000000
“dateto”: 20201010000000
"algorithm": "NumOpers"*"CycleT"
"variables": NumOpers,CycleT
"description": "Result"
}
Response:
{
"status": "0",
"data": [
  {
""result"": "1"
}
],
"message": "Success"
}
```

GET /processes/{process}/monitor

This method returns the monitor information of a process

Request attribute:

- **Process [mandatory]:** the id of the process

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the monitor item value of the selected process, or null in the case that there is no data to return.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

```
Request:
/processes/A0L1/monitor
{
"session": 01qgmfcZmZ4nqg2eNGpxY4JlpmhJuKw1H31p5rtD1a112SSK8S8odROFIKgn
 "company": DEMO04
}
Response:
{
"status": 0,
"Data":
{
null
}
"message": "Success"
}
```

GET /processes/{process}/mtbf

This method returns the mtbf information of a process
Request attribute:

- **Process [mandatory]:** the id of the process

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the four MTBF fields of the selected process.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:
/processes/A0L1/mtbf
{
"session": 01qgmfcZmZ4nqg2eNGpxY4JlpmhJuKw1H31p5rtD1a112SSK8S8odROFIKgn
"company": DEMO04
}
Response:
{
"status": 0,
"Data":
{
{
"MTBF": "0",
"MTTR": "0",
"MTBFQ": "0",
"MTBFLastUpdate": "04/09/2020 10:59:48"
}
}
"message": "Success"
}

GET /processes/{process}/production

This method returns the amount of goods produced by a process between a start and end date, pertaining to classifications of: bad, good, theoretical, and total.
Request attribute:

- **Process [mandatory]:** the id of the process

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **datefrom [mandatory]**: start date of the period.

- **dateto [mandatory]**: end date of the period.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the quantity of goods produced of a process.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:
/processes/A0L1/production
{
"session": 01qgmfcZmZ4nqg2eNGpxY4JlpmhJuKw1H31p5rtD1a112SSK8S8odROFIKgn
"company": DEMO04
“Datefrom”: 20201001000000
“Dateto”: 20201030000000
}
Response:
{
"status": 0,
"Data":
{
"Prod": "",
"ProdBad": "0",
"ProdTheor": "0",
"ProdGood": "0",
"ProdTotal": "0"
}
"message": "Success"
}

GET processes/{process}/productionKg

This method returns the bad, theoretical, good, and total weight of goods produced by a process between a start and end date.
Request attribute:

- **Process [mandatory]:** the id of the process

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **datefrom [mandatory]**: start date of the period.
- **dateto [mandatory]**: end date of the period.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the produced Kgs of a process

- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:
/processes/A0L1/productionKg
{
"session": 01qgmfcZmZ4nqg2eNGpxY4JlpmhJuKw1H31p5rtD1a112SSK8S8odROFIKgn
"company": DEMO04
“Datefrom”: 20201001000000
“Dateto”: 20201030000000
}
Response:
{
"status": 0,
"Data":
{
"ProdBadKgs": "0",
"ProdTheorKgs": "0",
"ProdGoodKgs": "0",
"ProdTotalKgs": "0"
}
"message": "Success"
}

GET /processes/{process}/prodTheorN

This method returns the theoretical amount of goods produced by a process between a start and end date.
Request attribute:

- **Process [mandatory]:** the id of the process

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **datefrom [mandatory]**: start date of the period.
- **dateto [mandatory]**: end date of the period.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the theoretical quantity produced of a process
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:
/processes/A0L1/prodTheorN
{
"session": 01qgmfcZmZ4nqg2eNGpxY4JlpmhJuKw1H31p5rtD1a112SSK8S8odROFIKgn
"company": DEMO04

```
“Datefrom”: 20201001000000
“Dateto”: 20201030000000
}
Response:
{
"status": 0,
"Data":
{
"ProdTheorN": "0"
}
"message": "Success"
}
```

GET /processes/{process}/QActivity

This method returns the activity of the process taking into account the production of the variables indicated in QColumn1 and QColumn2, where the total activity time is the time resulting from all the states of the period multiplying the time of each one by the percentage of activity of the state.

Request attribute:

- **Process [mandatory]:** the id of the process

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **datefrom [mandatory]**: start date of the period.
- **dateto [mandatory]**: end date of the period.
- **QColumn1 [mandatory]:** first quantity column to perform the calculation.
- **QColumn2:** second quantity column to perform the calculation.
- **QFactor:** time conversion ratio. If omitted, the activity will be displayed in units per hour.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the
- **message**: "success" or a descriptive information of the error in any other case.

Example:

```
Request:
/processes/A0L1/QActivity
{
"session": 01qgmfcZmZ4nqg2eNGpxY4JlpmhJuKw1H31p5rtD1a112SSK8S8odROFIKgn
 "company": DEMO04
```

```
“Datefrom”: 20201001000000
“Dateto”: 2020103000000
"QColumn1": QT
}
Response:
{
"status": "0",
"data": [
  {
"QTActivity": "577.95"
}
],
"message": "Success"
}
```

GET /processes/{process}/scrapcauses/top

This method returns the active product.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **process [mandatory]**: the Id of the process.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of results.
- **message**: "success" or a descriptive information of the error in any other case.

```
Content-Type: application/json

Request: /processes/A0L1/scrapcauses/top?
session=rB9RmQypCzdbiJjomQktYH2Uh8Rdljtal0Cn1ga1qjtRMKWyNNP405lyBs2h&company=
DEMO01

Reponse (success):

{
    "status": "0",
    "data": [
     {
            "Scrap": "R0001 - Scrap"
     }
    ]
 "message": "Success"
```

```
}
```

GET /processes/{process}/speeds

This method returns the speed fields of information of a process.
Request attribute:

- **Process [mandatory]:** the id of the process

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **datefrom [mandatory]**: start date of the period.
- **dateto [mandatory]**: end date of the period.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the various speed fields of the selected process.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:

```
{
/processes/A0L1/speeds
}
{
"session": 01qgmfcZmZ4nqg2eNGpxY4JlpmhJuKw1H31p5rtD1a112SSK8S8odROFIKgn
 "company": DEMO04
“Datefrom”: 20171001000000
“Dateoto”: 20201110000000
}
```

Response:

```
{
"status": 0,
"Data":
{
"Speed0": "0",
"Speed0h": "0",
"SpeedR": "0",
"SpeedRh": "0",
"SpeedAvgR": "0",
"SpeedAvgRh": "0",
"SpeedT": "0",
"SpeedTh": "433.7349",
"SpeedT*S%": "0",
"SpeedT*S%h": "0",
"CycleTEquiv": "0"
}
```

"message": "Success"
}

GET /processes/{process}/speeds/class

This method returns the speed fields of information of a process, corresponding to the class.
Request attribute:

- **Process [mandatory]:** the id of the process

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the various speed fields of the selected process by class.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:
{
/processes/A0L1/speeds/class
}
{
"session": 01qgmfcZmZ4nqg2eNGpxY4JlpmhJuKw1H31p5rtD1a112SSK8S8odROFlKgn
"company": DEMO04
}
Response:
{
"status": "0",
"data": [
{
"SpeedWOClassh[
SpeedWOByClass": "0"
}
],
"message": "Success"
}

GET /processes/{process}/state

This method returns the current state information of a process: status, result and current order, date and time they changed, alerts information, current cycle time, etc. If the process is omitted, it will query for all processes, which could take longer. Be sure to put enough timeout.

Request attributes:

- **process [optional]**: process identifier.

Request parameters:

- **company [mandatory]**: The company name or id.
- **session [mandatory]**: The Id of the active session.
- **showarea:** displays the parent areas of the returned processes.
- **showdescription:** displays the description of the master data used: processes, areas, statuses and results.
- **showfields:** displays fields and field values belonging to the current result.
- **showplanning:** displays the next planned status with its planned date and time.
- **device:** displays the KPIs configures in the device.

Response:

- **status**: Returns "0" on success, and an error code in any other case.
- **data**: Returns current state information of the process.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Request:

{"type":"get",
"url":"processes/A0L1",
"data":"company=demo01&session=lparIA3NO0TizbbcKdG3jrvmeholoqkXLemLVkGmHtXBaPS9Pc152MXBMXu3&showdescription=1&showarea=1&showfields=1&showplanning=1&device=INIT"}

Reponse (success):

{"status":"0",
"data":
[{"Id":"7101","NextPiece":"","Limit":"","Id_TUnext":"","Id_TUlimit":"","Id_TUSince":"20200824060000","Id_TU":"6301","PNSince":"20200720083753","PN":"2020040103-10","id_prodSince":"20200515095146","Id_Prod":"BB0932428102\/4351","Sch_Col":"0","Pro_Col":"0","Ti0_Col":"0","Ti1_Col":"0","Ti2_Col":"0","Co0_Col":"0","Co1_Col":"0","Co2_Col":"0","Aut_Col":"0","SPC0_Col":"0","SPC1_Col":"0","GQT_Col":"0","QuickUpd":"0","QUDateTime":"","CycleT":"2790780","CycleQ":"11","SPCReq":"","LastSentPR":"","LastResultId":"20200824105846AABCENTRAL","LastResultWhen":"20200821155056","LastResultPaused":"0","LastResultId_Paused":"","LastResultModified":"","TUTypeCurrent":"3","AvgCycleT":"15","AvgCycleQ":"4.07142E-07,"process":"RS15","status":"LIMPIEZA","result":"","areas":
[{"id":"7100","description":"BOLSAS"}],"fields":
[{"variable":"Lot","value":"BP2023082301"}],"next_id_tu":"","next_tu":"","next_id_tu_datetime":"","KPIs":
{"Good (Unit)":{"type":"Number","value":"0","unit":"Piezas"},"Produced (un.)":
{"type":"Number","value":"0","unit":"Piezas"},"Productive (Minutes)":
{"type":"Number","value":"0","unit":"min"},"Not productive (Minutes)":
{"type":"Number","value":"0","unit":"min"},"Nominal (un.\/Minutes)":
{"type":"Number","value":"0","unit":"Piezas\/min"},"Average (un.\/Minutes)":
{"type":"Number","value":"0","unit":"Piezas\/min"}}}],"message":"Success"}

Reponse (error):

{"status":1012,
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_processes_state.html'>for additional information follow this link<\/a>",
"message":"Process not found"}

GET /processes/{process}/state/time

This method returns the relevant times (such as the time in production) of a process over a selected time period, for a concrete state.
Request attribute:

- **Process [mandatory]:** the id of the process

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **datefrom [mandatory]**: start date of the period.
- **dateto [mandatory]**: end date of the period.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the various time fields of the selected process for the given concrete state.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:
/processes/A0L1/state/time
{
"session": LzRlv4vjbyHWPhO2B2ulNjmSZFNmvpW4aqbLaoz1n6KJF91RJPCsZcmVLzpy
"company": DEMO04
“Datefrom”: 20171011000000
“Dateto”: 20201110000000
}
Reponse:
{
"status": "0",
"data": [
{
"TimeStatus[TimeStatus]": "0"
}
],
"message": "Success"
}

GET /processes/{process}/times

This method returns the relevant times (such as the time in production) of a process over a selected time period.
Request attribute:

- **Process [mandatory]:** the id of the process

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **datefrom [mandatory]**: start date of the period.
- **dateto [mandatory]**: end date of the period.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the various time fields of the selected process.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:
/processes/A0L1/times
{
"session": LzRlv4vjbyHWPhO2B2ulNjmSZFNmvpW4aqbLaoz1n6KJF91RJPCsZcmVLzpy
"company": DEMO04
“Datefrom”: 20171011000000
“Dateto”: 20201110000000
}
Response:
{
"status": 0,
"Data":
{
"TimeMin": "1621440",
"TimeNotProdHr": "1739.05055555556",
"TimeNotProdMin": "104343.033333333",
"TimeNowMin": "1632148",
"TimeProdMin": "857179.4",
"TimeProdHr": "14286.3233333333",
"TimeResMin": "-54575"
}
"message": "Success"
}

OEE

OEE returns the actual OEE information for a process and its OEE objectives.

GET /processes/{process}/oee

Request attribute:

- **process [mandatory]**: Process identifier.

Request parameters:

- **company [mandatory]**: The company name or id.
- **session [mandatory]**: The Id of the active session.

Response:

- **status**: Returns "0" on success, and an error code in any other case.
- **data**: Returns the actual OEE and the target OEE for the current process.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

```
Content-Type: application/json

Request: /processes/A0L1/oee?
company=demo01&session=lparIA3NO0TizbbcKdG3jrvmehoIoqkXLemLVkGmHtXBaPS9Pc15
2MXBMXu3

Reponse (success):

{
  "status": "0",
  "data": {
    "actual": {
      "oee": "86.52"
    },
    "target": {
      "yellow": "77.32",
      "green": "86.98"
    }
  },
  "message": "Success"
}
```

Relations

This section handles the relations configuration.

Process-Status Configuration

Process-Status handles the relations between processes ans statuses management.

GET /processes/{process}/statuses

This method returns the list of statuses related to a process.

Request attribute:

- **process [mandatory]**: the id of the process whose related statuses you want to display.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **type:** status type to filter by (0 production, 1 idle, 2 dependency, 3 failure, 4 unscheduled)

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of statuses associated to the process and their attibutes.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"get",
"url":"processes/7101",
"data":"company=demo04&session=QhuFhXB9MXprVKEfakhwaZxgA5osZg1o2NDaZeDunjRBX7cETnq6fehDH0K1&type=4"}

Reponse (success):
"status": "0",
"data": [
{
"id": "TPM",
"description": "TPM",
"statustype": "4",
"maxtime": "0",
"next_status": "",
"propagate": "0",
"week_days": "0",
"scheduledtime": "0600",
"order": "00018",
"weekdays": "0",
"activity": "0"
}
...

],
"message": "Success"}

Response (error):
{"status":"1007",
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Processes_Config.html'>for additional information follow this link<\/a>",
"message":"Session not stablished"}

POST /processes/{process}/statuses

This method creates a relation between a process and a status. The user must have the administrator role.

Request attribute:

- **process [mandatory]**: the id of the process to who you want relate the status.

Request parameters (for additional information please review the status configuration and the process-status-result configuration):

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **status [mandatory]**: the Id of the status you want to relate with the given process.
- **maxtime:** maximum time in minutes that the process can be in this status. If any value different from 0 is specified, the system will automatically close this status when it is open and the indicated minutes are surpassed.
- **next_status:** the next status to automatically pass when the maximum minutes are surpassed.
- **propagate:** if the status will be automatically propagated to processes that are later in the area. Please see difference between production lines and areas.
- **week_days:** status can be automatically scheduled by days of the week and time.
- **scheduledtime:** time at which the status will automatically be inserted.
- **activity:** percentage of activity that is required from a working user when a process is in this status. This is a requirement configuration for the activity report.

DELETE /processes/{process}/statuses/{status}

This method deletes a relation between a process and a status. The user must have the administrator role.

Request attribute:

- **process [mandatory]**: the id of the process from who you want to remove the

status.

- **status [mandatory]**: the id of the status to delete.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "success" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.

Process-Status-Result Configuration

Process-Status-Result handles the relations between processes, statuses and results management.

GET /processes/{process}/statuses/{status}/results

GET/processes/{process}/statuses/{status}/results/{result}

This method returns the list of results related to a status and a process.

Request attribute:

- **process [mandatory]**: the id of the process whose related results you want to display.
- **status [mandatory]**: the id of the status whose related results you want to display.
- **result:** the id of the result whose information you want to display. If not indicated it shows all results.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **prefix:** if a text is indicated, it uses it to filter the results.
- **type:** result type to filter by (0 good, 1 rework, 2 scrap, 255 all). The default values is 255.
- **history**: indicates if all the changes of the PSR relations are shown or only the most recent value (0 is not shown, 1 is shown). The default value is 0.
- **showfields**: indicates if the associated fields and their default values are shown (0 not shown, 1 if shown). The dafault value is 0.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of results.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"get",
"url":"processes/7101/statuses/0/results",
"data":"company=demo04&session=QhuFhXB9MXprVKEfakhwaZxgA5osZg1o2NDaZeDunjRBX7cETnq6feh

DH0K1&type=0"}

Reponse (success):
"status": "0",
"data": [
{
"id": "PRD0001",
"description": "Product 0001",
"resulttype": "0",
"equivalence":"0.001",
"cycleot": "0",
"cyclet": "32",
"cycleq": "1",
"npcauto": "0",
"activity": "100",
"opers": "1",
"calculation": "",
"disallowconfig": "0",
"propagate": "0",
"datetime": "20210928060000",
"operationtime": "0",
"resultinpct": "0"
}
...
],
"message": "Success"}

Response (error):
{"status":"1007",
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Processes_Config.html'>for additional information follow this link<\/a>",
"message":"Session not stablished"}

POST /processes/{process}/statuses/{status}/results

This method creates a relation between a process a status and a result. The user must have the administrator role.

Request attribute:

- **process [mandatory]**: the id of the process to who you want relate the status.
- **status [mandatory]**: the id of the status to who you want relate the status.

Request parameters (for additional information please review the results configuration and the process-status-result configuration):

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **result [mandatory]**: the Id of the result you want to relate with the given process and status.
- **cycleot:** time in seconds of work of the resource (person) per cycle time.

- **cyclet:** seconds for each cycle time.
- **cycleq:** number of units of results expected after the cycle time.
- **npcauto:** indicates the number of units of results with a maintained speed for the system to automatically update the maximum speed of the process. Automatic update of maximum speed is disconnected if this value is left to 0.
- **activity:** percentage of activity that is required for the working user. This is a requirement configuration for the activity report.
- **opers:** maximum number of working users that can be working on this process, status and result.
- **calculation:** how to calculate the units produced by process. To be configured at the Calculation types configuration.
- **propagate:** if the result will be automatically propagated to processes that are later in the area. Please see difference between production lines and areas.
- **datetime:** date and time when this PSR relation starts applying. This is useful to change important parameters, like Cycle Time and Cycle Unit, without affecting historical data.

DELETE /processes/{process}/statuses/{status}/results/{result}

This method deletes a relation between a process, a status and a result. The user must have the administrator role.

Request attribute:

- **process [mandatory]**: the id of the process from who you want to remove the result.
- **status [mandatory]**: the id of the status from who you want to remove the result.
- **result [mandatory]**: the id of the result to delete.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "success" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.

Process-Recourse Configuration

Process-Recourse handles the relation between processes and recourses.

GET /processes/{process}/recourses

This method returns the list of recurses related to a process.

Request attribute:

- **process [mandatory]**: the id of the process whose related recourses you want to display.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of recourses associated to the process.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"get",
"url":"processes/7101/recourses",
"data":"company=demo04&session=QhuFhXB9MXprVKEfakhwaZxgA5osZg1o2NDaZeDunjRBX7cETnq6fehDH0K1"}

Reponse (success):
"status": "0",
"data": [
{
"recourse": "0701"
}
...
],
"message": "Success"}

Response (error):
{"status":"1007",
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Processes_Config.html'>for additional information follow this link<\/a>",
"message":"Session not stablished"}

POST /processes/recourses

This method creates a relation between a process and a recourse. The user must have the administrator role.

Request parameters (for additional information please review the recourses configuration):

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **process [mandatory]**: the id of the process to who you want relate the recourse.
- **recourse [mandatory]**: the Id of the recourse you want to relate with the process.

DELETE /processes/{process}/recourses/{recourse}

This method deletes a relation between a process and a recourse. The user must have the administrator role.

Request attribute:

- **process [mandatory]**: the id of the process from who you want to remove the recourse.
- **recourse [mandatory]**: the id of the recourse to delete.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "success" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.

Process-Result Configuraction

Process-State addresses the relationships between processes and results.

GET/processes/{process}/results/{result}

This method returns the properties of the relationship between a process and a result or all associated results.

Request attribute:

- **process [mandatory]:** the identifier of the process whose related results you want to display.
- **result:** the identifier of the result if we only want to show the information of this.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of results associated with the process and its attributes.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:


```
{
"type":"get",
"url":"processes/P001/results/R0000001",
"data":"company=testco&session=QhuFhXB9MXprVKEfakhwaZxgA5osZg1o2NDaZeDunjRBX7cETnq6fehDH0K1"
}
```


Reponse (success):


```
{
"status":"0",
"data":[
{"id":"R0000001","description":"Result0001","type":"P","y0":"50","g0":"60","y1":"80","g1":"90","y2":"90","g2":"95","y3":"60","g3":"70","y4":"10","g4":"5","y5":"240","g5":"120","y6":"60","g6":"30","y7":"50","g7":"50","equivalence":"1","weight":"1"}
],
"message":"Success"
}
```


Response (error):

{"status":"1007",
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Processes_Config.html'>for additional information follow this link<\/a>",
"message":"Session not stablished"}

POST/processes/{process}/results

This method creates a relationship between a process and a result. The user must be an administrator.

Request attribute:

- **process [mandatory]:** the id of the process to whom you want to relate the status.

Parámetros de solicitud (Request parameters, para obtener información adicional, revise la configuración de resultado y la configuración del estado del proceso-resultado):

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **result [mandatory]:** the Id of the result you want to associate with the given process.
- **equivalence**: equivalence between machine cycles and number of produced units of the result.
- **weight**: weight in kg with decimals of each unit of the result.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: null.
- **message**: "success" or a descriptive information of the error in any other case.

Request:
{
"type":"post",
"url":"processes/P001/results",
"data":"company=testco&session=uMCmYCvK0dMUNlenilQUNOYjtoCos30fZosNShoxdKJzkaqNXDjxlybgPB5z&result=R0000001&equivalence=2&weight=0.345"
}

Reponse (success):
{"status":0,"data":null,"message":"Success"}

Response (error):
{"status":"1007",
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Processes_Config.html'>for additional information follow this link<\/a>",
"message":"Session not stablished"}

DELETE/processes/{process}/results/{result}

This method erases the relationship between a process and a result. The user must be an administrator.

Request attributes:

- **process [mandatory]:** the id of the process from which you want to remove state.
- **result [mandatory]:** the id of the result to remove.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: null.
- **message**: "success" or a descriptive information of the error in any other case.

Request:
{
"type":"delete",
"url":"processes/P001/results/R0000001",
"data":"company=testco&session=uMCmYCvK0dMUNlenilQUNOYjtoCos30fZosNShoxdKJzkaqNXDjxlybgPB5z"
}

Reponse (success):
{"status":0,"data":null,"message":"Success"}

Response (error):
{"status":"1007",
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Processes_Config.html'>for additional information follow this link<\/a>",
"message":"Session not stablished"}

KPI

KPI gets the main KPIs (ratios).

GET processes/kpis

URL parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **process [mandatory]**: id of the process.
- **datefrom [mandatory]**: start date of the period.
- **dateto [mandatory]**: end date of the period.
- **team**: id of the team.
- **fields**: useful to filter with data fields.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the schema of areas and processes.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Content-Type: application/json

Request: /processes/kpis?company=demo01&session=LebEWTNf8GEhxfQeeGbBmVUlxmUlDLVdEoeVYrnWNzkdRXKxmgCNc92wnSjA&process=A0L1&dateFrom=20090302070000&dateTo=20090302150000

Reponse (success):


```
{
  "status": "0",
  "data": [
    {
      "RealTime": "480",
      "TotalTime": "480",
      "WaitingTime": "440,933333333333",
      "ProductionTime": "338,566666666667",
      "PlannedQuantity": "2444,45108497143",
      "ProducedQuantity": "2386",
      "GoodQuantity": "2380",
      "ScrapQuantity": "6",
```

```
    "ReworkQuantity": "0",
    "OperatingTime": "329,639922688854",
    "OEE": "68,67499",
    "OEETarget": "86,9757",
    "ProductiveEfficiency": "74,75958",
    "ProductiveEfficiencyTarget": "86,9757",
    "ProductiveAvailability": "76,7841",
    "Availability": "70,53472",
    "Speed": "97,60883",
    "Quality": "99,74854"
  }
 ],
 "message": "Success"
}
```


GET processes/kpis/dashboard

URL parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **process [mandatory]**: id of the process.
- **datefrom [mandatory]**: start date of the period.
- **dateto [mandatory]**: end date of the period.
- **team**: id of the team.
- **fields**: useful to filter with data fields.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the schema of areas and processes.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Content-Type: application/json

Request: /processes/kpis/dashboard?
session=Mehm6MjyZ7DPEPz2a4HhQeWCVPPcDBbsfWQAEyQqZhClDkJBqzxP0epMHOYm
&company=FEDERAL
MOGUL&process=L3&datefrom=20181010070000&dateto=20181010100000

Reponse (success):


```
{
  "status": "0",
  "data" : [
    {
        "OEE":"33.5594663023949",
        "DateTime":"20181004072045",
        "OEETargetYellow":"0",
        "OEETargetGreen":"72.369"
    }
]
  "message": "Success"
}
```


GET processes/kpis/followup

URL parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **process [mandatory]**: id of the process.
- **datefrom [mandatory]**: start date of the period.
- **dateto [mandatory]**: end date of the period.
- **planned:** indicates if the planned statuses should be taken.
- **hourly:** indicates if the data should be by hours or not.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the schema of areas and processes.

- **message**: "success" or a descriptive information of the error in any other case.

Example:

Content-Type: application/json

Request: processes/kpis/followup?
process=A0L1&datefrom=20241101060000&dateto=20241101070000&session=4mqPwIsCH
GdzofOcwXE3726RvOh5Lk6MseGryWPpD0XjlM4S9hdSakTuwZ5J&company=4444444444&p
lanned=0&hourly=0

Reponse (success):

{

"status": "0",

"data": [{"OEETheoricalQuantity":
{"type":"Number","value":"25200"},"OEEPlannedQuantityGreen":
{"type":"Number","value":"12927.6000090122"},"OEEPlannedQuantityYellow":
{"type":"Number","value":"9071.9998948574"},"SpeedTargetGreen":
{"type":"Number","value":"22680"},"SpeedTargetYellow":
{"type":"Number","value":"20160"},"PETheoricalQuantity":
{"type":"Number","value":"0"},"PEPlannedQuantityGreen":
{"type":"Number","value":"0"},"PEPlannedQuantityYellow":
{"type":"Number","value":"0"},"ProducedQuantity":
{"type":"Number","value":"0"},"ScrapQuantity":
{"type":"Number","value":"0"},"ReworkQuantity":
{"type":"Number","value":"0"},"GoodQuantity":
{"type":"Number","value":"0"},"OEETargetGreen":
{"type":"Number","value":"0.513"},"OEETargetYellow":
{"type":"Number","value":"0.36"},"Activity":{"type":"Number","value":"12600"},"Datetime":
{"type":"DateTime","value":"20241101060000"}}],

"message": "Success"

}

IoT

IoT handles the connection with the real world.

Configuration

This sections allows IoT configuration.

GET /iot

This method returns the list of available items.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **process**: the id of the process.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of items.
- **message**: "success" or a descriptive information of the error in any other case.

Example of response:


```
{
   "status": 0,
   "data": [{
      active: "1",
      consumptiontype: "",
      deadband: "0",
      description: "",
      device: "INIT",
      endpoint: "Kontron.OPCFatek.1",
      event: "0",
      function: "Production",
      host: "HOST",
      id: "343",
      identifier: "",
      item: "RS1.CTR.CTR0000",
      minimumcycle: "0",
      password: "",
      process: "A0L1",
      resetevery: "32000",
      senddescription: "0",
      storedb: "1",
      threshold: "50",
      type: "Counter",
      upd: "20190610130338",
      url: "localhost",
      user: "",
      value: "26469",
      variablenumber: "1",
      vendor: "edinnHIP"
   },
   {
      active: "1",
      consumptiontype: "",
      deadband: "0",
      description: "",
```

```
      device: "INIT",
      endpoint: "Kontron.OPCFatek.1",
      event: "0",
      function: "Production",
      host: "HOST",
      id: "346",
      identifier: "",
      item: "RS1.CTR.CTR0000",
      minimumcycle: "0",
      password: "",
      process: "A0L1",
      resetevery: "32000",
      senddescription: "0",
      storedb: "1",
      threshold: "80",
      type: "Counter",
      upd: "20190610130338",
      url: "localhost",
      user: "",
      value: "26469",
      variablenumber: "0",
      vendor: "edinnHIP"
   }],
   "message": "Success"
}
```


POST /iot

This method creates a new item. The user must have administrative permission.

Please remember that for counters that you want to be monitorized you need at least 2 entries in the IoT table: 1 for variable number 0 (the total) and another for variable number 1. This applies to these elements:

- PType_Production = 0
- PType_Scrap = 1
- PType_Rework = 2
- ...
- See the complete list here in functions.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **host**: localhost if not specified.
- **device [mandatory]**: the device which is the source of the item.
- **vendor [mandatory]**: the source from which the IoT gets the data. Please see constants.

- **url**: in OPC UA this is where the endpoints can be found. In OPC DA this is the IP address of the OPC Server. If not specified, "localhost" will be taken.
- **endpoint [mandatory]**: in OPC UA this indicates the endpoint to connect to. In OPC DA this indicates the name of the DCOM.
- **process**: the process identifier.
- **item [mandatory]**: the item identifier.
- **function [mandatory]**: the functionality of the item, please see constants.
- **type [mandatory]**: if the item is a counter, a signal or a user defined logic (UDL). Please see constants.
- **threshold [mandatory]** [0-100]: the minimum speed to consider that the process is working. This is mandatory when inserting a production (function), counter (type) and variable is totals.
- **minimumcycle [mandatory]** [0-100]: the minimum cycle time that has to be reached in order to consider the production.
- **description**: a free text to describe the item.
- **variablenumber [mandatory]**: the number of the variable that this item is representing, according to the calculation type associated with the process.
- **consumption_type** [optional]: the id of the consumption type.
- **identifier** [optional]: indicates the id of the:
  - result that will be automatically inserted if this item is functioning as scrap or rework item.
  - status that will be automatically inserted if this item is functioning as a justify of justify all item.
- **deadband** [optional][0-100]: this value allows to filter and ignore the changes of the signal are smaller than a certain percentage.
- **activate [mandatory]**: activates (1) or disactivates (0) this item.
- **storedb [mandatory]**: when storedb=0 it indicates that the item will be kept only in memory. If it is 1, it indicates that the item value will be stored also in the database in real time.
- **resetevery**: applies only to items which are of type counters, and indicates every how many counts this counter will reset.
- **senddescription [mandatory]**: indicates that the description indicated before will (1) or will be not (0) sent linked to the record of the generated result.
- **event [mandatory]** [100-200]: indicates if the item will be sent to the Registry

in the form of this number of event.

- **user [mandatory]**: the user to connect to the OPC UA.
- **password [mandatory]**: the password to connect to the OPC UA.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of items.
- **message**: "success" or a descriptive information of the error in any other case.

PUT /iot

This method updates an item. The user must have administrative permission. Mainly is used to update the production counter value.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **Id [mandatory]:** the id of the item.
- "**value" o "inc":** with "value" the item value can be changed, with "inc" (meaning increment) a determinated amount can be added to the current value of the item.
- **host**: localhost if not specified.
- **vendor**: the source from which the IoT gets the data. Please see constants.
- **url**: in OPC UA this is where the endpoints can be found. In OPC DA this is the IP address of the OPC Server. If not specified, "localhost" will be taken.
- **endpoint**: in OPC UA this indicates the endpoint to connect to. In OPC DA this indicates the name of the DCOM.
- **process**: the process identifier.
- **item**: the item identifier.
- **function**: the functionality of the item, please see constants.
- **type**: if the item is a counter, a signal or a user defined logic (UDL). Please see constants.
- **threshold** [0-100]: the minimum speed to consider that the process is working. This is mandatory when inserting a production (function), counter (type) and variable is totals.

- **minimumcycle** [0-100]: the minimum cycle time that has to be reached in order to consider the production.
- **variablenumber**: the number of the variable that this item is representing, according to the calculation type associated with the process.
- **consumptiontype** : the id of the consumption type.
- **identifier**: indicates the id of the:
  - result that will be automatically inserted if this item is functioning as scrap or rework item.
  - status that will be automatically inserted if this item is functioning as a justify of justify all item.
- **deadband** [0-100]: this value allows to filter and ignore the changes of the signal are smaller than a certain percentage.
- **activate**: activates (1) or disactivates (0) this item.
- **storedb**: when storedb=0 it indicates that the item will be kept only in memory. If it is 1, it indicates that the item value will be stored also in the database in real time.
- **resetevery**: applies only to items which are of type counters, and indicates every how many counts this counter will reset.
- **senddescription**: indicates that the description indicated before will (1) or will be not (0) sent linked to the record of the generated result.
- **event** [100-200]: indicates if the item will be sent to the Registry in the form of this number of event.
- **user**: the user to connect to the OPC UA.
- **password**: the password to connect to the OPC UA.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **message**: "success" or a descriptive information of the error in any other case.

DELETE /iot

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **id [mandatory]**: the Id of the item to delete.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of processes and its descriptions.
- **message**: "success" or a descriptive information of the error in any other case.

IoT

This sections allows IoT management.

GET /iot/read

This method returns the values of the item.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **id [mandatory]**: the id of the item.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the value of the item.
- **message**: "success" or a descriptive information of the error in any other case.

Example of response:


```
{
   "status": 0,
   "data": {
      "Val": "987",
      "ValSent": "986",
      "Upd": "20171017011517",
      "UpdSent": "20171017011516"
   },
   "message": "Success"
}
```


GET /iot/log

This method returns the log of the values of the items.

Request parameters:

- **company [mandatory]**: the company name or id.

- **session [mandatory]**: the Id of the active session.
- **datefrom [mandatory*]:** the start of the desired log. Equivalent to **startdatetime**
- **dateto [mandatory*]**: the end of the desired log. Equivalent to **enddatetime**
- **startdatetime [mandatory*]**: the start of the desired log. (deprecated, replace with **datefrom**)
- **enddatetime [mandatory*]**: the end of the desired log. (deprecated, replace with **dateto**)

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the value of the item.
- **message**: "success" or a descriptive information of the error in any other case.

Example of response:


```
{
   "status": "0",
   "data": [{
      datetime: "20190107171627",
      id: "1953480",
      id_signal: "343",
      mls: "258",
      val: "14500"
   },
   {
      datetime: "20190107171627",
      id: "1953481",
      id_signal: "345",
      mls: "258",
      val: "26680"
   }],
   "message": "Success"
}
```


Areas

This section handles areas.

Configuration

This section handles the areas configuration.

GET /areas

This method returns the list of areas visibles for the user.

Request parameters:

- **company [mandatory]**: The company name or id.
- **session [mandatory]**: The Id of the active session.

Response:

- **status**: Returns "0" on success, and an error code in any other case.
- **data**: Returns the list of areas and its attributes.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"get",
"url":"areas",
"data":"company=testco&session=08McAsZm2yvxupxXTwy5oGin9LekeT1KOPpfEEgZKAWKncP9dKbdPRg2nmzW"}

Reponse (success):
{"status": "0",
"data": [{"id": "ALL",
"description": "ALL",
"type": "0",
"elements": "A0L1,PROCT1,A0L2,B0L3,B0L4",
"formula": ""A0L1"+"A0L2"+"B0L3"+"B0L4"",
"y0": "82.3",
"g0": "93.62",
"y1": "100",
"g1": "100",
"y2": "99.9",
"g2": "99.9",
"y3": "0",
"g3": "0",
"y4": "0",
"g4": "0",
"y5": "0",
"g5": "0",
"y6": "0",
"g6": "0",
"y7": "0",
"g7": "0",
"background": "factory3.jpg",
"width": "1700",
"height": "1133",
"wbs": "W1.1",
"driverip": "8080",
"scheduled": "1",

"color": null,
"label": null,
"oee": null,
"state": null,
"graphic": null}],
"message": "Success"}

Reponse (error):
{"status": "1001",
"data": "for additional information follow this link",
"message": "Company name not found"}

GET /areas/{id}/processes

This method returns the list of processes belonging to the indicate area and visibles for the user.

Request attribute:

- **id [mandatory]**: the Id of the status.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: Returns "0" on success, and an error code in any other case.
- **data**: Returns the list of processes and its descriptions.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"get",
"url":"areas",
"data":"company=testco&session=08McAsZm2yvxupxXTwy5oGin9LekeT1KOPpfEEgZKAWKncP9dKbdPRg2nmzW"}

Reponse (success):
{"status": "0",
"data": [{"process": "7101","description": "Línea 1"},
{"process": "7102","description": "Línea 2"}],
"message": "Success"}

Reponse (error):
{"status": "1001",
"data": "for additional information follow this link",
"message": "Company name not found"}

POST /areas

This method creates an area. The user must have the administrator role.

Request parameters:

- **company [mandatory]:** The company name or id.
- **session [mandatory]:** The Id of the active session.
- **id [mandatory]:** Area identifier.
- **desc:** name of the area.
- **elements:** list of the processes thay are contained in the area.
- **alg:** algorithm that will be used to calculate the OEE of the area.
- **y0**: yellow objective for availability.
- **g0**: green objective for availability.
- **y1**: yellow objective for speed.
- **g1**: green objective for speed.
- **y2**: yellow objective for quality.
- **g2**: green objective for quality
- **y3**: yellow objective for productive availability.
- **g3**: green objective for productive availability.
- **y4**: yellow objective for the quantity of MTBF.
- **g4**: green objective for the quantity of MTBF.
- **y5**: yellow objective for the MTBF.
- **g5**: green objective for the MTBF.
- **y6**: yellow objective for the MTTR.
- **g6**: green objective for the MTTR.
- **y7**: yellow objective for the OCE.
- **g7**: green objective for the OCE.
- **img:** image of the area.
- **imgwidth:** image width.
- **imgheight:** image height.

- **etype:** type of the area.
- **recourses:** list of recourses that can see the area.
- **devices:** list of thevices that can see the area.
- **wbs:** the work breakdown structure of the area.
- **driverIP:**the driver IP for the area
- **scheduled:** whether the area is scheduled or not

Response:

- **status:** Returns "0" on success, and an error code in any other case.
- **data:** Returns the list of processes and its descriptions.
- **message:** "Success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"post",
"url":"areas",
"data":"company=testco","session":
"LebEWTNf8GEhxfQeeGbBmVUlxmUlDLVdEoeVYrnWNzkdRXKxmgCNc92wnSjA","id": "L1","desc":"Test area","devices":"ACA2A","recourses":"ADMIN","wbs":"W1.1"}

Reponse (success):
{"status":"0",
"data":"L1",
"message":"Success"}

Reponse (error):
{"status":"1007",
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Sessions.html'>for additional information follow this link<\/a>",
"message":"Session not stablished"}

DELETE /areas/{id}

This method deletes a results. The user must have the administrator role.

Request attribute:

- **id [mandatory]**: the Id of the status.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "success" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"delete",
"url":"areas/L1",
"data":"company=testco","session":
"LebEWTNf8GEhxfQeeGbBmVUlxmUlDLVdEoeVYrnWNzkdRXKxmgCNc92wnSjA"}

Reponse (success):
{"status":"0",
"data":"true",
"message":"Success"}

Reponse (error):
{"status":"1007",
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Sessions.html'>for additional information follow this link<\/a>",
"message":"Session not stablished"}

Operation

This section handles the areas operation.

GET /areas/{id}/status

Returns the actual status or the list of statuses of an area.

Request attribute:

- **id [mandatory]:** area identifier.

Request parameters:

- **company [mandatory]:** the company name or id.
- **session [mandatory]:** the id of the active session.

Response:

- **status:** returns "0" on success, and an error code in any other case.
- **data:** returns the actual status or the list of statuses of the current area.
- **message:** "success" or a descriptive information of the error in any other case.

Example:

```
Content-Type: application/json

Request: /areas/L1/status?
company=demo01&session=4rnTZHKyVnySZp036myysXYQc7HroVo8cbPvnXQKXsNEbg6OK
E0fdcL5KLQC&actual=1

Response (success):

{
   "status":"0";
   "data": {
        "production"
    },
   "message": "Success"
}
```

PUT /areas/{id}/status

To change the actual status of an area

Request attribute:

- **id [mandatory]:** area identifier.

Request parameters:

- **company [mandatory]:** the company name or id.
- **session [mandatory]:** the id of the active session.
- **status:** the new status.

Response:

- **status:** returns "0" on success, and an error code in any other case.
- **data:** null.
- **message:** "success" or a descriptive information of the error in any other case.

Example:

```
Content-Type: application/json

Request: /areas/L1/status?
company=demo01&session=4rnTZHKyVnySZp036myysXYQc7HroVo8cbPvnXQKXsNEbg6OKE0fd
cL5KLQC&actual=1
```

```
Reponse (success):

{
  "status": "0",
  "data": {
     "production"
  },
  "message": "Success"
}
```

OEE

OEE returns the actual OEE information for an area and its OEE objectives.

GET /areas/{id}/oee

Request attribute:

- **id [mandatory]**: area identifier.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the actual OEE and the target OEE for the current area.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

```
Content-Type: application/json

Request: /areas/L1/oee?
company=demo01&session=lparIA3NO0TizbbcKdG3jrvmehoIoqkXLemLVkGmHtXBaPS9Pc152MXBMXu3

Reponse (success):
```

```
{
 "status": "0",
 "data": {
  "actual": {
   "oee": "86.52"
  },
  "target": {
   "yellow": "77.32",
   "green": "86.98"
  }
 },
 "message": "Success"
}
```


Schedule

This section handles scheduled orders.

Configuration

This section handles scheduled orders configuration.

GET /schedule

Request parameters:

- **company [mandatory]:** The company name or id.
- **session [mandatory]:** The Id of the active session.
- **process:** process to which the order is assigned.
- **datefrom:** minimum date on which the order is scheduled.
- **dateto:** maximum date on which the order is scheduled.
- **pn [deprecated]:** work order code. Use the **order** parameter instead

- **order:** work order code.
- **status:** indicates the status of the orders that you want to obtain. The possible values are:
  - 0: Prescheduled
  - 1: Scheduled
  - 2: Working
  - 3: Paused
  - 4: Prefinished
  - 5: Finished
  - 6: Canceled
  - 7: All
  - 8: Pending (that is, prescheduled, schedueld, working or paused)
- If omitted, the default value is 8 (pending).
- **filter:** is a text string that is used to locate orders that contain it in any of these fields: order code, result code, result description, comment and reference.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns list of work orders that match the indicated criteria on success or the link to the request help on error.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Request:

{"type":"get",
"url":"schedule",
"data":"company=testco&session=kOORid1fkPhuSFAmakcSHEJ5Vzp3egJ87zfuCDDR5J4A4uLQBfMdwFankFfb&order=12345"}

Reponse (success):

{
"status":0,
"data":[
{
"Id":"20180515130923TMACENTRAL",
"MachDate":"P00120180515162312",
"OrderId":"12345",
"id_prod":"RES0001",
"QT":"82,5",
"Target":"82,5",

```
                "SetupTime":"0",
                "CycleT":"15",
                "CycleQ":"0,087766",
                "TeardownTime":"0",
                "DueDate":"20180515220000",
                "Status":"5",
                "Started":"20180523154422",
                "Ended":"20180523195228",
                "Comment":"Resultado de prueba 1",
                "Ref":"<Sch:12345><Req:12345><Seg:10>",
                "Id_Author":"CENTRAL",
                "Created":"20180515130925",
                "Id_Mod":"",
                "Modified":"",
                "Link":"",
                "URL":"",
                "Id_TU":"0",
                "Id_Sche":"",
                "Work":"",
                "WBS":"",
                "Esti":"0",
                "DisableNotif":"0"
            }
        ],
        "message":"Success"
}
```


Reponse (error):


```
{"status":1005,
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Schedule_Oper.html'>for additional information follow this link<\/a>",
"message":"No data available"}
```


POST /schedule (ISA-95 compliant, for more information please read the B2MML document)

This method creates a scheduled order using json encoded B2MML data. The user must have the scheduler role.

Request parameters:

- **company [mandatory]:** The company name or id.
- **session [mandatory]:** The Id of the active session.
- **data [mandatory]:** the json encoded B2MML data defining the scheduled order.

Response:

- **status:** Returns "0" on success, and an error code in any other case.
- **data:** the json encoded B2MML response.
- **message:** "Success" or a descriptive information of the error in any other case.

Example (the work order data is sent encoded to be used as the query part of a URL):

Request:
{"type":"post",
"url":"schedule",
"data":"**company=**testco**&session=**1FrqhchVXQoAgpxUbdKHWQkILZjMMAV0cesiSMmaRjwFvaWLwrpLyxJ6O9Zo**&data=**%7B%22SyncProductionSchedule%22%3A%7B%22ApplicationArea%22%3A%7B%22Sender%22%3A%7B%22ComponentID%22%3A%22ERP%22%2C%22ConfirmationCode%22%3A%22Always%22%7D%2C%22CreationDateTime%22%3A%222020-09-23T10%3A41%3A00%22%7D%2C%22DataArea%22%3A%7B%22Sync%22%3A%7B%22ActionCriteria%22%3A%7B%22ActionExpression%22%3A%7B%22%40actionCode%22%3A%22Add%22%7D%7D%7D%2C%22ProductionSchedule%22%3A%7B%22ID%22%3A%2220200901%22%2C%22ProductionRequest%22%3A%7B%22ID%22%3A%222020092304%22%2C%22StartTime%22%3A%222020-09-23T16%3A00%3A00%22%2C%22EndTime%22%3A%222020-09-23T18%3A00%3A00%22%2C%22SegmentRequirement%22%3A%7B%22ID%22%3A%2210%22%2C%22Description%22%3A%22BOLSAS%20DE%2024x28%22%2C%22EarliestStartTime%22%3A%222020-09-23T16%3A00%3A00%22%2C%22LatestEndTime%22%3A%222020-09-23T18%3A00%3A00%22%2C%22Duration%22%3A%22P0Y0M0DT2H0M0S%22%2C%22PersonnelRequirement%22%3A%7B%22PersonnelClassID%22%3A%7B%7D%2C%22PersonID%22%3A%7B%7D%2C%22Location%22%3A%7B%22EquipmentID%22%3A%22TESTCO%22%2C%22EquipmentElementLevel%22%3A%22Enterprise%22%7D%2C%22Quantity%22%3A%7B%22QuantityString%22%3A%221%22%2C%22DataType%22%3A%22double%22%2C%22UnitOfMeasure%22%3A%22UN%22%7D%7D%2C%22EquipmentRequirement%22%3A%7B%22EquipmentClassID%22%3A%7B%7D%2C%22EquipmentID%22%3A%227102%22%2C%22Location%22%3A%7B%22EquipmentID%22%3A%22TESTCO%22%2C%22EquipmentElementLevel%22%3A%22Enterprise%22%7D%2C%22Quantity%22%3A%7B%22QuantityString%22%3A%221%22%2C%22DataType%22%3A%22double%22%2C%22UnitOfMeasure%22%3A%22UN%22%7D%2C%22EquipmentRequirementProperty%22%3A%7B%22ID%22%3A%22ID_TU%22%2C%22Value%22%3A%7B%22ValueString%22%3A%226104%22%2C%22DataType%22%3A%7B%7D%2C%22UnitOfMeasure%22%3A%7B%7D%7D%7D%7D%2C%22MaterialRequirement%22%3A%5B%7B%22MaterialClassID%22%3A%7B%7D%2C%22MaterialDefinitionID%22%3A%22BB0932428102%5C%2F4351%22%2C%22MaterialLotID%22%3A%7B%7D%2C%22MaterialUse%22%3A%22Produced%22%2C%22Quantity%22%3A%7B%22QuantityString%22%3A%221000%22%2C%22DataType%22%3A%22double%22%2C%22UnitOfMeasure%22%3A%22Ml%22%7D%2C%22MaterialRequirementProperty%22%3A%7B%22ID%22%3A%22Equiv%22%2C%22Description%22%3A%22Equivalence%22%2C%22Quantity%22%3A%7B%22QuantityString%22%3A%221%22%2C%22DataType%22%3A%22double%22%2C%22UnitOfMeasure%22%3A%22KG%22%7D%7D%7D%2C%7B%22MaterialClassID%22%3A%7B%7D%2C%22MaterialDefinitionID%22%3A%22BO24035044%22%2C%22MaterialLotID%22%3A%7B%7D%2C%22MaterialUse%22%3A%22Consumed%22%2C%22Quantity%22%3A%7B%22QuantityString%22%3A%22238%22%2C%22DataType%22%3A%22double%22%2C%22UnitOfMeasure%22%3A%22KG%22%7D%2C%22MaterialRequirementProperty%22%3A%7B%22ID%22%3A%22Equiv%22%2C%22Description%22%3A%22Equivalence%22%2C%22Quantity%22%3A%7B%22QuantityString%22%3A%221%22%2C%22DataType%22%3A%22double%22%2C%22UnitOfMeasure%22%3A%22KG%22%7D%7D%7D%5D%7D%7D%7D%7D%7D%7D"}

Reponse (success):
{"status":"0",
"data":"{\"ConfirmBOD\":{\"ApplicationArea\":{\"Sender\":{\"LogicalID\":\"20200924090208ACJCENTRAL\",\"ComponentID\":\"EDINN\",\"ConfirmationCode\":\"Always\"},\"CreationDateTime\":\"2020-09-24T09:02:08\"},\"DataArea\":{\"Confirm\":{\"ResponseCriteria\":{\"ResponseExpression\":{\"@actionCode\":\"Accepted\"}}},\"BOD\":{\"Description\":\"Accepted 2020092304\",\"Note\":\"segment 10 of the order 2020092304 inserted successfully.\"}}}}",
"message":"Success"}

Reponse (error):
{"status":"0",
"data":"{\"ConfirmBOD\":{\"ApplicationArea\":{\"Sender\":{\"LogicalID\":\"20200924090448ZOZCENTRAL\",\"ComponentID\":\"EDINN\",\"ConfirmationCode\":\"Always\"},\"CreationDateTime\":\"2020-09-24T09:04:48\"},\"DataArea\":{\"Confirm\":{\"ResponseCriteria\":{\"ResponseExpression\":{\"@actionCode\":\"Rejected\"}}},\"BOD\":{\"Description\":\"Rejected 2020092304\",\"Note\":\"Must provide a equipment requirement (Equipment 8102 does not exist in database) (Operation cancelled by an error)\"}}}}",
"message":"Success"}

Example of uncoded work order data.


```
{
  "SyncProductionSchedule": {
    "ApplicationArea": {
      "Sender": {
        "ComponentID": "ERP",
        "ConfirmationCode": "ALWAYS"
      },
      "CreationDateTime": "2019-10-30T15:06:51.065224+00:00"
    },
    "DataArea": {
      "Sync": {
        "ActionCriteria": {
          "ActionExpression": { "@actionCode": "Add" }
        }
      },
      "ProductionSchedule": {
        "ID": "PS000001",
        "ProductionRequest": {
          "ID": "PR000001",
          "Description": "Request description",
          "Location": {
            "EquipmentID": "P10",
            "EquipmentElementLevel": "Site",
            "Location": {
              "EquipmentID": "L10",
              "EquipmentElementLevel": "ProductionLine"
            }
          },
          "StartTime": "2019-10-30T15:06:51.0662173+00:00",
          "EndTime": "2019-11-06T15:06:51.0662173+00:00",
          "RequestState": "Planned",
          "SegmentRequirement": {
            "ID": "SR10",
            "Description": "Segment description",
            "Location": {
              "EquipmentID": "P10",
              "EquipmentElementLevel": "Site",
              "Location": {
                "EquipmentID": "L10",
                "EquipmentElementLevel": "ProductionLine"
              }
            },
            "DocumentURL": "\\\\DOCS_FOLDER\\DOC.INF",
            "EarliestStartTime": "2019-10-30T15:06:51.0662173+00:00",
            "LatestEndTime": "2019-11-06T15:06:51.0662173+00:00",
            "Duration": "P0Y0M0DT1H36M0S",
            "ProductionParameter": {
              "ID": "OperationType",
              "Value": {
                "ValueString": "OP01",
                "DataType": "string"
              }
            },
            "ProductionParameter": {
              "ID": "WBS",
              "Value": {
```

```
            "ValueString": "1234",
            "DataType": "string"
          }
        },
        "PersonnelRequirement": [
          {
            "PersonnelClassID": "LB01",
            "PersonID": "WO001",
            "Quantity": {
              "QuantityString": "30.000",
              "DataType": "double",
              "UnitOfMeasure": "HUR"
            },
            "Location": {
              "EquipmentID": "CONAME",
              "EquipmentElementLevel": "Enterprise"
            }
          },
          {
            "PersonnelClassID": "LB01",
            "PersonID": "WO001",
            "Quantity": {
              "QuantityString": "1",
              "DataType": "double",
              "UnitOfMeasure": "UN"
            },
            "Location": {
              "EquipmentID": "CONAME",
              "EquipmentElementLevel": "Enterprise"
            }
          }
        ],
        "EquipmentRequirement": {
          "EquipmentClassID": "EC01",
          "EquipmentID": "EQ01",
          "Quantity": {
            "QuantityString": "1",
            "DataType": "double",
            "UnitOfMeasure": "UNIT"
          },
          "Location": {
            "EquipmentID": "CONAME",
            "EquipmentElementLevel": "Enterprise",
            "Location": {
              "EquipmentID": "L10",
              "EquipmentElementLevel": "PoductionLine"
            }
          }
        },
        "MaterialRequirement": [
          {
            "MaterialClassID": "MC01",
            "MaterialDefinitionID": "MT0001",
            "MaterialLotID": "LOT0001",
            "MaterialSubLotID": "SLOT01",
            "Location": {
              "EquipmentID": "SZ01",
              "EquipmentElementLevel": "StorageZone",
```

```
            "Location": {
              "EquipmentID": "SU01",
              "EquipmentElementLevel": "StorageUnit"
            }
          },
          "MaterialUse": "Produced",
          "Quantity": {
            "QuantityString": "100",
            "DataType": "double",
            "UnitOfMeasure": "UNIT"
          },
          "MaterialRequirementProperty": {
            "ID": "PROP01",
            "Value": {
              "ValueString": "VAL0001",
              "DataType": "string"
            }
          }
        },
        {
          "MaterialClassID": "MC02",
          "MaterialDefinitionID": "MT0002",
          "MaterialLotID": "LOT0002",
          "MaterialSubLotID": "SLOT02",
          "Location": {
            "EquipmentID": "SZ01",
            "EquipmentElementLevel": "StorageZone"
          },
          "MaterialUse": "Consumed",
          "Quantity": {
            "QuantityString": "100",
            "DataType": "double",
            "UnitOfMeasure": "UNIT"
          },
          "MaterialRequirementProperty": {
            "ID": "PROP02",
            "Value": {
              "ValueString": "VAL0002",
              "DataType": "string"
            }
          },
          "Any": {
            "Confirm": {
              "ValueString": "1",
              "DataType": "integer"
            }
          }
        }
      ],
      "SegmentRequirement": [
        {
          "ID": "Setup",
          "Duration": "PT2H0M0S"
        },
        {
          "ID": "Processing",
          "Duration": "PT60H0M0S"
        }
```

```
            ]
          }
        }
      }
    }
  }
}
```

Notes:

B2MML tags not present in this document will be ignored by the system, althought could be considered for future versions.

RequestState: Refers to the status of the scheduled order. Allowed values are:

- Preplanned: The order date and time and the allocation of resources are not final and should be planned or confirmed.
- Planned: The order date and time and the allocation of resources are final.
- Completed: The order has been completed in the ERP.

DocumentURL: Path to a document with information about the order.

Duration: If provided, the cycle time and the cycle quantity of the order will be calculated according to its value, otherwise, they will be inherited from the cycle time and the cycle quantity of the process.

Response example:

```
{
   "status":"0",
   "data":"{
      \"ConfirmBOD\":{
         \"ApplicationArea\":{
            \"Sender\":{\"ComponentID\":\"EDINN\",\"ConfirmationCode\":
            \"Always\"},
            \"CreationDateTime\":\"2019-11-12T11:38:16\"
         },
         \"DataArea\":{
            \"Confirm\":{
               \"ResponseCriteria\":{\"ResponseExpression\":{\"@actionCode\":
               \"Accepted\"}}
            },
            \"BOD\":{
               \"Description\":\"Accepted PR000001\",\"Note\":\"segment SR10
               of the order PR000001 inserted successfully.\"
            }
         }
      }
   }",
   "message":"Success"
}
```

Response example of a working order tha could not be inserted:

```
{
   "status":"0",
   "data":"{
      \"ConfirmBOD\":{
         \"ApplicationArea\":{
            \"Sender\":{\"ComponentID\":\"EDINN\",\"ConfirmationCode\":
            \"Always\"},
            \"CreationDateTime\":\"2019-11-12T11:38:16\"
         },
         \"DataArea\":{
            \"Confirm\":{
               \"ResponseCriteria\":{\"ResponseExpression\":{\"@actionCode\":
               \"Rejected\"}}
            },
            \"BOD\":{
               \"Description\":\"Rejected PR000001\",\"Note\":\"Must provide
               a equipment requirement (Equipment EQ01 does not exist in
               database) (Operation cancelled by an error)\"
            }
         }
      }
   }",
   "message":"Success"
}
```


POST /schedule

This method creates a scheduled order. The user must have the scheduler permission.

Request parameters (for additional information please review the schedule edition):

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **area [mandatory]:** area to which the order belongs.
- **process:** process to which the order is assigned.
- **date [mandatory]**: scheduled start date and time.
- **order [mandatory]**: order code.
- **result [mandatory]**: result to be produced.
- **target [mandatory]**: target quantity of the order.
- **duedate [mandatory]**: expected end date and time of the order.
- **duration**: duration of the order.
- **comment:** text that describes the order.

Reponse:

- **status**: returns "0" on success, and an error code in any other case.

- **data**: returns the link to the request help on error.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"post",
"url":"schedule",
"data":"company=7362919493&session=1FrqhchVXQoAgpxUbdKHWQkILZjMMAV0cesiSMmaRjwFvaWLwrpLyxJ6O9Zo&area=7100&process=7101&date=20201101000000&duedate=20201108000000&order=20201102&result=BA0932535201&target=100&comment=Bolsas25x35"}

Reponse (success):
{"status":0,
"data":null,
"message":"Success"}

Reponse (error):
{"status":"1000",
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Schedule_Config.html'>for additional information follow this link<\/a>",
"message":"Some parameters are missing"}

DELETE /schedule/{id}

This method deletes a scheduled order identified by its unique code in edinn. The user must have scheduler permission.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Request attributes:

- **id [mantatory]**: id of the working order.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the link to the request help on error.
- **message**: "Success" or a descriptive information of the error in any other case.

Ejemplo:

Request:
{"type":"delete",
"url":"schedule/20201009110800AABCENTRAL",
"data":"company=7362919493&session=1FrqhchVXQoAgpxUbdKHWQkILZjMMAV0cesiSMmaRjwFvaWLwrpLyxJ6O9Zo"}

Reponse (success):

{"status":"0",
"data":null,
"message":"Success"}

Reponse (error):
{"status":1007,
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Schedule_Config.html'>for additional information follow this link<\/a>",
"message":"Session not stablished"}

DELETE /schedule

This method removes a scheduled orders identified by its process and code. The user must have the scheduler permission.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **process [mandatory]:** the process where is the order.
- **order [mandatory]:** the code of the order.

Reponse:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the link to the request help on error.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"delete",
"url":"schedule/20201009110800AABCENTRAL",
"data":"company=7362919493&session=1FrqhchVXQoAgpxUbdKHWQkILZjMMAV0cesiSMmaRjwFvaWLwrpLyxJ6O9Zo&process=TT00&order=20201009110800AAB"}

Reponse (success):
{"status":"0",
"data":null,
"message":"Success"}

Reponse (error):
{"status":1007,
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Schedule_Oper.html'>for additional information follow this link<\/a>",
"message":"Session not stablished"}

PATCH /schedule/{id}/reschedule

This method is used to change the planned start date, end date, quantity, and / or duration of a scheduled order. User must have rescheduler permission.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **date**: scheduled start date and time.
- **target**: target quantity of the order.
- **duedate**: expected end date and time of the order.
- **duration**: duration of the order.

Request attributes:

- **id [mantatory]**: id of the working order.

Reponse:

- **status**: devuelve "0" en caso de éxito y un código de error en cualquier otro caso.
- **data**: devuelve el enlace a la ayuda de la solicitud en caso de error.
- **message**: "Success" o una información descriptiva del error en cualquier otro caso.

Example:

Request:
{"type":"patch",
"url":"schedule/20211105171958AACINIT/reschedule",
"data":"company=7362919493&session=1FrqhchVXQoAgpxUbdKHWQkILZjMMAV0cesiSMmaRjwFvaWLwrpLyxJ6O9Zo&date=20201009113000&duedate=20201009163000&target=950&duration=3000"}

Reponse (success):
{"status":"0",
"data":null,
"message":"Success"}

Reponse (error):
{"status":1007,
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Schedule_Config.html'>for additional information follow this link<\/a>",
"message":"Session not stablished"}

PUT /schedule/{id}

This method replaces all the parameters of a scheduled order. The user must have the scheduler permission.

Request parameters (for additional information please review the schedule edition):

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

- **area [mandatory]:** area to which the order belongs.
- **process:** process to which the order is assigned.
- **date [mandatory]**: scheduled start date and time.
- **order [mandatory]**: order code.
- **result [mandatory]**: result to be produced.
- **target [mandatory]**: target quantity of the order.
- **duedate [mandatory]**: expected end date and time of the order.
- **duration**: duration of the order.
- **comment:** text that describes the order.

Request attributes:

- **id [mantatory]**: id of the working order.

Reponse:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the link to the request help on error.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Request:
{"type":"put",
"url":"schedule/20211027133604AABINIT",
"data":"company=7362919493&session=1FrqhchVXQoAgpxUbdKHWQkILZjMMAV0cesiSMmaRjwFvaWLwrpLyxJ6O9Zo&area=7100&process=7101&date=20201101000000&duedate=20201108000000&order=20201102&result=BA0932535201&target=100&comment=Bolsas25x35"}

Reponse (success):
{"status":0,
"data":null,
"message":"Success"}

Reponse (error):
{"status":"1000",
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Schedule_Config.html'>for additional information follow this link<\/a>",
"message":"Some parameters are missing"}

Operation

GET /schedule/active

Request parameters:

- **company [mandatory]:** The company name or id.
- **session [mandatory]:** The Id of the active session.
- **process:** process to which the order is assigned.

Response:

- **status:** returns "0" on success, and an error code in any other case.
- **data:** returns list of work orders that match the indicated criteria on success or the link to the request help on error.
- **message:** "Success" or a descriptive information of the error in any other case.

Example:

Request:
/schedule/active
{
"session": 01qgmfcZmZ4nqg2eNGpxY4JlpmhJuKw1H31p5rtD1a112SSK8S8odROFlKgn
"company": DEMO04
}
Reponse (success):

{
"status":0,

"data":[
"OrderWO": "85767261",
"ProdTotalWO": "60",
"ProdTheorTotalWO": "6500",
"TimeSetupWO": "0:00:00",
"TimeTeardownWO": "0",
"DueDateWO": "18/04/2017 5:10:09",
"StartedWO": "15/04/2017 0:33:00",
"CommentWO": "",
"PriorityWO": "0",
"EstimationWO": "53816",
"ProdGoodWO": "60",
"ProdBadWO": "0",
"ProdReworkWO": "0",
"ProdRemainingWO": "6440"
]

"message":"Success
}

Quality

This section handles Quality (SPC).

Configuration

This section handles Quality (SPC) configuration.

POST /quality/config (ISA-95 compliant, for more information please read the B2MML document)

This method creates the quality configuration using json encodad B2MML data. The user must have the administrator role.

Request parameters (for additional information please review the SPC configuration):

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **data [mandatory]**: the json encoded B2MML data defining the SPC charcteristics.

Example:

```
{
   "SyncMaterialInformation":
   {
      "ApplicationArea":
      {
         "Sender":{"ComponentID":"ERP","ConfirmationCode":"Always"},
         "CreationDateTime":"2019-04-11T12:18:59"
      },
      "DataArea":
      {
         "Sync":{"ActionCriteria":{"ActionExpression":
         {"@actionCode":"Change"}}},
         "MaterialInformation":
         {
            "ID":"8561",
            "Description":"8561 1.4L",
            "QAMaterialTestSpecification":
            {
               "Name":"PR1",
               "Description":"PROGRAM 1",
               "TestedMaterialDefinitionProperty":
               {
                  "MaterialDefinitionID":"8561",
                  "PropertyID":"PR1M1",
                  "Description":"MESARURE 1",
                  "PublishedDate":"2019-04-11T12:18:59",
                  "SizeDb":"1",
                  "Res":"0.01",
                  "CPK":"0",
                  "Link":[],
                  "Limits":
                  {"UAL":"1","UL":"0.2","UCL":"0.18","LCL":"0.12","LL":"0.1",
                  "LAL":"0"},
                  "Source":{"ID":"MAN1","Description":"MANOMETER
```

```
                    1","UnitOfMeasure":"KG","Res":"0.001"},
                    "AutoLimit":"0",
                    "EquipmentID":"A0L1",
                    "Every":"60",
                    "Color":"255"
                }
            }
        }
    }
}
```

Response example:

```
{
   "status":"0",
   "data":"{
      \"ConfirmBOD\":{
         \"ApplicationArea\":{
            \"Sender\":{\"ComponentID\":\"EDINN\",\"ConfirmationCode\":
            \"Always\"},
            \"CreationDateTime\":\"2019-11-12T12:07:32\"
         },
         \"DataArea\":{
            \"Confirm\":{
               \"ResponseCriteria\":{\"ResponseExpression\":{\"@actionCode\":
               \"Accepted\"}}
            },
            \"BOD\":{
               \"Description\":\"Accepted 8561\",\"Note\":
               \"MaterialInformation for 8561 processed successully\"
            }
         }
      }
   }",
   "message":"Success"
}
```

Tree

Tree gets the schema of areas and processes.

GET /tree

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **version (1/2)**: the version of the request.
- **device**: id of the device from wich the request is made.

- **showoee (0/1):** shows the OEE of areas and processes: oee_actual, oee_target_yellow, oee_target_green. (for version = 1)
- **showalerts (0/1):** shows the notification alerts for each process: notify_pr, notify_ti. (for version = 1)
- **showdashboard (0/1)**: shows the first dashboard configured for each process. (for version = 1)
- **area**: makes the request from the given area, if empty, gets the whole tree. (for version = 2)
- **period**: indicates the perdiod to show color and alerts from the period list; for more information about period list go to periods. (for version = 2)
- **type**: indicates the ratio type to show; the available ratios list is: 0 - OEE, 1 - Operative OEE, 2 - OCE, 3 - OPCE. (for version = 2)
- **recursive**: indicates if the request gets the sub-areas and processes recursively or gets only the first level from the indcated area or from root if area is empty. (for version = 2)

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the schema of areas and processes on success or the link to the request help on error.
- **message**: "Success" or a descriptive information of the error in any other case.

Example (version = 1):

Request:

{"type":"get",
"url":"tree",
"data":"company=testco&session=neQXoKWYLALJ1yVjr5d2ANOhZrwp52p5qeyM7WdzAi96rJ5QKmGHslyF5tzV&version=1"}

Reponse (success):

{
  "status": "0",
  "data": {
    "areas": [
      {
        "id": "100",
        "desc": "NAVE1",
        "processes": [
          {
            "id": "101",
            "desc": "RS101",
            "type": "0",
            "shifts": []
          },

```
        {
          "id": "102",
          "desc": "RS102",
          "type": "0",
          "shifts": []
        },
        {
          "id": "103",
          "desc": "RS103",
          "type": "0",
          "shifts": []
        }
      ]
    },
    {
      "id": "200",
      "desc": "NAVE2",
      "processes": [
        {
          "id": "201",
          "desc": "RS201",
          "type": "0",
          "shifts": []
        },
        {
          "id": "202",
          "desc": "RS202",
          "type": "0",
          "shifts": []
        }
      ]
    }
  ]
 },
 "message": "Success"
}
```


Reponse (error):


```
{"status":"1000",
"data":"<a href='http:\/\/127.0.0.1:8080\edinnM2\/help\/en\/API_Tree.html'>for additional information follow this link<\/a>",
"message":"Some parameters are missing"}
```


Example (version = 2):

Request:


```
{"type":"get",
"url":"tree",
"data":"version=2&company=7362919493&session=Bm79Sa1HHxGG3N9MrOUogBWkCf5Tc1B6sK1ebwaz2xyqGdUpQF2WQeSZgvPa&device=INIT&area=&period=1&type=0&recursive=1"}
```


Reponse (success):


```
{
 "status": "0",
 "data": {
  "parentids": [
   "|",
   "||7100",
```

```
    "||7100"
  ],
  "childids": [
    "7100",
    "7101",
    "7102"
  ],
  "childdescs": [
    "BOLSAS",
    "MACH 7101",
    "MACH 7102"
  ],
  "childtypes": [
    0,
    1,
    1
  ],
  "areas": [
    {
      "id": "7100",
      "description": "BOLSAS",
      "type": 0,
      "elements": "7101,7102",
      "formula": "\"7101\"+\"7102\"",
      "y0": 0,
      "g0": 0,
      "y1": 0,
      "g1": 0,
      "y2": 0,
      "g2": 0,
      "y3": 0,
      "g3": 0,
      "y4": 0,
      "g4": 0,
      "y5": 0,
      "g5": 0,
      "y6": 0,
      "g6": 0,
      "y7": 0,
      "g7": 0,
      "background": "",
      "width": 0,
      "height": 0,
      "wbs": "",
      "driverip": "",
      "scheduled": 1,
      "color": 4517779,
      "label": "7100-BOLSAS:rn0%>=0%",
      "oee": {
        "Id": "7100",
        "OEE0TT": 49492,
        "OEE1TT": 7200,
        "OEE2TT": 28800,
        "OEE3TT": 172800,
        "OEE4TT": 0,
        "OEE5TT": 0,
        "OEE0A": 0,
        "OEE1A": 0,
```

```
      "OEE2A": 0,
      "OEE3A": 0.1335,
      "OEE4A": 0,
      "OEE5A": 0,
      "OEE0O": 0,
      "OEE1O": 0,
      "OEE2O": 0,
      "OEE3O": 0.143,
      "OEE4O": 0,
      "OEE5O": 0,
      "OEE0P": 1,
      "OEE1P": 1,
      "OEE2P": 1,
      "OEE3P": 0.2062,
      "OEE4P": 0,
      "OEE5P": 0,
      "OEE0Q": 1,
      "OEE1Q": 1,
      "OEE2Q": 1,
      "OEE3Q": 0.9959,
      "OEE4Q": 0,
      "OEE5Q": 0,
      "OEELastUpdate": "20210610125222"
    },
    "state": {
      "Id": "7100",
      "Video": "",
      "Since": "",
      "Sch_Col": 7,
      "Pro_Col": 0,
      "Ti0_Col": 7,
      "Ti1_Col": 7,
      "Ti2_Col": 6,
      "Co0_Col": 0,
      "Co1_Col": 0,
      "Co2_Col": 0,
      "Aut_Col": 0,
      "SPC0_Col": 0,
      "SPC1_Col": 0
    },
    "graphic": {
      "": "[]"
    }
  }
],
"processes": [
  {
    "order": 1,
    "id": "7101",
    "desc": "MACH 7101",
    "calculation": "BOLSAS",
    "cycleot": 0,
    "cyclet": 15,
    "maxcyclet": 15,
    "cycleq": 0.07575,
    "npcauto": 0,
    "unit": "Ml",
    "minopers": 0,
```

```
"opers": 1,
"y0": 50,
"g0": 60,
"y1": 80,
"g1": 90,
"y2": 90,
"g2": 95,
"y3": 60,
"g3": 70,
"y4": 10,
"g4": 5,
"y5": 240,
"g5": 120,
"y6": 60,
"g6": 30,
"y7": 50,
"g7": 50,
"trace": 0,
"next_processes": "",
"resps": "9999",
"stshort": 120,
"prshort": 60,
"microstop": "MIC",
"url": "",
"lockrec": 480,
"man": 13,
"genpstart": 0,
"interign": 0,
"limprodtosche": 1,
"allowconf": 0,
"type": 0,
"requirenotify": 0,
"wbs": "",
"notifyratios": 0,
"represent": 0,
"excessordefect": 0,
"schedulefields": 0,
"img": "",
"imgurl": "http:///edinnm2/img/7362919493/Mach_7101_",
"operationtime": 216000,
"color": 5066239,
"label": "7101-MACH 7101rnANONIMO (0,150)rn10/6 18:20 -> 10/10 22:00rn0%<51%",
"tooltip": "",
"oee": {
  "Id": "7101",
  "OEE0TT": 24742,
  "OEE1TT": 3600,
  "OEE2TT": 14400,
  "OEE3TT": 86400,
  "OEE4TT": 0,
  "OEE5TT": 0,
  "OEE0A": 0,
  "OEE1A": 0,
  "OEE2A": 0,
  "OEE3A": 0.0531,
  "OEE4A": 0,
  "OEE5A": 0,
  "OEE0O": 0,
```

```
    "OEE1O": 0,
    "OEE2O": 0,
    "OEE3O": 0.0612,
    "OEE4O": 0,
    "OEE5O": 0,
    "OEE0P": 1,
    "OEE1P": 1,
    "OEE2P": 1,
    "OEE3P": 0.3195,
    "OEE4P": 0,
    "OEE5P": 0,
    "OEE0Q": 1,
    "OEE1Q": 1,
    "OEE2Q": 1,
    "OEE3Q": 0.9918,
    "OEE4Q": 0,
    "OEE5Q": 0,
    "OEELastUpdate": "20210610125222"
  },
  "state": {
    "Id": "7101",
    "NextPiece": "",
    "Limit": "",
    "Id_TUnext": "",
    "Id_TUlimit": "",
    "Id_TUSince": "20210610182029",
    "Id_TU": "6105",
    "PNSince": "20210610182029",
    "PN": "21000764-10",
    "Id_ProdSince": "20210610182029",
    "Id_Prod": "BA1156535200VM",
    "Sch_Col": 7,
    "Pro_Col": 0,
    "Ti0_Col": 7,
    "Ti1_Col": 7,
    "Ti2_Col": 6,
    "Co0_Col": 0,
    "Co1_Col": 0,
    "Co2_Col": 0,
    "Aut_Col": 0,
    "SPC0_Col": 0,
    "SPC1_Col": 0,
    "GQT_Col": 0,
    "QuickUpd": 0,
    "QUDateTime": "",
    "CycleT": 82,
    "CycleQ": 1,
    "SPCReq": "",
    "LastSentPR": "",
    "LastResultId": "20210610182007ABRINIT",
    "LastResultWhen": "20210610182029",
    "LastResultPaused": 2,
    "LastResultId_Paused": "",
    "LastResultModified": "",
    "TUTypeCurrent": 3,
    "AvgCycleT": 15,
    "AvgCycleQ": 0.0000192731
  }
```

```
  },
  {
    "order": 2,
    "id": "7102",
    "desc": "MACH 7102",
    "calculation": "BOLSAS",
    "cycleot": 0,
    "cyclet": 15,
    "maxcyclet": 15,
    "cycleq": 0.07575,
    "npcauto": 0,
    "unit": "Ml",
    "minopers": 0,
    "opers": 1,
    "y0": 50,
    "g0": 60,
    "y1": 80,
    "g1": 90,
    "y2": 90,
    "g2": 95,
    "y3": 60,
    "g3": 70,
    "y4": 10,
    "g4": 5,
    "y5": 240,
    "g5": 120,
    "y6": 60,
    "g6": 30,
    "y7": 50,
    "g7": 50,
    "trace": 0,
    "next_processes": "",
    "resps": "9999",
    "stshort": 120,
    "prshort": 60,
    "microstop": "MIC",
    "url": "",
    "lockrec": 480,
    "man": 13,
    "genpstart": 0,
    "interign": 0,
    "limprodtosche": 1,
    "allowconf": 0,
    "type": 0,
    "requirenotify": 0,
    "wbs": "",
    "notifyratios": 0,
    "represent": 0,
    "excessordefect": 0,
    "schedulefields": 0,
    "img": "",
    "imgurl": "http:///edinnm2/img/7362919493/Mach_7102_",
    "operationtime": 216000,
    "color": 5066239,
    "label": "7102-MACH 7102rnCOFB (507,091)rn26/5 14:34 -> 31/12 22:00rn0%<51%",
    "tooltip": "",
    "oee": {
      "Id": "7102",
```

```
    "OEE0TT": 24750,
    "OEE1TT": 3600,
    "OEE2TT": 14400,
    "OEE3TT": 86400,
    "OEE4TT": 0,
    "OEE5TT": 0,
    "OEE0A": 0,
    "OEE1A": 0,
    "OEE2A": 0,
    "OEE3A": 0.2139,
    "OEE4A": 0,
    "OEE5A": 0,
    "OEE0O": 0,
    "OEE1O": 0,
    "OEE2O": 0,
    "OEE3O": 0.2139,
    "OEE4O": 0,
    "OEE5O": 0,
    "OEE0P": 1,
    "OEE1P": 1,
    "OEE2P": 1,
    "OEE3P": 0.0928,
    "OEE4P": 0,
    "OEE5P": 0,
    "OEE0Q": 1,
    "OEE1Q": 1,
    "OEE2Q": 1,
    "OEE3Q": 1,
    "OEE4Q": 0,
    "OEE5Q": 0,
    "OEELastUpdate": "20210610125230"
  },
  "state": {
    "Id": "7102",
    "NextPiece": "",
    "Limit": "",
    "Id_TUnext": "",
    "Id_TUlimit": "",
    "Id_TUSince": "20210610060000",
    "Id_TU": "FAI",
    "PNSince": "20210526143425",
    "PN": "20001652-10",
    "Id_ProdSince": "20210526143425",
    "Id_Prod": "BB1828611101/29372",
    "Sch_Col": 7,
    "Pro_Col": 0,
    "Ti0_Col": 7,
    "Ti1_Col": 7,
    "Ti2_Col": 6,
    "Co0_Col": 0,
    "Co1_Col": 0,
    "Co2_Col": 0,
    "Aut_Col": 0,
    "SPC0_Col": 0,
    "SPC1_Col": 0,
    "GQT_Col": 0,
    "QuickUpd": 0,
    "QUDateTime": "",
```

```
        "CycleT": 324.518,
        "CycleQ": 1,
        "SPCReq": "",
        "LastSentPR": "",
        "LastResultId": "20210609180532AGQCENTRAL",
        "LastResultWhen": "20210609180531",
        "LastResultPaused": 0,
        "LastResultId_Paused": "",
        "LastResultModified": "",
        "TUTypeCurrent": 3,
        "AvgCycleT": 15,
        "AvgCycleQ": 0.0000165072
      }
    }
  ]
 },
 "message": "Success"
}
```


Reponse (error):

{"status":"1000",
"data":"<a href='http:\/\/127.0.0.1:8080\/edinnM2\/help\/en\/API_Tree.html'>for additional information follow this link<\/a>",
"message":"Some parameters are missing"}

Lang

Lang gets the list of language.

GET /lang

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **target:** To get the language of a device.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the schema of areas and processes.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

Content-Type: application/json

Request: /lang?
company=demo01&session=LebEWTNf8GEhxfQeeGbBmVUlxmUlDLVdEoeVYrnWNzkdRXKx

```
mgCNc92wnSjA
Reponse (success):
{
  "status": "0",
  "data": {
      "langs": [
       {
          "id": "About",
          "Desc_de": "Etwa",
          "Desc_en": "About",
          "Desc_es": "Acerca de",
          "Desc_it": "Di",
          "Desc_pt": "Acerca de",
          "Desc_sk": "o",
          "Desc_zh": ""
       },
      ...
  "message": "Success"
}
```

Registry

This section handles registry messages. You can create messages in the registry.

POST /registry

This method creates a new registry message. The user must have the registry role.

Request parameters:

- **company [mandatory]**: The company name or id.
- **session [mandatory]**: the Id of the active session.
- **responsible:** responsible or addressee of the message.
- **datetime:** date and time of the event that gnerates the message.

- **level [mandatory]:** one of the event codes from the event list.
- **process:** process affected by the message.
- **message:** message content.
- **checked:** indicates if the message has been viewed.
- **link:** link to the record related with this message.
- **popup:** indicates if this message triggers a popup window.

Response:

- **status**: Returns "0" on success, and an error code in any other case.
- **data**: empty on success, or an error information in case of error.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Request:

```
{
 "company": "demo01",
 "session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
 "responsible": "esdemo",
 "datetime": "20190606131125",
 "level": "1",
 "process": "A0L1",
 "message": "Fallo en la etiquetadora",
 "checked": "0",
 "link": "TSTU20190606131125ADFCENTRAL",
 "popup": "0",
}
```

Response:

```
{
"status": 0,
"data": "",
```

```
"message": "Success"
}
```

Console

Services

This section handle the services management.

POST /console/ServiceHandler

This method is used to create, start, stop and delete any of the edinn services.

Request parameters:

- **action [mandatory]**: which action you want to do: 0-create, 1-start, 2-stop, 3-delete.
- **serviceName [mandatory]**: the Id of the service.
- **serviceDisplayName:** the name of the service.
- **company [mandatory]**: the company Id.
- **host [mandatory]**: the IP or name of the host.
- **account [mandatory]**: the user name of the windows system.
- **password [mandatory]**: the windows system user password.
- **directory [mandatory if action is 0]**: the directory of the service executable.
- **startType [mandatory if action is 0]**: determines if the start method is manual or atomatic.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns "0" on succes, and an error description in any other case.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

```
Content-Type: application/json
Request: /console/ServiceHandler?
action=1&company=0000000001&serviceName=edinnM2_Central_0000000001&session=TI
lB31jdRNBSEp4IwQT9QiwzX54q6LUsEztbwYiJXGsOoE0YGCUbAB4urUI8&host=192.168.1.100
&account=admin=1234

Reponse (success):
{
   "status": "0",
   "data": "0",
   "message": "Success"
}
```

GET /console/ServiceHandler

This method is used to get the status or the configuration of the service.

Request parameters:

- **action [mandatory]**: which action you want to do: 0-status, 1-consfiguration.
- **serviceName [mandatory]**: the Id of the service.
- **company [mandatory]**: the company Id.
- **host [mandatory]**: the IP or name of the host.
- **account [mandatory]**: the user name of the windows system.
- **password [mandatory]**: the windows system user password.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: return the code of the status on succes, and an error description in any other case.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

```
Content-Type: application/json
Request: /console/ServiceHandler?
user=LocalSystem&password=&host=&company=0000000001&session=O1ggOxp8AbNdS3gOd
NUxri7XjzXIuSIIO5LIEpBZDddsE9BR1EFjUpg57qwB&serviceName=edinnM2_Central_00000
00001&action=0

Reponse (success):
{
   "status": "0",
   "data": "0",
   "message": "Success"
}
```

Tasks

This section handle the services management.

POST /console/TaskHandler

This method is used to execute any of the edinn tasks.

Request parameters:

- **taskName [mandatory]**: The name of the task: ReportsCache, ReportsCacheUnschedule, Optimizer, Autocalendar, Updateserver, Bigdata.
- **company [mandatory]**: the company Id.
- **host [mandatory]**: the IP or name of the host.
- **port [mandatory]**: the data base port.
- **directory [mandatory]**: the directory of the service executable.
- **extParams**: aditional parameters separated by "," on Autocalendar and Updateserver. Separated by blankspace on Bigdata.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns a boolean based on the result of the transaction.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

```
Content-Type: application/json
Request: /console/TaskHandler?
taskName=BIGDATA&company=0000000001&host=192.168.133.1&port=10000&directory=Y
:
\VB6\edinnM2\Console&session=9hCSX9oBWVbeilKi0L7zjKDPq0hSusjPII1CVCQIdgGdTYv2
IQ9wijp5Q2km&extParams=192.168.133.1 10000 0000000001 0 99999 999 A0L1
20170301070004 18991230000000 1 1 0

Reponse (success):
{
   "status": "0",
   "data": True,
   "message": "Success"
}
```

Registry

This section handle the registry management.

POST /console/RegistryHandler

This method is used to save keys in the registry.

Request parameters:

- **subKey [mandatory]**: name of the registry key.
- **value [mandatory]**: value of the key.
- **keyPath [mandatory]:** path of the key.
- **company [mandatory]**: the company Id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns True on succes.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

```
Content-Type: application/json
Request: /console/RegistryHandler?
subKey=SrvDefault&keyPath=SOFTWARE\edinn\edinnM2\Supervisor_0000000001&compan
y=0000000001&value=TEST&session=Xl4AAZNHFIYrWAp9YhcpzGKOuM6ZqPAQbKx88It9V5lMr
xcMZtVhLCmxwYEY

Reponse (success):
{
   "status": "0",
   "data": True,
   "message": "Success"
}
```

GET /console/RegistryHandler

This method is used to load a registry value or check if exist.

Request parameters:

- **subKey [mandatory]**: name of the registry key.
- **keyPath [mandatory]:** path of the key.
- **company [mandatory]**: the company Id.

- **session [mandatory]**: the Id of the active session.
- **operation [mandatory]**: the operation value: 0-Load, 1-Check.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: return the value of the key on the load and a boolean on the check.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

```
Content-Type: application/json
Request: /console/ServiceHandler?
operation=0&subKey=RstProcAfter&keyPath=SOFTWARE\edinn\edinnM2\Supervisor_000
0000001&company=0000000001&session=Acb9bacCtxGHZXJN6EMyr4wDALkJ1zOOnWEGPDFHRy
8P8XoSNOD2Qn3CRe47

Reponse (success):
{
   "status": "0",
   "data": "TEST",
   "message": "Success"
}
```

WebServer

This section handle the WebServer management.

POST /console/WebServerHandler

This method is used to modify certain php.ini values.

Request parameters:

- **params [mandatory]:** parameters to save on the web server separated by "|-|" : TimeZone|-|MaxLifeTime|-|PHPiniPath.
- **company [mandatory]**: the company Id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns True on succes.
- **message**: "success" or a descriptive information of the error in any other case.

Example:

```
Content-Type: application/json
Request: /console/WebServerHandler?
company=0000000001&params=Europe/Madrid|-|1400|-|
localhost\websrv\php\php.ini|-|
&session=Xl4AAZNHFIYrWAp9YhcpzGKOuM6ZqPAQbKx88It9V5lMrxcMZtVhLCmxwYEY

Reponse (success):
{
   "status": "0",
   "data": True,
   "message": "Success"
}
```

GET /console/WebServerHandler

This method is used to load some php.ini values.

Request parameters:

- **params [mandatory]:** php ini path.
- **company [mandatory]**: the company Id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns some php.ini parameters separated by "|-|": TimeZone|-| MaxLifeTime|-|PHPiniPath
- **message**: "success" or a descriptive information of the error in any other case.

Example:

```
Content-Type: application/json
Request: /console/WebServerHandler?
company=0000000001&params=C:\Program
Files\edinn\edinnM2\Server\websrv\php\php.ini&session=Ptpl2mrxTKBi9JJWkYT5glu
jIRBfmMvO1ZAK3Gkk8dHt7hKVjgkOVTkoI9R5

Reponse (success):
{
   "status": "0",
   "data": Europe/Madrid|-|4444|-|* Read from C:\Program
   Files\edinn\edinnM2\Server\websrv\php\php.ini|-|0,
   "message": "Success"
}
```

Big Data

This section handles Big Data.

Operation

This section handles the queries to Big Data.

GET /bigdata

This method returns the comined informacion about a process.

Request parameters:

- **company [mandatory]**: The company name or id.
- **session [mandatory]**: The Id of the active session.
- **datefrom [mandatory]**: The starting date of the selection.
- **dateto [mandatory]**: The ending date of the selection.
- **process**: The Id of the process of whose data you want to show. If not indicated it shows all processes.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the big data inormation about the process.
- **message**: "success" or a descriptive information of the error in any other case.

Ejemplo:


```
Request:
{"type":"get",
"url":"bigdata",
"data":"company=testco&session=GfZ0YWyyJCRTeRdQvOcvSTJYBzmnSCS70q9rQGb8zgrCykt7BMDhoRiRd16K&process=7201&datefrom=20201101060000&dateto=20201101140000"}

Reponse (success):
{"status":"0","data":
[{"`DateTime`":"20201101060000","`Mach_Id`":"7201","`PN`":"","`Prod_Id`":"002R","`TSPR_Id_Author`":"","`TSPR_Created`":"","`TSPR_Id_Mod`":"","`TSPR_Modified`":"","`ProdType`":"0","`TiUs_Id`":"FAI","`Secs`":"28800","`TSTU_Id_Author`":"CENTRAL","`TSTU_Created`":"20201101162340","`TSTU_Id_Mod`":"CENTRAL","`TSTU_Modified`":"20201102131459","`TUType`":"3","`CycleT`":"15","`CycleQ`":"15","`Avai_Time`":"28800","`Prod_Time`":"0","`Total_Prod`":"0","`Theo_Prod`":"28800","`Tgt_Prod`":"0","`Green_Prod`":"14774,4000102997","`Yellow_Prod`":"10367,999879837","`Scrap_Prod`":"0","`Rework_Prod`":"0","`Good_Prod`":"0","`Availa
```

```
bility`":"0","`Speed`":"0","`Quality`":"1","`OEE`":"0","`Shif_Id`":"4","`Oper
Names`":"","`TSPE_Id_Author`":"","`TSPE_Created`":"","`Team_Id`":"","`TSTE_Id
_Author`":"","`TSTE_Created`":"","`TSPR_Comment`":"","`TSTU_Comment`":"","`TS
PR_Id`":"","`TSTU_Id`":"20201101162340AAOCENTRAL","`TSPE_Id`":"","`TSTE_Id`":
"","`Warn`":"","`Created`":"20201102135520"}],"message":"Success"}
```


Reponse (error):


```
{"status":1012,
"data":"<a href='http:\/
\/127.0.0.1:8080\/edinnM2\/help\/en\/API_processes_state.html'>for additional
information follow this link<\/a>",
"message":"Process not found"}
```


Let us see the description of each field:

- **DateTime**: date and time of the record (in inverted format YYYYMMDDhhmmss).
- **Mach_Id**: process identifier.
- **PN**: identifier (OrderId) of the work order.
- **Prod_Id**: result identifier.
- **TSPR_Id_Author**: identifier of the resource which created the results record.
- **TSPR_Created**: date and time of the creation of the results record (in inverted format YYYYMMDDhhmmss).
- **TSPR_Id_Mod**: identifier of the resource which modified the results record.
- **TSPR_Modified**: date and time of the modification of the results record (in inverted format YYYYMMDDhhmmss).
- **ProdType**: type of the result.
- **TiUs_Id**: identifier of the status.
- **Secs**: seconds passed since previous record.
- **TSTU_Id_Author**: identifier of the resource which created the status record.
- **TSTU_Created**: date and time of the creation of the status record (in inverted format YYYYMMDDhhmmss).
- **TSTU_Id_Mod**: identifier of the resource which modified the status record.
- **TSTU_Modified**: date and time of the modification of the status record (in inverted format YYYYMMDDhhmmss).
- **TUType**: type of the status.
- **CycleT**: time in seconds per cycle of production.
- **CycleQ**: quantity of results obtained every CycleT seconds.
- **Avai_Time**: available time to produce in seconds.
- **Prod_Time**: production time.
- **Total_Prod**: total results.
- **Theo_Prod**: results that could have been produced with the available time [(CycleQ / CycleT) * Avai_Time].
- **Tgt_Prod**: results that could have been produced with the production time [(CycleQ / CycleT) * Prod_Time].
- **Green_Prod**: first (green) target of results [Theo_Prod * 1er Obj. OEE].
- **Yellow_Prod**: second (yellow) target of results [Theo_Prod * 2o Obj. OEE].
- **Scrap_Prod**: scrap results.
- **Rework_Prod**: reworked results.
- **Good_Prod**: good results.
- **Availability**: percentage (up to 1) of availability.

- **Speed**: percentage (en tanto por 1) of speed.
- **Quality**: percentage (en tanto por 1) of quality.
- **OEE**: percentage (en tanto por 1) of OEE.
- **Shif_Id**: shift identifier.
- **OperNames**: identifiers, separated by commas, of the resources that were working.
- **TSPE_Id_Author**: identifier of the resource which created the IN&OUT record of resources.
- **TSPE_Created**: date and time of the creation of the IN&OUT record of resources (in inverted format YYYYMMDDhhmmss).
- **Team_Id**: identifier of the team of resources.
- **TSTE_Id_Author**: identifier of the resource which created the IN&OUT record of team of resources.
- **TSTE_Created**: date and time of the creation of the IN&OUT record of team of resources (in inverted format YYYYMMDDhhmmss).
- **TSPR_Comment**: comment inserted in the result.
- **TSTU_Comment**: comment inserted in the status.
- **TSPR_Id**: internal identifier of the results record.
- **TSTU_Id**: internal identifier of the status record.
- **TSPE_Id**: internal identifier of the IN&OUT resouces record.
- **TSTE_Id**: internal identifier of the IN&OUT team of resouces record.
- **Warn**: warnings generated by the system for this record.
- **Created**: date and time of the creation of this record (in inverted format YYYYMMDDhhmmss).

Fields

This section handles Fields.

Configuration

> **BETA:** This functionality is under development and there may be changes not reflected in this help.

This section handles the fields configuration.

GET fields/{id}/config

This method returns the list of fields.

Request attribute:

- **id [mandatory]**: the Id of the field.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **module [mandatory]:** indicates the module where is the field you want to get. 0: results, 1: autocontrol.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the properties of the fields.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Content-Type: application/json

Request: /fields/config?
company=edinn&session=0PY4RhPbK7YW5sbRXiwASrA2HEeCP0eGFzunnfiEJYWrp4whnU
11FLESeYpB&mod=0&idfield=S0000

Reponse (success):


```
{
  "status": "0",
  "data": [
    {
      "id": "20210520092950AAC",
      "id2": "S0000",
      "var": "1a2s1",
      "desc": "testApi2",
      "values": "1a2s",
      "ftype": "0",
      "mandat": "0",
      "id_comb": "",
      "io": "0",
      "id_calc": "",
      "fclass": "0",
      "maxlength": "255",
```

```
    "calculated": "0",
    "locked": "0",
    "inherits": "0",
    "stock": "0"
  },
 ],
 "message": "Success"
}
```


POST /fields/config

This method creates a new field. The user must have the administrator role.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **parent [mandatory]**: the Id of the result or autocontrol.
- **variable [mandatory]**: internal name of the field. Allows to carry out statistical studies, links with algorithms (types of calculation), etc. You can change the description (field) field, but not the variable name.
- **description [mandatory]**: literally that appears to the working user.
- **values [mandatory]**: default value or the possible answers that the user could give (separated by commas).
- **module [mandatory]:** indicates the module where you want to store the field. 0: results, 1: autocontrol.
- **type**:

  0: Real, number with decimals.

  1: Integer, number without decimals.

  2: Date, date in the dd/mm/yyyy format.

  3: Time, time in the hh:mm format.

  4: Text, free text.

  5: IP Address, IP network address.

6: Time Secs, time in hh:mm:ss format

7: Selection, allows to make a drop-down menu. Use the combos configuration window to configure options.

8: URL, url upon the system syntax. Please see how to configure URLs.

9: Alphabetic, only letters.

10: Date and time, dd/mm/yyyy hh:mm format.

11: Date and time (secs.), dd/mm/yyyy hh:mm:ss format.

12: Mult. Selection, multiple selection.

13: MAC, MAC address.

14: Percentage, number for percentages.

15: Autoincrement, unique numeric value for the field independent of the result or task to which it is associated.

16: Button, calls a function declared in a field management UDL.

17: TCP port, TCP port between 1 and 65535

18: Now date, current date in dd/mm/yyyy format.

19: Now time, current time in hh:mm format.

20: Now time (secs.), current time in hh:mm:ss format.

21: Now date and time, current date and time in dd/mm/yyyy hh:mm format.

22: Now date and time (secs.), current date and time in dd/mm/yyyy hh:mm:ss format.

23: E-mail, Email address.

- **domain:** if you selected the optins 7 or 12, then you shouuld specify the id of the combo.
- **apply**: indicates to what this field should be apply (Production, Inputs, Outputs, I&O [Inputs and Outputs], etc.).
- **mandatory**: when the field is mandatory. The options are the following:
  - When configuring fields for **results**:
    - **0:** only if empty.
    - **1:** the field is always mandatory.
    - **2:** the field is never mandatory.

- **3**: when results are being inserted manually.
- **4**: when finishing or pausing a order. This option will only apply for fields on inputs.
- **5**: when starting an order.

- When configuring fields for **autocontrol**:
  - **0:** never is mandatory.
  - **1:** when the status of the task changes.
  - **2:** if the task status is changed to Ok.
  - **3:** if the task status is changed to warning.
  - **4:** if the task status is changed to error.
  - **5:** if the task status is changed to error or warning.
  - **6:** when the task is started.

- **calculation**: calculation type to be applied to the field. This is useful to convert the values of the fields.
- **fclass**: this is to indicate to other computer systems the type of the field. Please see *integration with other systems*.
  - **0:** General purpose field class.
  - **1:** Unit of measure. If informed, prevails over the unit of measure of the process.
  - **2:** Cycle time. Is informed, prevails over the one of the process and the PSR relation.
  - **3:** Cycle quantity or cycle units. Is informed, prevails over the one of the process and the PSR relation.
  - **4:** inform about the material lot.
  - **5:** stores the corresponding B2MML tag.
  - **6:** stores the corresponding B2MML tag.
  - **7:** stores the corresponding B2MML tag.
  - **8:** stores the corresponding B2MML tag.
  - **9:** stores the corresponding B2MML tag.
  - **10:** stores the corresponding B2MML tag.

- **11:** stores the corresponding B2MML tag.
- **12:** stores the corresponding B2MML tag.
- **13:** stores the corresponding B2MML tag.
- **14:** stores the corresponding B2MML tag.
- **15:** stores the corresponding B2MML tag.
- **16:** stores the corresponding B2MML tag.
- **17:** stores the corresponding B2MML tag.
- **18:** stores the corresponding B2MML tag.
- **19:** every production counting is equivalent to how many pieces. For example: 1 unit (a box) is equivalent to 6 pieces.
- **20:** visual conversion from production units to visual units.

- **maxlength**: maximum length of the field.
- **calculated**: indicates if the field will be calculated by a UDL.
- **locked**: indicates if the field cannot be changed by the user.
- **inherits**: indicates if the field inherits its value from the field with the same variable name of the previous production record.
- **stock:** indicates if the field is used to diferentiate stock.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "Success" on success, and empty in case of error.
- **message**: "Success" or a descriptive information of the error in any other case.

Content-Type: application/json

Request: /fields/config?
company=edinn&session=0PY4RhPbK7YW5sbRXiwASrA2HEeCP0eGFzunnfiEJYWrp4w
hnU11FLESeYpB&idfield=S0000&mod=0

Reponse (success):

{

"status": "0",

"data":true,

"message": "Success"

```
}
```

PUT /fields/{id}/config

This method modify a field. The user must have the administrator role.

Request attribute:

- **id [mandatory]**: the Id of the field.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **variable[mandatory]**: internal name of the field. Allows to carry out statistical studies, links with algorithms (types of calculation), etc. You can change the description (field) field, but not the variable name.
- **description[mandatory]**: literally that appears to the working user.
- **value[mandatory]**: default value or the possible answers that the user could give (separated by commas).
- **module [mandatory]:** indicates the module where the field you want to modify is located. 0: results, 1: autocontrol.
- **type**: type of the field
- **idcomb:** if you selected the optins 7 or 12, then you shouuld specify the id of the combo.
- **io**: indicates to what this field should be apply (Production, Inputs, Outputs, I&O [Inputs and Outputs], etc.).
- **mandat**: when the field is mandatory. The options are the following:
- **idcalc**: calculation type to be applied to the field. This is useful to convert the values of the fields.
- **fclass**: this is to indicate to other computer systems the type of the field. Please see *integration with other systems*.
- **maxlength**: maximum length of the field.
- **calculated**: indicates if the field will be calculated by a UDL.
- **locked**: indicates if the field cannot be changed by the user.
- **inherits**: indicates if the field inherits its value from the field with the same variable name of the previous production record.

- **stock:** indicates if the field is used to diferentiate stock.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "Success" on success, and empty in case of error.
- **message**: "Success" or a descriptive information of the error in any other case.

```
Content-Type: application/json

Request:/fields/config?
company=edinn&session=0PY4RhPbK7YW5sbRXiwASrA2HEeCP0eGFzunnfiEJYWrp4w
hnU11FLESeYpB&results=S0000&variable=1a2s&description=testApi&values=modifiedFie
ld&mod=0&id=20210520092549AAB

Reponse (success):

{

  "status": "0",

  "data":true,

  "message": "Success"

}
```

DELETE /fields/{id}/config

This method deletes a configured fields. The user must have the administrator role.

Request attribute:

- **id [mandatory]**: the Id of the field.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **module [mandatory]:** indicates the module where the field you want to delete is located. 0: results, 1: autocontrol.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "Success" on success, and empty in case of error.
- **message**: "Success" or a descriptive information of the error in any other case.

```
Content-Type: application/json

Request:/fields/config?
company=edinn&session=0PY4RhPbK7YW5sbRXiwASrA2HEeCP0eGFzunnfiEJYWrp4w
hnU11FLESeYpB&id=20210520092549AAB&mod=0

Reponse (success):

{

  "status": "0",

  "data":true,

  "message": "Success"

}
```

Operation

BETA: This functionality is under development and there may be changes not reflected in this help.

This section handles the fields usage.

GET fields/{id}/values

This method returns the list of fields values.

Request attribute:

- **id [mandatory]**: the Id of the field.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **module [obligatorio]:** indicates the module where is the field you want to get. 0: results, 1: autocontrol.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the properties of the fields.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Content-Type: application/json

Request:/fields/values?
company=edinn&session=0PY4RhPbK7YW5sbRXiwASrA2HEeCP0eGFzunnfiEJYWrp4whnU11FLESeYpB&idfield1=20200411155128ACBCENTRAL&mod=0&idfield2=20200411155128ACBCENTRAL&parentid=20190507075910AAFDTRONCHO&values=testvalue&idproc=HC00

Reponse (success):


```
{
  "status": "0",
  "data": [
    {
        "id": "20200423095359AAQDTRONCHO",
        "machdate": "HC0020200423095359",
        "id2": "20200421121153AAPDTRONCHO",
        "id_field": "20190507075910AAFDTRONCHO",
        "val": "",
        "id_author": "DAVID.TRONCHONI",
        "created": "20200423095359",
        "id_mod": "",
        "modified": ""
    },
     {
        "id": "20200423095359AATDTRONCHO",
        "machdate": "HC0020200423095359",
        "id2": "20200421121153AAPDTRONCHO",
        "id_field": "20190507093159AANDTRONCHO",
        "val": "20200423140000",
        "id_author": "DAVID.TRONCHONI",
        "created": "20200423095359",
        "id_mod": "",
        "modified": ""
    }
```

```
 ],
 "message": "Success"
}
```

GET fields/values/gen

This method returns the list of fields values.

Request attribute:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **idproc [mandatory]:** id of the process.
- **module [obligatorio]:** indicates the module where is the field you want to get. 0: results, 1: autocontrol.
- **field1 [obligatorio]**: id of the result or autocontrol
- **field2 [obligatorio]**: id of the proces, status, result relation
- **field3 [obligatorio]**: id of the order, production registry or autocontrol
- **value [mandatory]: v**alue to insert
- **parent [mandatory]: i**d of the field you want to assign the value
- **result:** Id of the result (indicate only if there is a result change)

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the properties of the fields.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

```
Content-Type: application/json

Request:/fields/values?
company=edinn&session=0PY4RhPbK7YW5sbRXiwASrA2HEeCP0eGFzunnfiEJYWrp4whnU
11FLESeYpB&idfield1=20200411155128ACBCENTRAL&mod=0&idfield2=20200411155128AC
BCENTRAL&parentid=20190507075910AAFDTRONCHO&values=testvalue&idproc=HC00

Reponse (success):

{
```

```
"status": "0",
"data": [
    {
        "id": "20200423095359AAQDTRONCHO",
        "machdate": "HC0020200423095359",
        "id2": "20200421121153AAPDTRONCHO",
        "id_field": "20190507075910AAFDTRONCHO",
        "val": "",
        "id_author": "DAVID.TRONCHONI",
        "created": "20200423095359",
        "id_mod": "",
        "modified": ""
    },
     {
        "id": "20200423095359AATDTRONCHO",
        "machdate": "HC0020200423095359",
        "id2": "20200421121153AAPDTRONCHO",
        "id_field": "20190507093159AANDTRONCHO",
        "val": "20200423140000",
        "id_author": "DAVID.TRONCHONI",
        "created": "20200423095359",
        "id_mod": "",
        "modified": ""
    }
],
"message": "Success"
}
```


POST /fields/values

This method creates a new field values. The user must have the administrator role.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **idproc [mandatory]:** id of the process.
- **module [obligatorio]:** indicates the module where is the field you want to get. 0: results, 1: autocontrol.
- **field [mandatory]**: id of the result
- **value [mandatory]:** value to insert
- **parent [mandatory]:** id of the field you want to assign the value

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "Success" on success, and empty in case of error.
- **message**: "Success" or a descriptive information of the error in any other case.

Content-Type: application/json

Request: /fields/values?
company=edinn&session=0PY4RhPbK7YW5sbRXiwASrA2HEeCP0eGFzunnfiEJYWrp4w
hnU11FLESeYpB&idfield1=20200411155128ACBCENTRAL&mod=0&idfield2=20200411155
128ACBCENTRAL&parentid=20190507075910AAFDTRONCHO&values=testvalue&idproc
=HC00

Reponse (success):

```
{
  "status": "0",
  "data":true,
    "message": "Success"
}
```

DELETE /fields/{id}/values

This method deletes a configured fields value. The user must have the administrator role.

Request attribute:

- **id [mandatory]**: the Id of the field.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **module [mandatory]:** indicates the module where the field you want to delete is located. 0: results, 1: autocontrol.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "Success" on success, and empty in case of error.
- **message**: "Success" or a descriptive information of the error in any other case.

```
Content-Type: application/json

Request:/fields/values?
company=edinn&session=0PY4RhPbK7YW5sbRXiwASrA2HEeCP0eGFzunnfiEJYWrp4w
hnU11FLESeYpB&id=20200411155129ADQCENTRAL&mod=0

Reponse (success):

{
  "status": "0",
  "data":true,
  "message": "Success"
}
```

Domains

Configuration

> **BETA:** This functionality is under development and there may be changes not reflected in this help.

This section handles the domains configuration.

GET /domains

This method returns the list of domains.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of domains.
- **message**: "Success" or a descriptive information of the error in any other case.

```
Request:
{
 "session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
 "company": "demo01",
}

Response:
{
"status": "0",
"data": [
        {
        "Id": "clients",
        "Desc": "Clientes"
        },
          {
        "Id": "No",
        "Desc": ""
        },
          {
        "Id": "P1",
```

```
        "Desc": "P1"
        }
    ],
"message": "Success"
}
```

POST /domains/{id}

This method creates a domain. The user must have the administrator role.

Request attribute:

- **id [mandatory]**: the Id of the domain.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **name [mandatory]**: the name of the measure unit.
- **description [mandatory]:**the description of the domain.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "success" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.

```
Request:
{
 "session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
 "company": "demo01",
 "idcomb": "test",
 "desc": "testcomb"
}
Response:
```

```
{
"status": 0,
"data": true,
"message": "Success"
}
```

PUT/domains/{id}

This method modify a domain. The user must have the administrator role.

Request attribute:

- **id [mandatory]**: the Id of the domain.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **description [mandatory]:**the description of the combo.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "success" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.

```
Request:
{
 "session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
 "company": "demo01",
 "idcomb": "test",
 "desc": "testcomb"
}
Response:
{
```

```
"status": 0,
"data": true,
"message": "Success"
}
```

DELETE /domains/{id}

This method deletes a domain. The user must have the administrator role.

Request attribute:

- **id [mandatory]**: the Id of the domain.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "success" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.

```
Request:
{
 "session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
 "company": "demo01",
 "idcomb": "test"
}
Response:
{
"status": 0,
"data": true,
"message": "Success"
```

```
}
```

Values

> **BETA:** This functionality is under development and there may be changes not reflected in this help.

This section handles the domains values configuration.

GET /domains/{id}/values

This method returns the list of domains.

Request attribute:

- **id [mandatory]**: the Id of the domain.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: returns the list of values of the combo.
- **message**: "Success" or a descriptive information of the error in any other case.

```
Request:
{
 "session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
 "company": "demo01",
 "idcomb": "No"
}

Response:
```

```
{
"status": "0",
"data": [
        {
        "Val": "No"
        },
          {
        "Val": "Si"
        }
    ],
"message": "Success"
}
```


POST/domains/{id}/values

This method create a value of a domain. The user must have the administrator role.

Request attribute:

- **id [mandatory]**: the Id of the domain.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.
- **description [mandatory]:**the description of the combo.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "success" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.


```
Request:
{
```

```
"session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
"company": "demo01",
"idcomb": "test",
"value": "value1"
}
Response:
{
"status": 0,
"data": true,
"message": "Success"
}": "Success"
}
```


DELETE /domains/{id}/values

This method deletes the values of a domain. The user must have the administrator role.

Request attribute:

- **id [mandatory]**: the Id of the domain.

Request parameters:

- **company [mandatory]**: the company name or id.
- **session [mandatory]**: the Id of the active session.

Response:

- **status**: returns "0" on success, and an error code in any other case.
- **data**: "success" on success, and empty in case of error.
- **message**: "success" or a descriptive information of the error in any other case.


```
Request:
{
"session": "cJvqHJcEssmpGukxrZF73migL5M4LVUZniyesQHBatOvTkrbpkYgrHKvfVpP",
```

```
 "company": "demo01",
 "idcomb": "test"
}
Response:
{
"status": 0,
"data": true,
"message": "Success"
}
```


Analysis

This section describes the calls for obtaining analysis reports.

GET /analysis/productivity

This method returns the same information as the Analysis/Productivity report.

Request parameters:

- **company [required]:** The name or ID of the company.
- **session [required]:** The ID of the active session.
- **datefrom [required]:** The start date of the selection.
- **dateto [required]**: the end date of the selection.
- **tree**: the tree structure with the selected areas and processes. Default value: empty, equivalent to all areas and processes. Format: area1{process1,process2,etc.},area2{...
- **inhours**: indicates whether the times will be returned in hours (1) or in minutes (0). Default value: 0.
- **inkg**: indicates whether the quantities will be returned in kilograms (1) or in units (0). Default value: 0.
- **detail**: indicates if the results will be returned grouped (0) or with detail (1). Default value: 0.

- **explode_operation**: indicates whether the results will be returned grouped by order (0) or differentiated by operation (1). Default value: 0.
- **totalize**: indicates whether a row will be returned with the totals (1) or not (0). Default value: 1.
- **fields**: if indicated, it displays and differentiates the results based on the values of the indicated fields. Up to two fields separated by "|" can be included.
- **contains**: if specified, filters the data returned based on whether the specified text is in the chosen field. You can put a value for each field, separated by "|".

Response:

- **status**: returns "0" in case of success and an error code in any other case.
- **data**: returns the same data as the productivity report.
- **message**: "Success" or a descriptive information of the error in any other case.

Example:

Request:

{"type":"get",
"url":"analysis/productivity",
"data":"company=1763635394&session=MYNzf634ce2lST5eliLPTOYfOVhzhpufAEsvDlb9xm8zISQ0WLH6Uh3CVSxs&datefrom=20210501054500&dateto=20210601054500&inhours=1&inkg=1&detail=1&explode_operation=1&fields=Planta|ProductionLine"}

Reponse (success):

{"status":"0",
"data":[{"area":"SOL-SOLIDOS\/POLVOS","process":"RO02-Rovema 02","planta":"1010","productionline":"L18","production_order":"100263275-0010","result":"86002149\/0012","Description":"SODIUM SACCHARIN. 2.1 KG","good_quantity":"1","units":"PK","scrap":"0","rework":"0","good_quantity_kg":"2.1","start":"07\/05\/2021 09:22:20","end":"01\/06\/2021 05:45:00","total_time":"596.34","availability_losses":"330.13","production_time":"266.21","not_required_time":"0.04","attendance_time":"894.57","productivity":"0"},
{"area":"","process":"","planta":"","productionline":"","production_order":"","result":"","Description":"","good_quantity":"1","units":"PK","scrap":"0","rework":"0","good_quantity_kg":"2.1","start":"07\/05\/2021 09:22:20","end":"01\/06\/2021 05:45:00","total_time":"596.34","availability_losses":"330.13","production_time":"266.21","not_required_time":"0.04","attendance_time":"894.57","productivity":"0"}],
"message":"Success"}

Reponse (error):

{"status":"1070",
"data":"<a href='http:\/\/mayen19:8080\/edinnM2\/help\/en\/API_Analysis_Productivity.html'>for additional information follow this link<\/a>",
"message":"The dateFrom should be earlier than dateTo"}

Subscriptions

Common

This section explains the common aspects of the messages edinn sends to the systems with which it is integrated. To receive these messages proceed with the appropriate configuration in the interface tab of the server console.

Common aspects about ISA-95 compliant messages (for more information please read the B2MML document).

Attending to the ISA-95 classification, edinn has four types of messages (SyncProductionSchedule, SyncProductionPerformance, CancelProductionPerformance and GetMaterialInformation) and expects two types of responses (ConfirmBOD and ShowMaterialInformation).

Production Schedule Synchronizationa and Production Performance Synchronization are Synchronization messages.

Each message has two sections:

- **ApplicationArea**: contains the metadata.
    - Sender: identifies the message sender.
        - LogicalID: unique identifier of the message that is usefull to prevent to duplicate information.
        - ComponentID: identifier of the sender, usually the software that generates the message.
        - ConfitmationCode: indicates when the sender expects a response. The possible values are: Always, OnError and Never.
    - CreationDateTime: date and time when the message was created.
- **DataArea**: contains the action to perform ans the data itself.
    - Sync/ActionCriteria/ActionsExpression: action to perform. The possible values are: Add, Change and Delete.
    - ProductionSchedule or ProductionPerformance: Contains the data to be porocessed. Detailed information in the corresponding section of help document.

```
{
     "SyncProductionSchedule": {
          "ApplicationArea": {
               "Sender": {
                    "LogicalID": "20200526172352AABCENTRAL",
                    "ComponentID": "EDINNM2",
                    "ConfirmationCode": "Always"
               },
```

```
                "CreationDateTime": "2020-05-22T12:15:45"
          },
          "DataArea": {
                "Sync": {
                     "ActionCriteria": {
                          "ActionExpression": {
                               "@actionCode": "Change"
                          }
                     }
                },
                "ProductionSchedule": {
                   ...
                }
                OR
                "ProductionPerformance": {
                   ...
                }
          }
     }
}
```

Production Schedule Synchronization messages:

- **ID**: Identifies the group of orders.
- **ProdutionSchedule**: Contains the information about each order.
    - **ID**: Identifies the order.
    - **SegmentRequirement**: Contains the information about each operation.
        - **ID**: Identifies the operation.

```
...
"ProductionSchedule": {
     "ID": "20200401",
     "ProductionRequest": {
          "ID": "2020040128-10",
          ...
          "SegmentRequirement": {
               "ID": "10",
               ...
          }
     }
}
...
```

Production Performance Synchronization messages:

- **ProdutionResponse**: Contains the information about each order.
    - **ProductionRequestID**: Identifies the order.
    - **SegmentResponse**: Contains the information about each operation.

- **ID**: Identifies the operation.

```
...
"ProductionPerformance": {
     "ProductionResponse": {
          "ProductionRequestID": "20200202",
          "SegmentResponse": {
               "ID": "10",
...
```

Production Performance Cancellation messages begin with the key CancelProductionPerformance and the structure is similar to the synchronization messages.

BOD Confirmation is the expected response to Synchronization and Cancelation messages.

Each message has two sections: ApplicationArea and DataArea. DataArea contains the response itself.

- **Confirm**: Contains the key value in ResponseCriteria/ResponseExpression/actionCode. The possible values are: Accpeted or Rejected.
- **BOD**: Contains detailed information about the response.
    - **Description**: A descriptive message about the cause of the error (in case of rejected message).
    - **Note**: A code to identify the notification in the subscriber, needed for audit purposes.

Expected response for accepted messages.

```
{
     "ConfirmBOD": {
          "ApplicationArea": {
               "Sender": {
                    "LogicalID": "20200526172352AABCENTRAL",
                    "ComponentID": "ERP",
                    "ConfirmationCode": "Always"
               },
               "CreationDateTime": "2020-05-26T13:25:23"
          },
          "DataArea": {
               "Confirm": {
                    "ResponseCriteria": {
                         "ResponseExpression": {
                              "@actionCode": "Accepted"
                         }
                    }
               },
               "BOD": {
                    "Description": "Accepted",
                    "Note": "1590492323.1673"
               }
          }
```

```
      }
}
```

Expected response for rejected messages.

```
{
      "ConfirmBOD": {
            "ApplicationArea": {
                  "Sender": {
                        "LogicalID": "20200526172352AABCENTRAL",
                        "ComponentID": "ERP",
                        "ConfirmationCode": "Always"
                  },
                  "CreationDateTime": "2020-05-14T15:56:15"
            },
            "DataArea": {
                  "Confirm": {
                        "ResponseCriteria": {
                              "ResponseExpression": {
                                    "@actionCode": "Rejected"
                              }
                        }
                  },
                  "BOD": {
                        "Description": "Error descripion",
                        "Note": "Error aditional information"
                  }
            }
      }
}
```

Send Time Usages

This section shows the messages that edinn send with the time usage information (ISA-95 compliant, for more information please read the B2MML document).

Send time usage with process (equipment) information:

According to the ISA-95 standard, this is a production performance syinchronization information and expects a BOD confirmation as response message.

Edinn sends time information grouped by time class. Each class generates a SegmentResponse which its ID is the class identifier.

- **ID**: Identifier of the time class.
- **ActualStartTime**: The time when the period started.
- **ActualEndTime**: The time when the period ended.
- **EquipmentActual**: Contains the informatio about the process tha perform de operation

- **Quantity**: Contains the duration of the time class.

```
{
     "SyncProductionPerformance": {
          "ApplicationArea": {
               "Sender": {
                    "LogicalID": "20200306113003AABCENTRAL",
                    "ComponentID": "EDINNM2",
                    "ConfirmationCode": "Always"
               },
               "CreationDateTime": "2020-03-06T11:30:03"
          },
          "DataArea": {
               "Sync": {
                    "ActionCriteria": {
                         "ActionExpression": {
                              "@actionCode": "Add"
                         }
                    }
               },
               "ProductionPerformance": {
                    "ProductionResponse": {
                         "ProductionRequestID": "20200202",
                         "SegmentResponse": {
                              "ID": "10",
                              "ProcessSegmentId":
"TSTU20200306113003JADCENTRAL",
                              "ActualStartTime": "2020-03-
06T11:02:35",
                              "ActualEndTime": "2020-03-
06T11:29:56",
                              "EquipmentActual": {
                                   "EquipmentID": "7101",
                                   "Quantity": {
                                        "QuantityString": "1.000",
                                        "DataType": "integer",
                                        "UnitOfMeasure": "UN"
                                   }
                              },
                              "SegmentResponse": [
                                   {
                                        "ID": "Setup",
                                        "ProcessSegmentId": [],
                                        "ActualStartTime": "2020-
03-06T11:02:35",
                                        "ActualEndTime": "2020-03-
06T11:29:56",
                                        "EquipmentActual": {
                                             "EquipmentID":
"7101",
                                             "Quantity": {

"QuantityString": "0.014",
                                                  "DataType":
"double",
```

```
"UnitOfMeasure": "HUR"
                                                }
                                          }
                                    },
                                    {
                                          "ID": "Productive",
                                          "ProcessSegmentId": [],
                                          "ActualStartTime": "2020-
03-06T11:02:35",
                                          "ActualEndTime": "2020-03-
06T11:29:56",
                                          "EquipmentActual": {
                                                "EquipmentID":
"7101",
                                                "Quantity": {
"QuantityString": "0.442",
                                                      "DataType":
"double",
"UnitOfMeasure": "HUR"
                                                }
                                          }
                                    }
                              ]
                        }
                  }
            }
      }
}
```

Send time usage with labour:

Adds information about the labour time for each time period.

```
{
      "SyncProductionPerformance": {
            "ApplicationArea": {
                  "Sender": {
                        "LogicalID": "20200526172352AABCENTRAL",
                        "ComponentID": "EDINNM2",
                        "ConfirmationCode": "Always"
                  },
                  "CreationDateTime": "2020-05-26T13:25:23"
            },
            "DataArea": {
                  "Sync": {
                        "ActionCriteria": {
                              "ActionExpression": {
                                    "@actionCode": "Add"
                              }
                        }
                  },
```

```
                "ProductionPerformance": {
                     "ProductionResponse": {
                          "ProductionRequestID": "2020040102",
                          "SegmentResponse": {
                               "ID": "10",
                               "ProcessSegmentId":
"TSTU20200526124402AAFINIT",
                               "ActualStartTime": "2020-05-
26T10:30:00",
                               "ActualEndTime": "2020-05-
26T10:33:50",
                               "EquipmentActual": {
                                    "EquipmentID": "7101",
                                    "Quantity": {
                                         "QuantityString":
"1,000000",
                                         "DataType": "integer",
                                         "UnitOfMeasure": "UN"
                                    }
                               },
                               "SegmentResponse": {
                                    "ID": "Failure",
                                    "ProcessSegmentId": [],
                                    "ActualStartTime": "2020-05-
26T10:30:00",
                                    "ActualEndTime": "2020-05-
26T10:33:50",
                                    "EquipmentActual": {
                                         "EquipmentID": "7101",
                                         "Quantity": {
                                              "QuantityString":
"0,063889",
                                              "DataType":
"double",
                                              "UnitOfMeasure":
"HUR"
                                         }
                                    },
                                    "PersonnelActual": {
                                         "PersonID": "0703",
                                         "Quantity": {
                                              "QuantityString":
"0,063889",
                                              "DataType":
"double",
                                              "UnitOfMeasure":
"HUR"
                                         }
                                    }
                               }
                          }
                     }
                }
           }
      }
```

```
}
```

Cancel time usage with labour:

This message send a cancellation all or a part of a time notification.

```
{
     "CancelProductionPerformance": {
          "ApplicationArea": {
               "Sender": {
                    "LogicalID": "20200526172352AABCENTRAL",
                    "ComponentID": "EDINNM2",
                    "ConfirmationCode": "Always"
               },
               "CreationDateTime": "2020-05-26T13:25:22"
          },
          "DataArea": {
               "Cancel": [],
               "ProductionPerformance": {
                    "ProductionResponse": {
                         "ProductionRequestID": "2020040102",
                         "SegmentResponse": {
                              "ID": "10",
                              "ProcessSegmentId":
"20200526124402AAFINIT",
                              "ActualStartTime": "2020-05-
26T10:30:00",
                              "ActualEndTime": "2020-05-
26T10:33:50",
                              "SegmentResponse": {
                                   "ID": "Productive",
                                   "ProcessSegmentId": [],
                                   "ActualStartTime": "2020-05-
26T10:30:00",
                                   "ActualEndTime": "2020-05-
26T10:33:50",
                                   "EquipmentActual": {
                                        "EquipmentID": "7101",
                                        "Quantity": {
                                             "QuantityString":
"0,063889",
                                             "DataType":
"double",
                                             "UnitOfMeasure":
"HUR"
                                        }
                                   },
                                   "PersonnelActual": {
                                        "PersonID": "0703",
                                        "Quantity": {
                                             "QuantityString":
"0,063889",
                                             "DataType":
"double",
                                             "UnitOfMeasure":
```

```
"HUR"
                                                    }
                                                }
                                            }
                                        }
                                    }
                                }
                            }
                        }
                    }
}
```

Send Quantity Notifications

This section shows the messages that edinn send with the quantity information, both for prduction, scrap, reword and consumption (ISA-95 compliant, for more information please read the B2MML document).

Sends production and labour time information on shift change or order status change or in a manual production:

ProductionPerformance: Section where the data to be processed is:

- **ProcutionResponse**: The response for each order.
    - **ProductionRequestID**: Identifier of the order.
    - **SegmenetResponse**: The information about each operation inside the order.
        - **ID**: Identifier of the operation.
        - **ProcessSegmentId**: Identifier of the message in edinn. Unsed for audit purposes.
        - **ActualStartTime**: time where the period notified started.
        - **ActualEndTime**: time where the period notified ended.
        - **EquipmentAcutal**: Process where the operatio was performed.
            - **Quantity**: amount of processes employed.
        - **MaterialActual**: Result to be notified to the subcriber.
            - **MaterialDefinitionID**: Identifier of the result.
            - **MaterialLotID**: Lot number.
            - **MaterialSubLotID**: For "Consumed" and "Output" results, the identifier of the production related with.
            - **Location**: Information about where, in the intallation the material

was produced (plant, area, production line) and/or has to be stored (storage zone, storage unit).

- **MaterialUse**: One of the three possible uses of the result: "Produced" is the main result of the order, "Consumed" is the raw material used in the process, "Output" is a secoundary result obtained in the process.
- **Quantity**: Amount of result to be notified to the subscriber.

- **PersonnelActual**: Labour time employed in the operation performance.
  - **PersonID**: Recourse identifier.
  - **Quantity**: Amount of time employed by the recourse.

```
{
     "SyncProductionPerformance": {
          "ApplicationArea": {
               "Sender": {
                    "LogicalID": "20200526172352AABCENTRAL",
                    "ComponentID": "EDINNM2",
                    "ConfirmationCode": "Always"
               },
               "CreationDateTime": "2020-05-15T11:27:20"
          },
          "DataArea": {
               "Sync": {
                    "ActionCriteria": {
                         "ActionExpression": {
                              "@actionCode": "Add"
                         }
                    }
               },
               "ProductionPerformance": {
                    "ProductionResponse": {
                         "ProductionRequestID": "2020040101",
                         "SegmentResponse": {
                              "ID": "10",
                              "ProcessSegmentId":
"TSPR20200515112720PKTCENTRAL",
                              "ActualStartTime": "2020-05-
15T11:18:31",
                              "ActualEndTime": "2020-05-
15T11:27:10",
                              "EquipmentActual": {
                                   "EquipmentID": "7101",
                                   "Quantity": {
                                        "QuantityString": "1.000",
                                        "DataType": "integer",
                                        "UnitOfMeasure": "UN"
                                   }
                              },
                              "MaterialActual": {
```

```
                                        "MaterialDefinitionID":
"BB0932428102-4351",
                                        "MaterialLotID": [],
                                        "MaterialSubLotID": [],
                                        "Location": {
                                             "EquipmentID":
"PBPACKAGING",
                                             "EquipmentElementLevel":
"Enterprise"
                                        },
                                        "MaterialUse": "Produced",
                                        "Quantity": {
                                             "QuantityString":
"21.000",
                                             "DataType": "double",
                                             "UnitOfMeasure": "MI"
                                        }
                                   },
                                   "PersonnelActual": {
                                        "PersonID": "0703",
                                        "Quantity": {
                                             "QuantityString":
"14.000",
                                             "DataType": "double",
                                             "UnitOfMeasure": "HUR"
                                        }
                                   }
                              }
                         }
                    }
               }
          }
}
```

Sending scrap information:

```
{
     "SyncProductionPerformance": {
          "ApplicationArea": {
               "Sender": {
                    "LogicalID": "20200526172352AABCENTRAL",
                    "ComponentID": "EDINNM2",
                    "ConfirmationCode": "Always"
               },
               "CreationDateTime": "2020-05-15T11:27:20"
          },
          "DataArea": {
               "Sync": {
                    "ActionCriteria": {
                         "ActionExpression": {
                              "@actionCode": "Add"
                         }
                    }
               },
               "ProductionPerformance": {
```

```
                    "ProductionResponse": {
                         "ProductionRequestID": "2020040101",
                         "SegmentResponse": {
                              "ID": "10",
                              "ProcessSegmentId":
"TSPR20200515112720PKUCENTRAL",
                              "ActualStartTime": "2020-05-
15T11:18:31",
                              "ActualEndTime": "2020-05-
15T11:27:11",
                              "EquipmentActual": {
                                   "EquipmentID": "7101",
                                   "Quantity": {
                                        "QuantityString": "1.000",
                                        "DataType": "integer",
                                        "UnitOfMeasure": "UN"
                                   }
                              },
                              "MaterialActual": {
                                   "MaterialDefinitionID":
"BB0932428102-4351.S",
                                   "MaterialLotID": [],
                                   "MaterialSubLotID": [],
                                   "Location": {
                                        "EquipmentID":
"PBPACKAGING",
                                        "EquipmentElementLevel":
"Enterprise"
                                   },
                                   "MaterialUse": "Produced",
                                   "Quantity": {
                                        "QuantityString": "1.000",
                                        "DataType": "double",
                                        "UnitOfMeasure": "MI"
                                   }
                              }
                         }
                    }
               }
          }
}
```

Send consumption of raw materials:

```
{
     "SyncProductionPerformance": {
          "ApplicationArea": {
               "Sender": {
                    "LogicalID": "20191210194048QQTCENTRAL",
                    "ComponentID": "EDINNM2",
                    "ConfirmationCode": "Always"
               },
               "CreationDateTime": "2019-12-10T19:40:48"
          },
```

```
          "DataArea": {
               "Sync": {
                    "ActionCriteria": {
                         "ActionExpression": {
                              "@actionCode": "Add"
                         }
                    }
               },
               "ProductionPerformance": {
                    "ProductionResponse": {
                         "ProductionRequestID": "19018632",
                         "SegmentResponse": {
                              "ID": "10",
                              "ProcessSegmentId":
"TSIO20191210194048IHPCENTRAL",
                              "ActualStartTime": "2019-12-
10T16:37:11",
                              "ActualEndTime": "2019-12-
10T19:39:42",
                              "MaterialActual": {
                                   "MaterialDefinitionID":
"BO01032050",
                                   "MaterialLotID": "2019022272",
                                   "MaterialSubLotID":
"20191210194048IHPCENTRAL",
                                   "Location": {
                                        "EquipmentID": "IMCOVEL",
                                        "EquipmentElementLevel":
"Enterprise"
                                   },
                                   "MaterialUse": "Consumed",
                                   "Quantity": {
                                        "QuantityString":
"6693,813",
                                        "DataType": "double",
                                        "UnitOfMeasure": "ML"
                                   },
                                   "MaterialProperty": {
                                        "ID": "TERMINADA",
                                        "Value": {
                                             "ValueString": "No",
                                             "DataType": [],
                                             "UnitOfMeasure": []
                                        }
                                   }
                              }
                         }
                    }
               }
          }
     }
}
```

Send output of materials:

Output materials are coproducts obtained in the production process different than the main result of the order.

```
{
     "SyncProductionPerformance": {
          "ApplicationArea": {
               "Sender": {
                    "LogicalID": "20191210194048QQTCENTRAL",
                    "ComponentID": "EDINNM2",
                    "ConfirmationCode": "Always"
               },
               "CreationDateTime": "2019-12-10T19:40:48"
          },
          "DataArea": {
               "Sync": {
                    "ActionCriteria": {
                         "ActionExpression": {
                              "@actionCode": "Add"
                         }
                    }
               },
               "ProductionPerformance": {
                    "ProductionResponse": {
                         "ProductionRequestID": "19018632",
                         "SegmentResponse": {
                              "ID": "10",
                              "ProcessSegmentId":
"TSIO20191210194048IHPCENTRAL",
                              "ActualStartTime": "2019-12-
10T16:37:11",
                              "ActualEndTime": "2019-12-
10T19:39:42",
                              "MaterialActual": {
                                   "MaterialDefinitionID":
"BO01032050",
                                   "MaterialLotID": "2019022272",
                                   "MaterialSubLotID":
"20191210194048IHPCENTRAL",
                                   "Location": {
                                        "EquipmentID": "IMCOVEL",
                                        "EquipmentElementLevel":
"Enterprise"
                                   },
                                   "MaterialUse": "Output",
                                   "Quantity": {
                                        "QuantityString":
"6693,813",
                                        "DataType": "double",
                                        "UnitOfMeasure": "ML"
                                   },
                                   "MaterialProperty": {
                                        "ID": "TERMINADA",
                                        "Value": {
                                             "ValueString": "No",
                                             "DataType": [],
```

```
                                                    "UnitOfMeasure": []
                                               }
                                          }
                                     }
                                }
                           }
                      }
                 }
            }
}
```

Responses to a quantity notification:

The respose to the quantity notifications have a extra section after the Note tag called UserArea.

Notified: It is used for quantity notifications of the result sent as "Produced" in MaterialUse. Note that the response to a consumed material must be "Consumed" instead of "Notified", and in case of output material must be "Output".

- **ID**: Identifier fo the notification or the stock movement in the subscriber.
- **SubLotId**: Used to identify the record in edinn for "Consumed" and "Output" notifications. For "Produced" material is empty.

```
{
      "ConfirmBOD": {
            "ApplicationArea": {
                  "Sender": {
                        "LogicalID": "20200526172352AABCENTRAL",
                        "ComponentID": "ERP",
                        "ConfirmationCode": "Always"
                  },
                  "CreationDateTime": "2016-12-01T17:44:41"
            },
            "DataArea": {
                  "Confirm": {
                        "ResponseCriteria": {
                              "ResponseExpression": {
                                    "@actionCode": "Accepted"
                              }
                        }
                  },
                  "BOD": {
                        "Description": "Accepted
20161130114849AADCENTRAL",
                        "Note": "The segment 20161130114849AADCENTRAL
has finished successful.",
                        "UserArea": {
                              "Notified": {
                                    "ID": "439289",
                                    "SubLotID": []
                              },
                              "Consumed": {
```

```
                                    "ID": [],
                                    "SubLotID": []
                               },
                               "Output": {
                                    "ID": [],
                                    "SubLotID": []
                               }
                          }
                     }
                }
           }
}
```

Cancel Quantity Notifications

This section shows the messages that edinn send with the quantity cancellation, both for prduction, scrap, rework and consumption (ISA-95 compliant, for more information please read the B2MML document).

Cancel production:

To cancel a production performance, there is a section called Any after the MaterialActual.

UserArea: cutomizable section to add specific information.

- **Notified:**
    - **ID**: Identifies the record to remove in the subscriber.
    - **SubLotID**: Identifies the record in edinn to be cancelled.

```
{
     "CancelProductionPerformance": {
          "ApplicationArea": {
               "Sender": {
                    "LogicalID": "20200526172352AABCENTRAL",
                    "ComponentID": "EDINNM2",
                    "ConfirmationCode": "Always"
               },
               "CreationDateTime": "2020-05-26T17:23:52"
          },
          "DataArea": {
               "Cancel": [],
               "ProductionPerformance": {
                    "ProductionResponse": {
                         "ProductionRequestID": "2020040102",
                         "SegmentResponse": {
                              "ID": "10",
                              "ProcessSegmentId":
"TSPR20200526104816ABQCENTRAL",
                              "ActualStartTime": [],
```

```
                                    "ActualEndTime": "2020-05-
26T10:48:15",
                                    "MaterialActual": {
                                          "MaterialDefinitionID":
"BB0932428102-4351",
                                          "MaterialLotID": [],
                                          "MaterialSubLotID": [],
                                          "Location": {
                                                "EquipmentID":
"PBPACKAGING",
                                                "EquipmentElementLevel":
"Enterprise"
                                          },
                                          "MaterialUse": "Produced",
                                          "Quantity": {
                                                "QuantityString": "0,100",
                                                "DataType": "double",
                                                "UnitOfMeasure": "MI"
                                          }
                                    },
                                    "Any": {
                                          "UserArea": {
                                                "Notified": {
                                                      "ID":
"1590491974.5437",
                                                      "SubLotID":
"20200526104816ABQCENTRAL"
                                                }
                                          }
                                    }
                              }
                        }
                  }
            }
      }
}
```

Notify Production Schedule

This section shows the messages that edinn send on order status change (ISA-95 compliant, for more information please read the B2MML document).

Order status change:

To inform about the change of order status, in the SegmentRequirement section edinn adds a section called SegmentState.

- **SegementState** contains the status of the operation. The possible values are: Forecast (for scheduled operations), Started, Paused, Released (for prefinshed operations), Completed (for finished operations) and Canceled.

```
{
     "SyncProductionSchedule": {
          "ApplicationArea": {
               "Sender": {
                    "LogicalID": "20200526172352AABCENTRAL",
                    "ComponentID": "EDINNM2",
                    "ConfirmationCode": "Always"
               },
               "CreationDateTime": "2020-05-22T12:15:45"
          },
          "DataArea": {
               "Sync": {
                    "ActionCriteria": {
                         "ActionExpression": {
                              "@actionCode": "Change"
                         }
                    }
               },
               "ProductionSchedule": {
                    "ID": "20200401",
                    "ProductionRequest": {
                         "ID": "2020040128-10",
                         "StartTime": "2019-12-12T10:41:00",
                         "EndTime": "2019-12-20T09:11:00",
                         "RequestState": {
                              "@OtherValue": "Started",
                              "$": "Other"
                         },
                         "SegmentRequirement": {
                              "ID": "10",
                              "SegmentState": {
                                   "@OtherValue": "Started",
                                   "$": "Other"
                              }
                         }
                    }
               }
          }
     }
}
```

Material Information & Stock

This section shows the messages that edinn send requesting stock information (ISA-95 compliant, for more information please read the B2MML document).

GetMaterialInformation is the message to ask the subscriber the amount of material available to be consume in a order.

- **Location**: Defines where to search the material information.

- **MaterialDefinition**: Identifies the result its information are asking for.
    - **Id**: Identifier of the result.
    - **MaterialDefinitionProperty**: If informed, ask for the amount of material in a especific unit of measure.

```
{
     "GetMaterialInformation": {
          "ApplicationArea": {
               "Sender": {
                    "LogicalID": "20200526172352AABCENTRAL",
                    "ComponentID": "EDINNM2",
                    "ConfirmationCode": "Always"
               },
               "CreationDateTime": "2013-05-27T12:46:44"
          },
          "DataArea": {
               "Get": [],
               "MaterialInformation": {
                    "Location": {
                         "EquipmentID": "ESP1",
                         "EquipmentElementLevel": "Site"
                    },
                    "MaterialDefinition": {
                         "ID": "34003198",
                         "MaterialDefinitionProperty": {
                              "ID": "UnitOfMeasure",
                              "Description": "UnitOfMeasure",
                              "Value": {
                                   "ValueString": "KG",
                                   "DataType": [],
                                   "UnitOfMeasure": []
                              }
                         }
                    }
               }
          }
     }
}
```

ShowMaterialInformation returns the stock of the requested result by lot and location.

MaterialDefinition: Returns the list of material lots.

MaterialLot: Returns the location and stock of each lot.

- **ID**: Lot identifier
- **MaterialDefinitionID**: Identifier of the result that the lot belongs to.
- **Location**: storage location of the lot.
- **Quantity**: amount of material present in each location.

```
{
     "ShowMaterialInformation": {
          "ApplicationArea": {
               "Sender": {
                    "LogicalID": "20200526172352AABCENTRAL",
                    "ComponentID": "ERP",
                    "ConfirmationCode": "Always"
               },
               "CreationDateTime": "2013-05-27T12:46:45"
          },
          "DataArea": {
               "Show": [],
               "MaterialInformation": {
                    "Location": {
                         "EquipmentID": "ESP1",
                         "EquipmentElementLevel": "Site"
                    },
                    "MaterialDefinition": {
                         "ID": "34003198",
                         "MaterialLotID": [
                              "201133R018",
                              "201138R033"
                         ]
                    },
                    "MaterialLot": [
                         {
                              "ID": "201133R018",
                              "MaterialDefinitionID": "34003198",
                              "Location": {
                                   "EquipmentID": "ESP1",
                                   "EquipmentElementLevel":
"Site",
                                   "Location": {
                                        "EquipmentID": "C523",
                                        "EquipmentElementLevel":
"StorageZone"
                                   }
                              },
                              "StorageLocation": [],
                              "Quantity": {
                                   "QuantityString": "0.37",
                                   "DataType": "double",
                                   "UnitOfMeasure": "KG"
                              }
                         },
                         {
                              "ID": "201138R033",
                              "MaterialDefinitionID": "34003198",
                              "Location": {
                                   "EquipmentID": "ESP1",
                                   "EquipmentElementLevel":
"Site",
                                   "Location": {
                                        "EquipmentID": "C529",
                                        "EquipmentElementLevel":
```

```
"StorageZone"
                                        }
                                    },
                                    "StorageLocation": [],
                                    "Quantity": {
                                        "QuantityString": "14315.724",
                                        "DataType": "double",
                                        "UnitOfMeasure": "KG"
                                    }
                                }
                            ]
                        }
                    }
                }
}
```

Constants

Here you can find all the necessary constants for the API.

IoT Functions where (i) indicates Input and (o) indicates Output:

PType_Production = 0	(i) Production counter.
PType_Scrap = 1	(i) Scrap counter.
PType_Rework = 2	(i) Rework counter.
PType_Consumption = 3	(i) Consumption counter.
PType_Generic = 4	(i) Generic multipurpose signal.
PType_AvailProd = 5	(i) Indicates with 1 when there are elements at the entry of the process. This means that the process can work because it has elements to process ready at its entry.
PType_Saturation = 6	(i) Indicates with 1 when there are elements at the exit of the process. This means that the process can not work due to saturation at its exit.
PType_ProdPulse = 7	(i) Production pulse by the process. Not recommended, use PType_Production instead.

PType_ScraPulse = 8	(i) Scrap pulse by the process. Not recommended, use PType_Scrap instead.
PType_RewoPulse = 9	(i) Rework pulse by the process. Not recommended, use PType_Rework instead.
PType_OnOff = 10	(i) Forces a process to become working (when 1) or not working (when 0). Not recommended, use a production counter instead.
PType_OnOff_T = 11	(i) On off but with timing control (microstop, production minimum time, etc.).
PType_Good_S = 12	(i) Goods counter (send Scrap). This item will generate scrap by calculating the difference between a Production counter and this counter.
PType_Good_R = 13	(i) Goods counter (send Rework). This item will generate rework by calculating the difference between a Production counter and this counter.
PType_Good_SR = 14	(i) Goods counter (send Scrap or Rework). This item will generate scrap or rework by calculating the difference between a Production counter and this counter.
PType_Sched_R = 15	(o) Becomes 1 when the Scheduling button is flashing in red, which means that there are orders/tasks which are already delayed.
PType_Sched_G = 16	(o) Becomes 1 when the Scheduling button is flashing in green. Not in use.
PType_Sched_Y = 17	(o) Becomes 1 when the Scheduling button is flashing in yellow, which means that there are orders/tasks which are about to be delayed.
PType_Produ_R = 18	(o) Becomes 1 when the Results button is flashing in red, which means that the process should have produced results.
PType_Produ_G = 19	(o) Becomes 1 when the Results button is flashing in green. Not in use.
PType_Produ_Y = 20	(o) Becomes 1 when the Results button is flashing in yellow, which means that the process should produce results soon.
PType_Statu_R = 21	(o) Becomes 1 when the Statuses button is flashing in red, which means that the process has failures which are not justified, and they are about to be blocked.

PType_Statu_G = 22	(o) Becomes 1 when the Statuses button is flashing in green. Not in use.
PType_Statu_Y = 23	(o) Becomes 1 when the Statuses button is flashing in yellow, which means that the process has failures which are not justified.
PType_Consu_R = 24	(o) Becomes 1 when the Consumption button is flashing in red, which means that the process has consumptions which are not justified, and they are about to be blocked.
PType_Consu_G = 25	(o) Becomes 1 when the Consumption button is flashing in green. Not in use.
PType_Consu_Y = 26	(o) Becomes 1 when the Consumption button is flashing in yellow, which means that the process has consumptions which are not justified.
PType_Autoc_R = 27	(o) Becomes 1 when the Autocontrol button is flashing in red, which means that the process has pending autocontrol tasks which are urgent and about to be blocked.
PType_Autoc_G = 28	(o) Becomes 1 when the Autocontrol button is flashing in green. Not in use.
PType_Autoc_Y = 29	(o) Becomes 1 when the Autocontrol button is flashing in yellow, which means that the process has pending autocontrol tasks.
PType_SPC_R = 30	(o) Becomes 1 when the Quality SPC button is flashing in red, which means that the process has pending and urgent quality measures to be taken.
PType_SPC_G = 31	(o) Becomes 1 when the Quality SPC button is flashing in green.
PType_SPC_Y = 32	(o) Becomes 1 when the Quality SPC button is flashing in yellow, which means that the process has pending quality measures to be taken.
PType_SPC_A = 33	(o) Becomes 1 when there is an SPC Quality alert for the process.
PType_LifeBit = 34	(o) Becomes 1 when the edinn IoT
PType_Justify = 35	(i) Justifies a failure when this signal is activated.
PType_Justify_A = 36	(i) Justifies all the current failure, not just while the signal was activated.
PType_ResChange =	(i) Changes the result to the one indicated, when this signal is

37	activated.
PType_OEE_R = 38	(o) Becomes 1 when the process is RED in terms of OEE.
PType_OEE_G = 39	(o) Becomes 1 when the process is GREEN in terms of OEE.
PType_OEE_Y = 40	(o) Becomes 1 when the process is YELLOW in terms of OEE.
PType_SubsUnattended = 41	(o) Becomes 1 when the process has a unattended subscription that has already been produced.
PType_StatusActive = 42	(o) Becomes 1 when the process is in a certain status.
PType_Measure_Res = 43	(i) Numeric data which will be inserted as a measure which belongs to a program which is related to a result.
PType_Result_Id = 44	(i) Changes the current result to the content of this signal.
PType_Measure = 45	(i) Numeric data which will be inserted as a measure which belongs to a program but is not related to a result.

IoT Type

- TOPICS_None = 0
- TOPICS_Counters = 1
- TOPICS_Signals = 2
- TOPICS_UDL = 3

IoT Vendors

- MON_DRIVER_Manual = 4
- MON_DRIVER_Vision = 5
- MON_DRIVER_OPCDA = 8
- MON_DRIVER_VisionIP = 9
- MON_DRIVER_edinnHIP = 11
- MON_DRIVER_OPCUA = 12
- MON_DRIVER_NONE = 255

Errors

Error codes and description:

0	Success
1000	Some parameters are missing
1001	Company name not found
1002	Can\'t store the session credentials
1003	Can\'t store record
1004	Incorrect credentials
1005	No data available
1006	Element not found
1007	Session not stablished
1008	Process list is empty for the current user
1009	User must have administrative role to perform the operation
1010	The date entered is not valid. Please enter a date between start and end time
1011	Recourse not found
1012	Process not found
1013	Calculation type not found
1014	Production could not be calculated
1015	Incorrect status type
1016	Could not GetFields
1017	Could not delete record
1018	Monitor record not found
1019	Company parameter must be provided
1020	Company parameter must be numeric and have 10 digits
1021	Pass parameter must be provided
1022	Password does not match
1023	Process parameter must be provided for this message

1024 Process parameter don\'t exist

1025 Host parameter must be provided for this message

1026 Url parameter must be provided for this message

1027 Endpoint parameter must be provided for this message

1028 Item parameter must be provided for this message

1029 Function parameter must be provided for this message

1030 Function parameter must have a value between 0 and 40

1031 Type parameter must be provided for this message

1032 Type parameter must have a value between 1 and 3

1033 Threshold parameter must be provided for this message

1034 Threshold parameter must have a value between 0 and 100

1035 Minimum Cycle parameter must be provided for this message

1036 Minimum Cycle parameter must have a value between 0 and 100

1037 Vendor parameter must be provided for this message

1038 Vendor parameter must have a value between 0 and 12 or equal to 255

1039 Dead Band parameter must have a value between 0 and 100

1040 Activate parameter must be provided for this message

1041 Activate parameter must have a value between 0 and 1

1042 Reset every parameter must have a value between 0 and 3200

1043 Event parameter must have a value between 100 and 255

1044 Send description parameter must be provided for this message

1045 Send description parameter must have a value between 0 and 1

1046 Storedb parameter must be provided for this message

1047 Storedb parameter must have a value between 0 and 1

1048 User parameter of item must be provided for this message

1049 Password parameter of item must be provided for this message

1050 Variable Number parameter must be provided for this message

1051 Variable number parameter must have a value

1052 Variable number is not correct

1053 Monitor log record could not be inserted

1054 Id parameter must be provided for this message

1055 Message could not be sent

1056 Recourse process assignment could not be created

1057 Process status record could not be created

1058 Process could not be created

How to's

Configure OEE Targets

In this procedure we will detail the steps to follow to configure the targets of ratios OEE, PE and OCE for the areas.

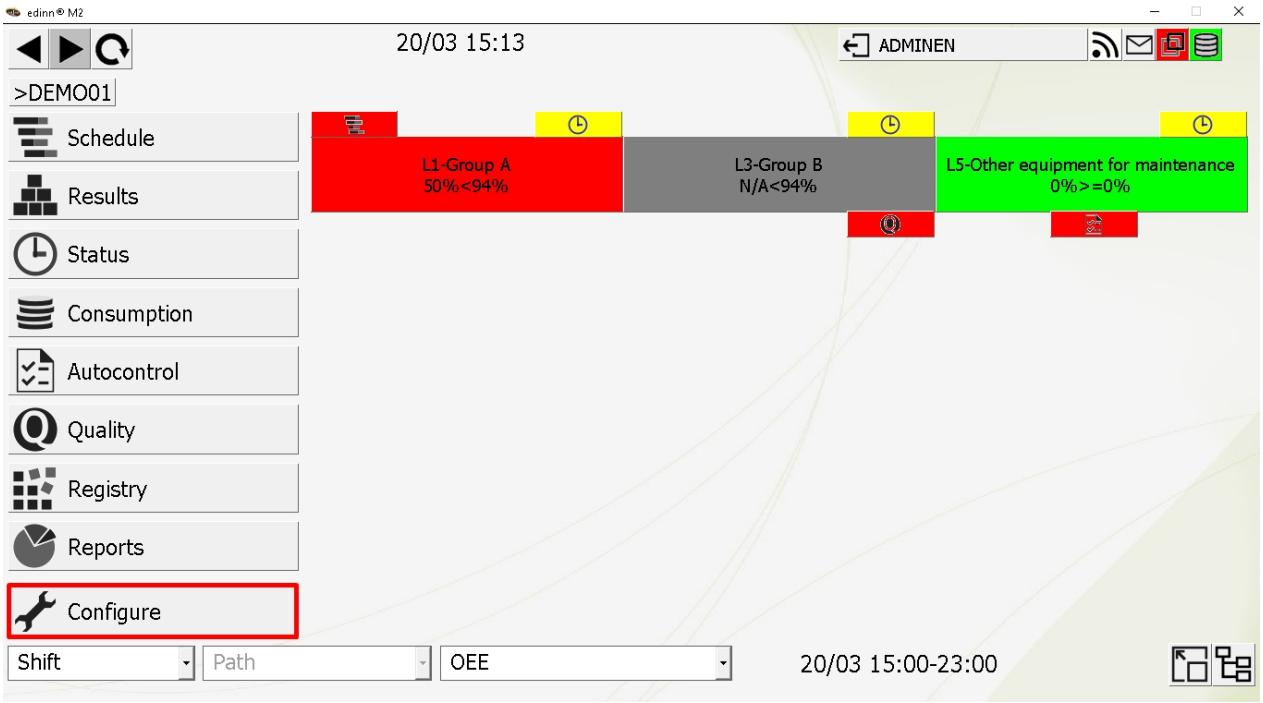
If we do not perform the following steps, the system will take for the targets of the areas, in this order:

1. The targets indicated in the Process-Result relation.
2. The targets indicated in the Process.

Step 1. Configuration

Proceed:

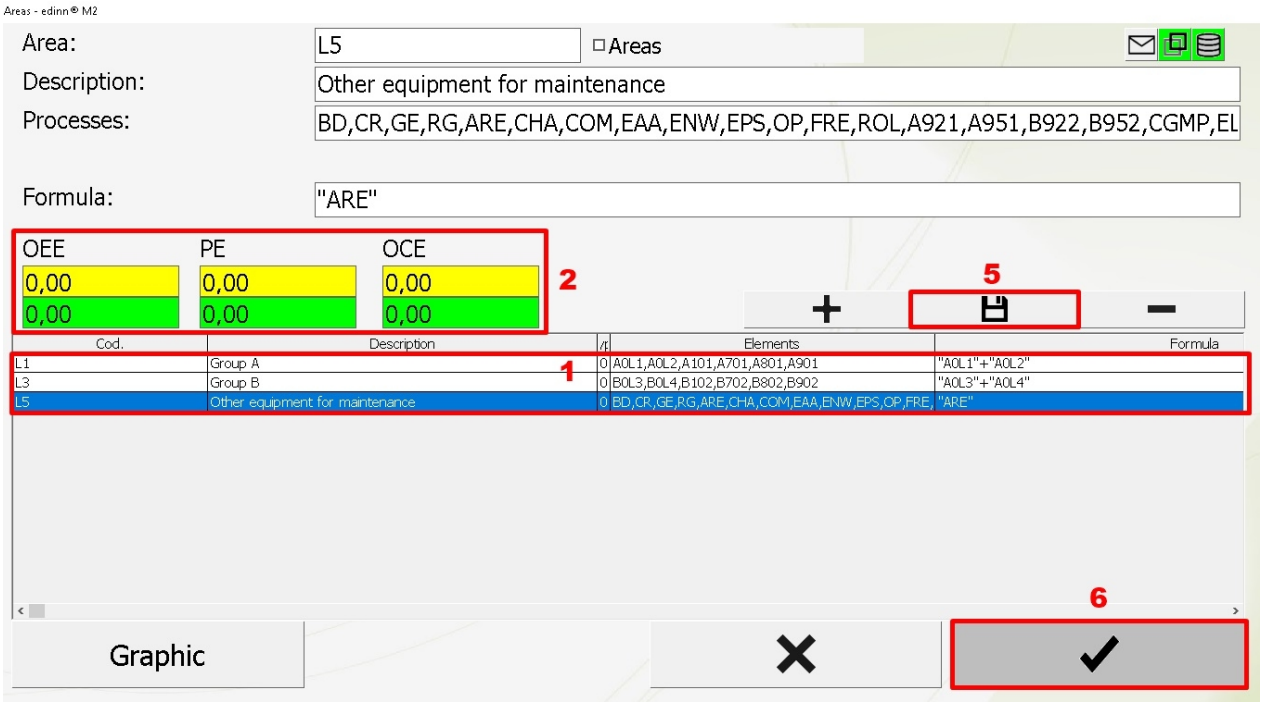
1. Click on the  button from the Main Window of the **terminal**.
2. **CLICK TO CONFIGURE > Areas**

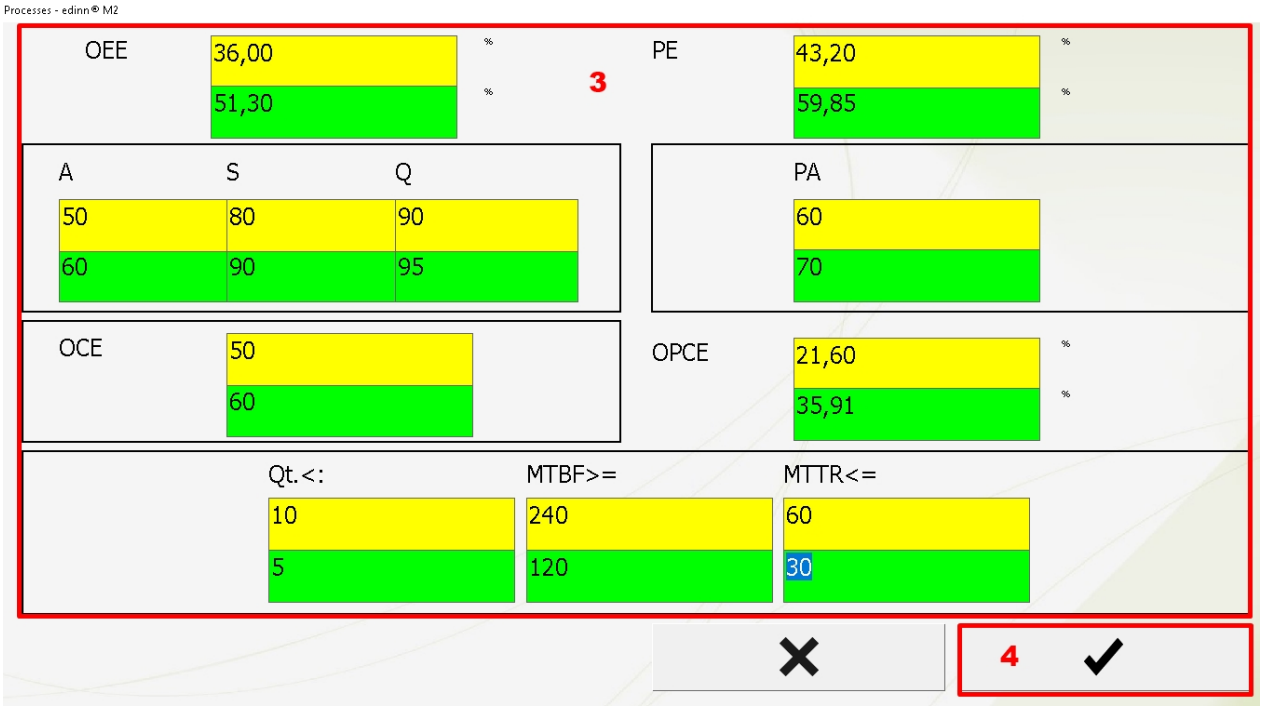


Step 2. Ratios configuration

You can see the order in the images below to configure the Ratios for an area.

1. Select from the table the area from which you want to set the ratios.
2. Click any of the green and yellow targets for the OEE and PE (is the same as the OEE but without considering idle statuses) and OCE ratios for the areas.
3. Configure the parameters of the Ratios Window and press ✓.
4. Save the changes by clicking 💾 and then press ✓.





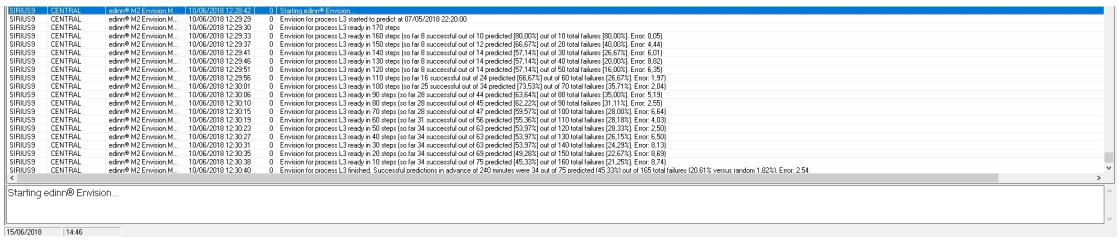
Failure prediction

How to configure (Beta)

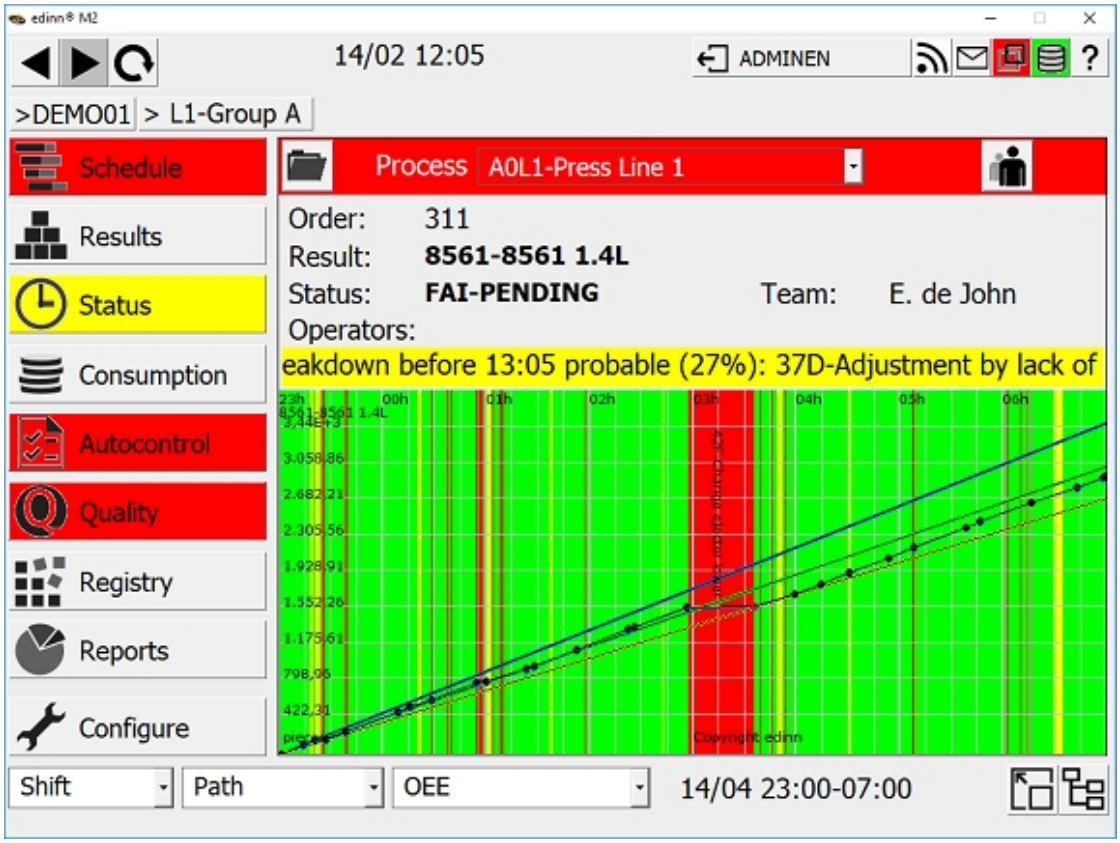
NOTE: The Envision Module (Artificial Intelligence) is currently in Beta version and subject to changes. Therefore, the information indicated here could have differences with your version.

Configuration for the prediction of failures is as follows:

- Configure the server as indicated here.
- From the console of the server, if the log detail is in very high detail, you can view the logs of the system which will generate log information as shown below, indicating performed predictions, success rate and other useful information for the fine tuning of this module.

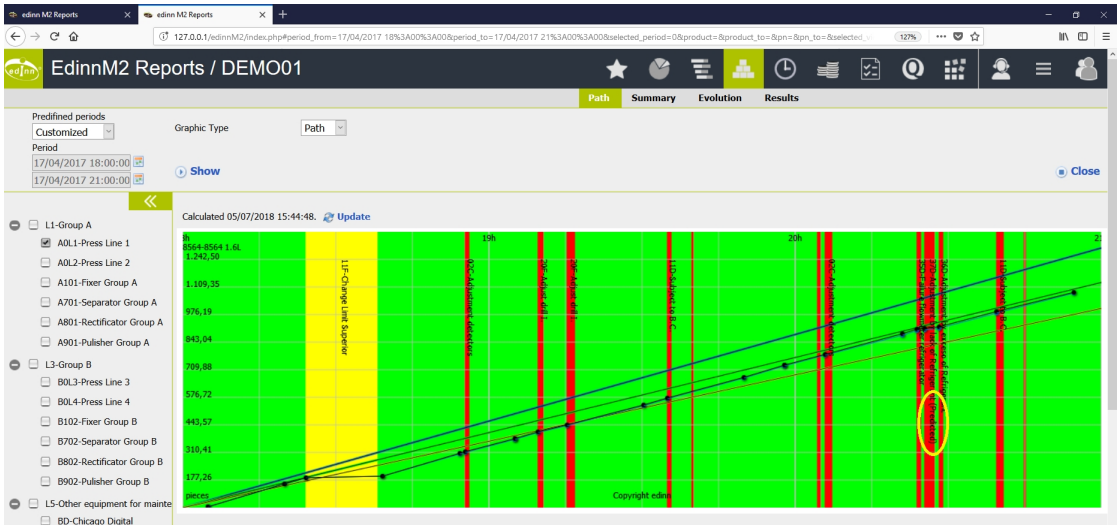


- On the terminal application, it is not necessary to configure anything. Once the option "Notify" has been activated in the console and the services have been restarted, then, when the Envision Module will perform a prediction in the actual time, a yellow warning will appear in the terminal application, as shown below:



The user will be able to mark the alert as viewed and hide it by clicking on the yellow label and confirming.

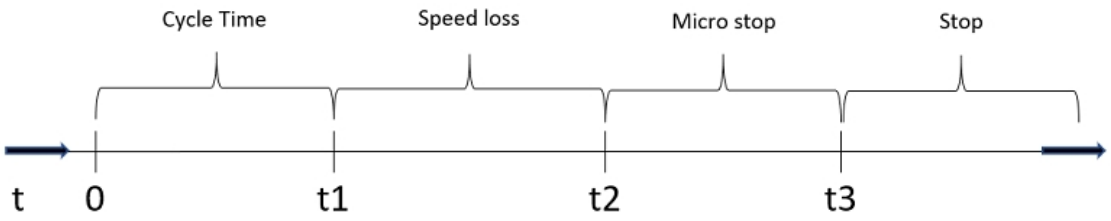
- Additionally, in the path graphic, failures which have been predicted will be marked with "Predicted", as shown in the image below.



Configure Times

In this document we will explain the necessary configuration to set up the **cycle time**, what is **speed loss**, what is **micro stop** and what is a **stop**.

For a better comprehension, please see the graphic below:



Where:

- **Cycle time (t1)**: it is configured by indicating every how much time we obtain how many results, in processes configuration and PSR relation configuration. The system will take the data configured in PER relations. Configuration in Processes is only for when new relations with new results are being created, that the system takes this data as default. It is important to note that you can also:
  - Configure this data in an order and the system will take it prior to PER relations.

    - Make this times depending on calculations of fields, which can be automatic or manually inserted by users.

- **Threshold (t2)**: it is configured as a percentage in the server console, monitor tab, for every total production counter. It is the percentage of speed that we accept to consider that the process is in production, although at a lower speed. In other words, if the speed of the process is lower than the nominal speed, stablished by the combination of units and cycle time, multiplied by this percentage, then the process will be considered as not in production.

- **Micro stop (t3)**: it is configured in seconds in the field **Stop Minimum** in processes configuration. It is important to notice that you should also indicate in processes configuration the status that the system will take as a micro stop.

- **Stop**: every stop which exceeds the t3 time, will be considered as a stop (FAI, from failure) and the system will search for an automatic or manual justification.

Therefore, every cycle time which is:

- **Longer than t1 and shorter than t2**: will be considered inside the speed loss type.

- **Longer than t2 and shorter than t3**: will be shown as a micro stop and considered inside the availability loss type.

- **Longer than t3**: will be shown as a stop and considered inside the availability loss type.

Example:

Let us suppose that for a certain process, status and result, we have the best cycle time (also known as nominal capacity) of 60 seconds for 1 piece, and we want:

- Until 90 seconds, the system should consider a speed loss.

- Every stop which is less than 3 minutes, should be considered as a micro stop.

The values that we should configure for this example are:

- **t1**: CycleQ = 1 (piece) and CycleT = 60 (seconds).

- **t2**: As we want to admit 90 segundos, the threshold should be 60/90 = 0,6666 = 66,66%.

- **t3**: As we want that every stop which is less than 3 minutes to be auto justified as a micro stop, then we will type 180 seconds.

Activate Big Data

In this document we will explain how to **activate Big Data** so that it can be used by different parts of the system: reports, dashboards and more.

WARNING: once you activate Big Data, the server console will generate a date and time which indicates to the system since when should this information be considered. Some critical parts of the system, like for example the reports, will start using this information since that date and time, and therefore it is recommended to not change this date and time unless you know how to correctly prepare the data for other periods.

1. Activate Big Data

Some systems have already activated the Big Data functionality. To check if your system has Big Data activated, check if the **Big Data Autocalc** option is activated in the Calculations tab in the server console.

If it is activated, you do not need to follow this procedure as your system should be using Big Data automatically since the date and time indicated in the **Since** field.

If it is not activated, **mark** the **Big Data Autocalc** option. After doing so, the date and time Since which is in the same section will indicate since when the system will start using Big Data. Do not change this date and time unless you know how to correctly prepare Big Data for previous periods.

2. Restart Central Service

Stop and restart the Central Service from the Activity tab of the Console tab. The Big Data task will be shown working in the Activity list. If the task is not being showed, check the logs to find the cause.

Data preparation for previous periods

It is possible to prepare Big Data for previous periods. If you need this functionality, please contact your edinn support.

Configure Incentives

In this section we will see how to configure **incentives**.

For the calculation of incentives it is used the Activity Ratio and a Consumption Type of incentive type.

Step 1: Activity configuration

It is necessary to configure the system to correctly calculate the activity ratio of every person. Activity ratio can be configured in:

- **Status configuration**: sets up an activity percentage for every status. This value is used as a template for the activity than can be configured in PSR relations, in the relation of a process and a status.
- **PSR relations configuration in Statuses:** allows to configure an activity percentage for every status when it is happening in a process.
- **PSR relations configuration in Results**: allows to configure an activity percentage for every status, when it is happening in a process which is producing a specific result.
- **Resources configuration**: allows to configure the target of activity ratio for every human resource (personnel).

To configure any the previous:

1. Click on the configuration button from the main window of the **Win32 Terminal**.
2. Select the desired option from the above in the drop down list **CLICK TO CONFIGURE**.

Check if you have correctly configured the **Activity Ratio** by obtaining an Activity Report of the personnel, checking if the obtained activities adjust to your needs.

Step 2: Create a Consumption Type as incentive

In this step you will create a Consumption Type, **marked as an incentive**.

1. Click on the configuration button from the main window of the **Win32 Terminal**.
2. Select the option Consumption Types in the drop down list **CLICK TO CONFIGURE**.

Step 3: Assign to persons the Consumption Type of type incentive

In this step you will assign the Consumption Type, created in the previous step and marked as incentive, to all the human resources (persons) which you need to calculate incentives.

1. Click on the configuration button from the main window of the **Win32 Terminal**.
2. Select the option Resources in the drop down list **CLICK TO CONFIGURE**.

From this window, assign an incentive to every person through the theoretical consumptions button, where you will go to another window where you will find the consumption type incentive that you created in the previous step, and you will be able to assign to the selected resource (person) an incentive by temporary rate.

Step 4: Check the incentives in the Activity Report

In this step we will check the calculations of activity ratios and incentives and how they are shown.

To do this, launch the Activity Report for different areas, processes and persons and check if everything is as desired.

Weird symbols or blanck buttons in the terminal

If you see weird symbols on the **buttons or blank buttons in your edinn M2 win32 terminal**, this is normally because the font **edinnM2Icons** was not installed correctly.

To solve this problem, the most frequently solution is (if one step does not work, continue with the next of the following):

1. Force automatic or manual update of the terminal.
2. **Restart your computer**: in some cases, this solved the problem.
3. **Install by double clicking** the font file **edinnM2Icons**, which should be:
    1. In the default windows fonts path: **C:\Windows\Fonts**
    2. Or where the edinn M2 win32 terminal is normally installed: **C:\Program files (x86)\edinn\edinnM2\client**

Need help? 

Recommendations

Recommendations

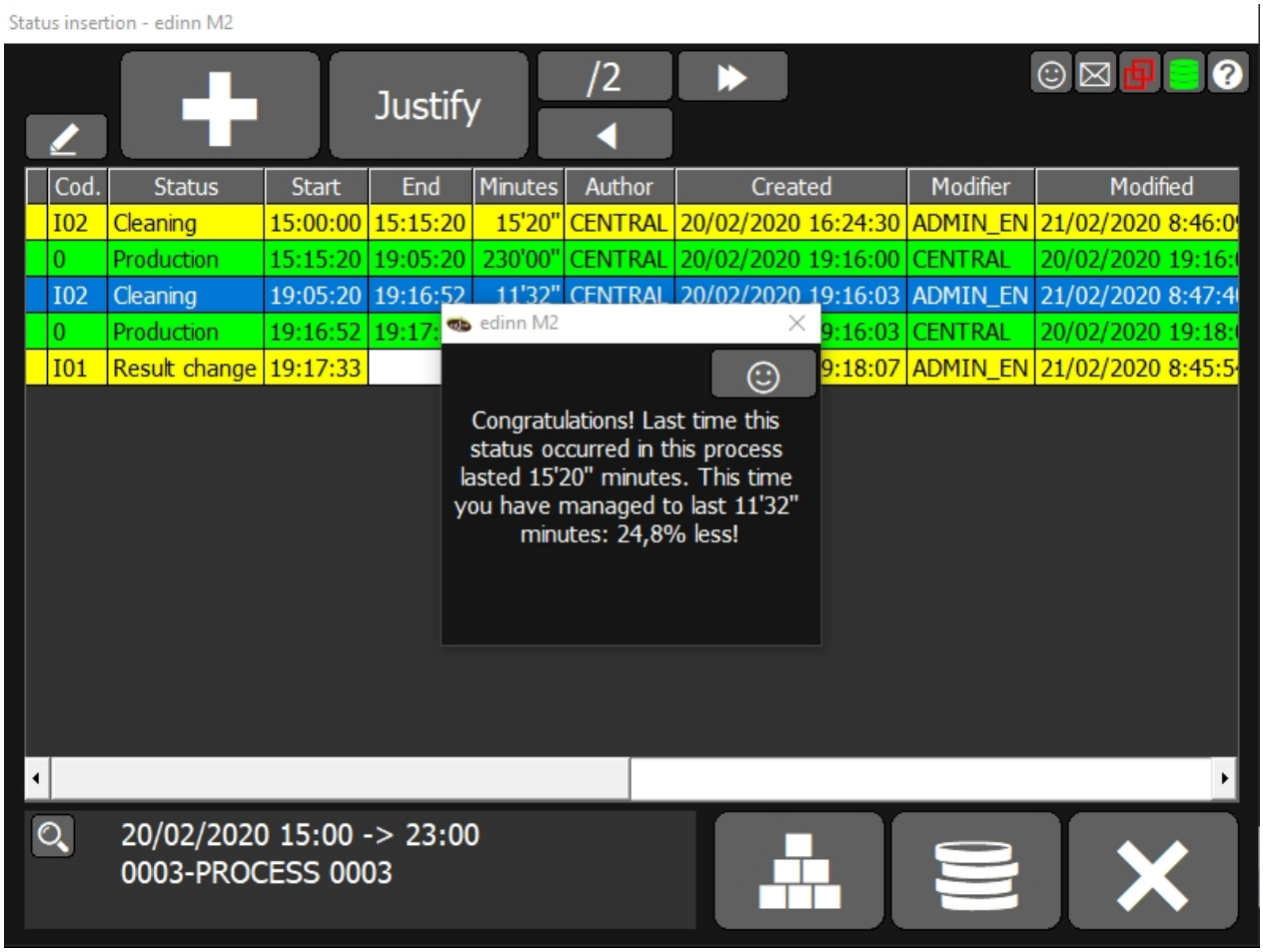
The purpose of edinn's recommendations is to provide the user with useful information so that he can do his job better. To provide this information, the edinn platform uses all its capabilities: calculation of KPIs and trends of OEE and other, Big Data and Artificial Intelligence. Instead of keeping all this complex and advanced information in isolated silos, the edinn platform makes it available to the user in an easy way and exactly when needed.

Current version provides:

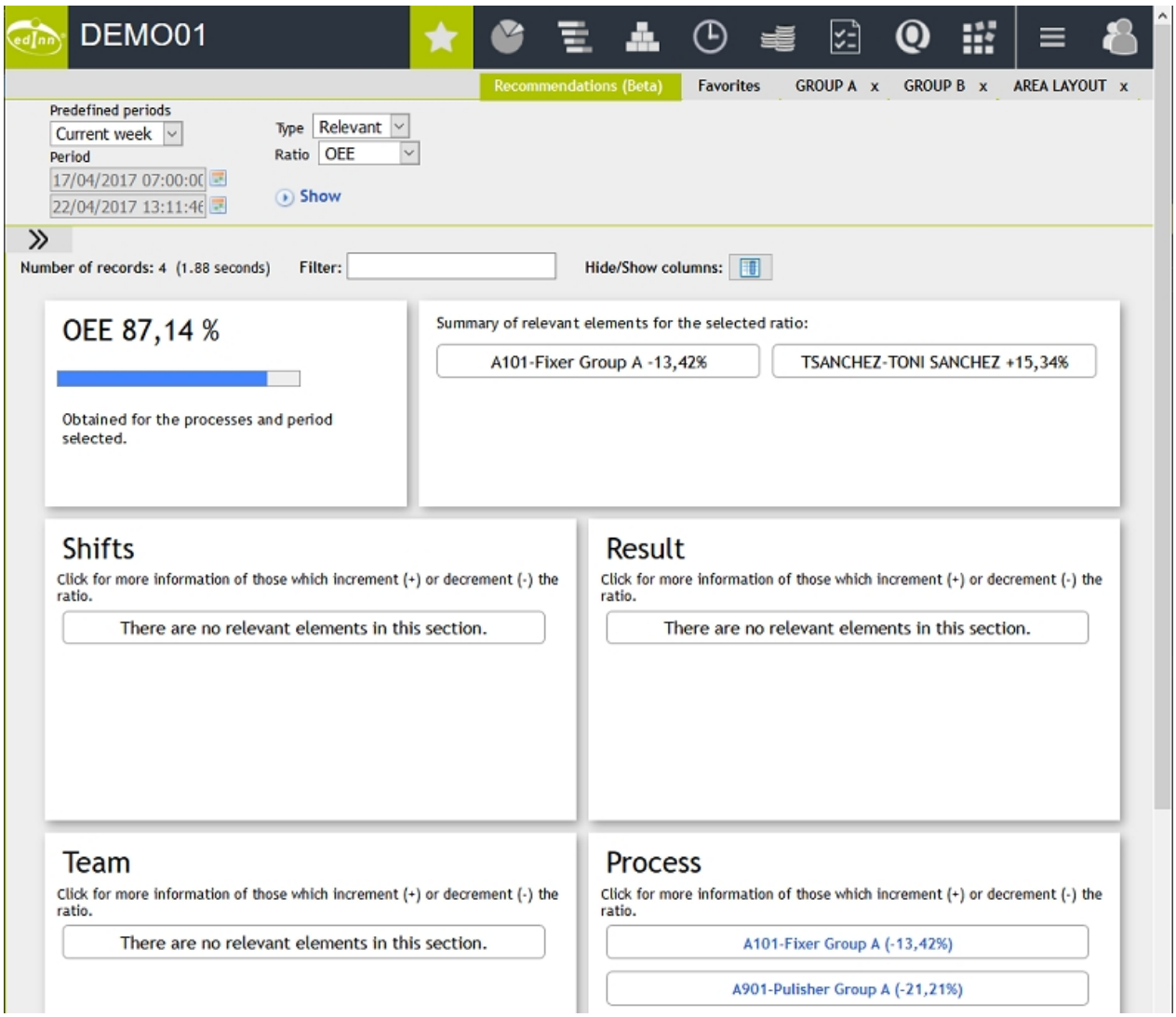
- **Recommendations in the terminal**: they are associated to certain records and when

the users clicks on them, an **icon** indicating the type of recommendation will be **shown on top right of the window**, **on the left of the icon which notifies of pending messages**. The user can **click**, as many times as he needs, **on the icon on the top right of the screen to navigate to the detail** of the recommendation. In addition, all recommendations can be viewed in the registry, as they are events of the platform. Recommendations are automatically updated as the work continues. To avoid overloading the user, recommendation windows are only shown once per user session and type; and it is not necessary to accept these windows as they disappear by time.

- **Recommendations on Statuses**



- **Recommendations on Results**
- **Recommendations on Reports**: for more information please see the Recommendations report.



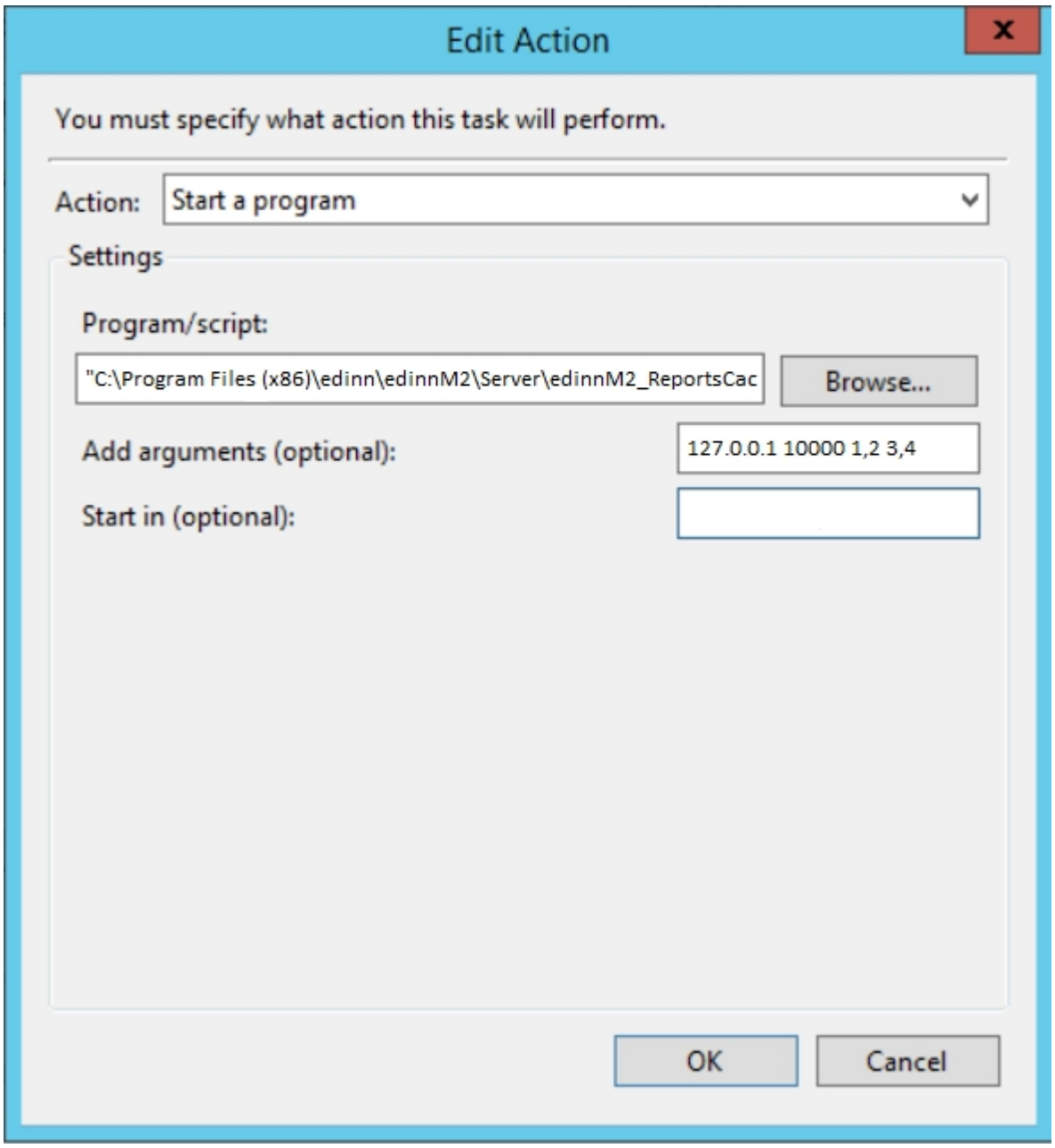
Prioritize favorites

Sometimes we need certain important favorite reports and/or dashboards to be refreshed with a high frequency, whilst the rest of favorites are not as prioritaire. If we do not prioritize, then the important reports or dashboards will have to wait until all the rest are refreshed.

Then: **how do we prioritize reports and/or dashboards?**

Priority is set up by programming in Windows different tasks to refresh reports. This can be done from the reports tab in the console with the button to schedule in Windows the task to refresh (cache) reports. Please note that this button will create only 1 task per company, and therefore, you might need to create manually a new one.

By double clicking in the Windows programmed tasks you can add 2 parameters to it (once we finish there will be 4 parameters, but the first 2 will be there when we edit the task), separated by 1 space character, like this:



These 4 parameters are:

1. The IP address of the database: 127.0.0.1 in the image.
2. The IP Port of the database: 10000 in the image.
3. Optionally, the list of the IDs (separated by comas and without any space character between them) of the favorite reports that we want this task to refresh (ids 1 and 2 in the image). This way this task will only refresh the favorite reports which IDs are indicated in this parameter.
4. Optionally, the list of the IDs (separated by comas and without any space character between them) of the favorite reports that we want this task to NOT refresh (ids 3 and 4 in the image). This way this task will refresh all the favorite reports except those which IDs are indicated in this parameter.

These parameters are exclusive between them, this is, you cannot use both. Let us see a typicall example: if we want a programmed task to refresh with high frequency the favorite reports with IDs 1 and 2, and another task to refresh less frequently all the rest of the favorites, except the 1 and 2 which are more important, then we should create 2 tasks like this:

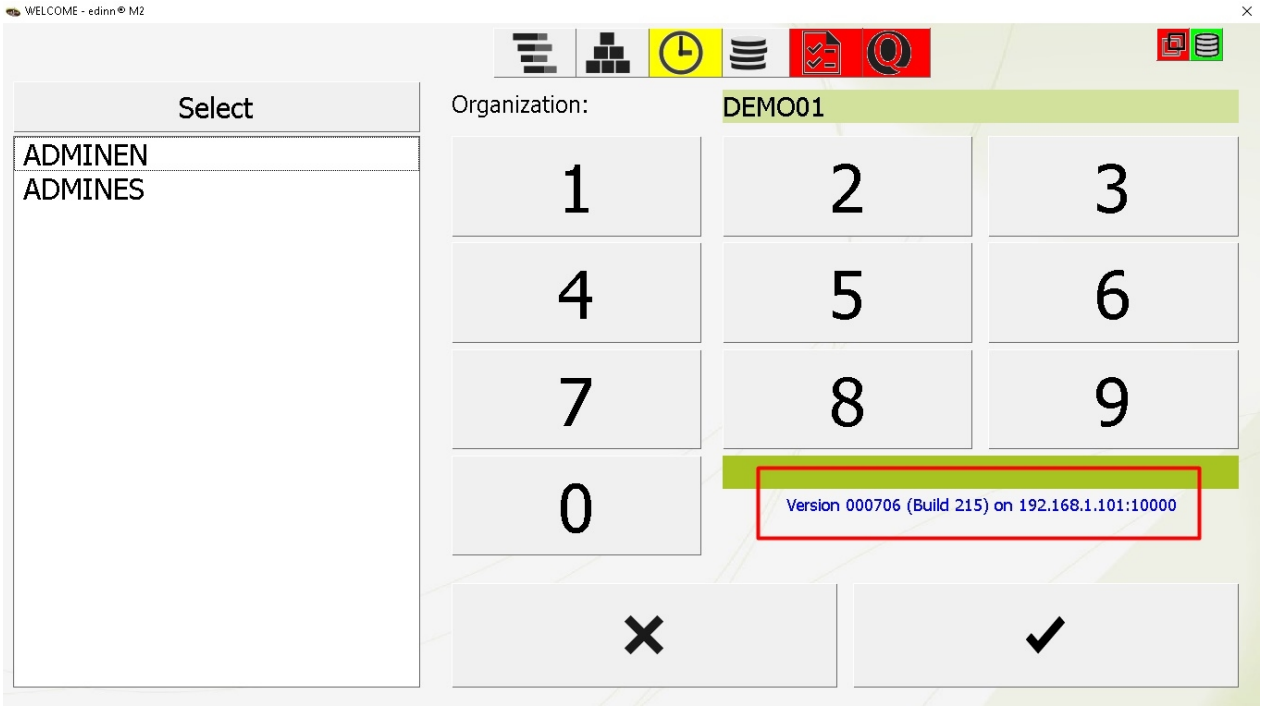
- One, with high frequency, and these parameters: `127.0.0.1 10000 1,2`
  This way this task will only refresh favorite reports with ID 1 and 2.
- Another, with less frequency, and these parameters: `127.0.0.1 10000  1,2`
  Please note that there is 1 (and only 1) space character in the third parameter, where in the previous task was 1,2. In this way, this task will refresh all favorite reports except those with IDs 1 and 2.

Know your version

There are 2 ways to know **your version**, depending of the type of login of your device:

In the **STANDARD login:**

In the **EASY login:**



Server Recovery

These procedures cover:

1. How to configure the recovery on two servers.
2. How to act in case of the fall of the master server.

WHY IS SO IMPORTANT THIS PROCEDURE?

Backup and recovery refers to the process of automatically **backing up** data to prevent **data loss** and setting up a slave server to speed up server availability in case of failure of the server.

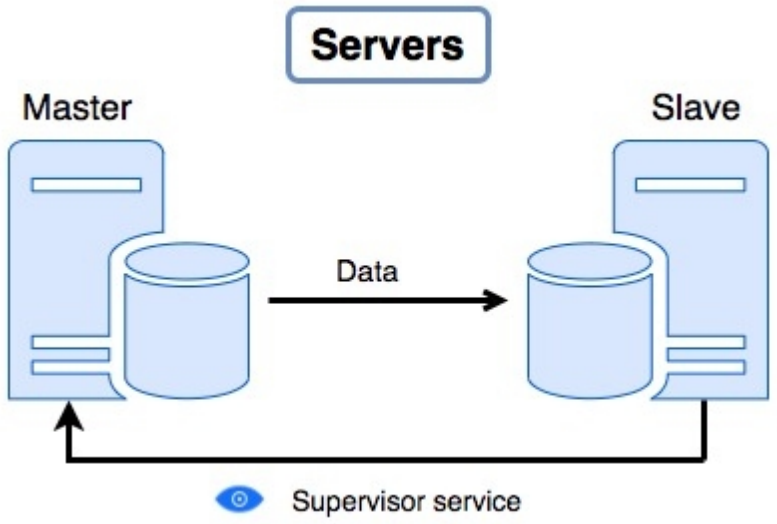
Potentially, the edinn recovery procedure could have more than one slave server, but that option is not covered in this documentation. Please request to edinn for more information.

This procedure is designed for **mission critical systems**. This includes servers and storage systems that, if they fail, have a significant impact on the operation of the company. However, the total recovery of the information is not guaranteed, neither is that no data will be loss, but a **reduction of the loss** produced is ensured.

HOW?

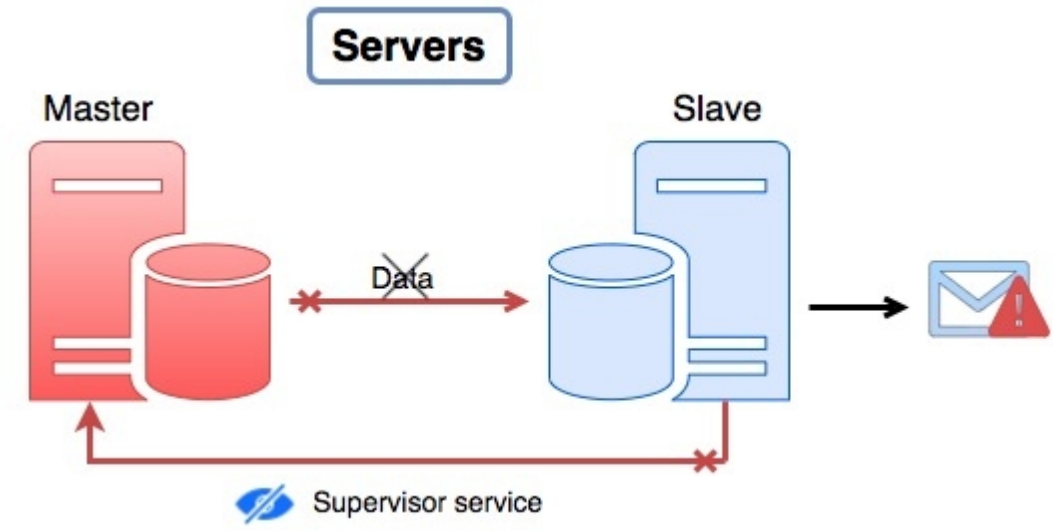
By configuring two **identical servers** so that one acts as a **master**, and another as a **slave**. The slave has an initial backup database of the master server with the same structure and data. This database will be filled with the information provided by the

master server on an automatic and continuous way.



HOW DOES THE SLAVE KNOW THAT THE MASTER HAS FAILED?

The slave has only one edinn **service** started: the **supervisor**. This service will be **constantly** (every minute) **checking** the connection and the correct operation of the master. In case of a failure, it will send an **e-mail** to the person in charge (that is why the configuration of the communication is so important) indicating the failure of the server or the communication with it, and will write the result into the windows **event log**.



HOW TO ACT?

If the master server fails, or the communication with it:

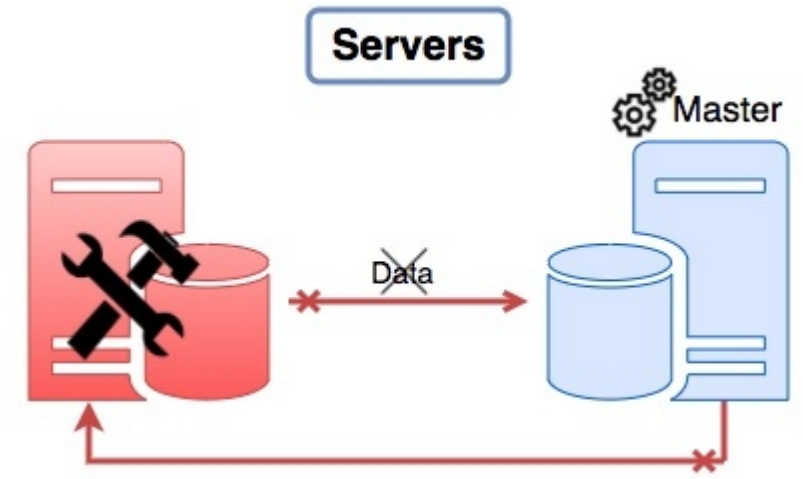
- The slave server will detect this failure and will **automatically become the master server**. Operation terminals will automatically connect to the slave server, now working as master.
- If the master server would recover and connect to network, it will detect that

slave server became master and the master server will stop its services and block all accesses to itself, to avoid that users connect to master server.

If you would like to **manually turn the slave server into the master server**, then, you should follow the following procedure:

1. In the slave server, change the recovery configuration and give it the role of master.
2. In the slave server, disable the recovery functionality, since there will be no more slave servers.
3. In the slave server, in the activity tab, start all the services as it has become the new master server.

NOTE: For a more detailed procedure, please see: HELP! How to recover?



WARNING: after a slave server has been turned into a master server, and operation terminals and applications have connected to the slave server, to restore the master server you must to follow another specific procedure named Restoring master server. It is not enough to just restore the role of the slave server to slave and reconnect operation terminals and application to the master server, as doing only this, could cause data loss.

HELP! How to manually Recover?

WARNING: Please remember that the slave server will automatically become master in case the master server has fallen or it is not connected to the network.Therefore, in the majority of the situations, you do not need to perform this procedure. Proceed only if you want to **manually turn the slave server into the master server**.

How to detect a failure

Once you have definitely confirmed that there is a failure in the master server, if you need to switch all terminals and applications to the backup (slave) server, follow this procedure.

Before initiating the recovery procedure, it is very important to assess the master server failure. There are many different possible reasons for a **failure**: the connection between servers, the connection with the database, the server itself, etc.

HOW TO CONFIRM IF THE FAILURE REQUIRES RECOVERY:

1. First, as the slave server is continuously connected and inquiring the master server, in case of a failure, the **supervisor service of the slave server** will:
    1. **Send an error mail** to warn about the need of change from slave to master.
    2. Write in the **event log of the operative system.**
2. If the previous has happened, or if you want to manually check, you can check the **connection** with the master server and its database.

> **NOTE:** For more information, please see: Introduction Server Recovery.

1. Manual check of the MASTER connection

There are 2 ways to confirm the failure of the master server. You should perform **BOTH A) and B)**:

A) **From the terminal**

Check if the connection of the master, with the server and the database, is correctly working:

1. Double click on the **edinn® M2** icon of your desktop or search in the Window search engine and execute "***edinn M2***".
2. In the **Login window** you will see 2 icons at the top:

3. If the icon  is **red**, means that connection with the **server** has failed. Contact

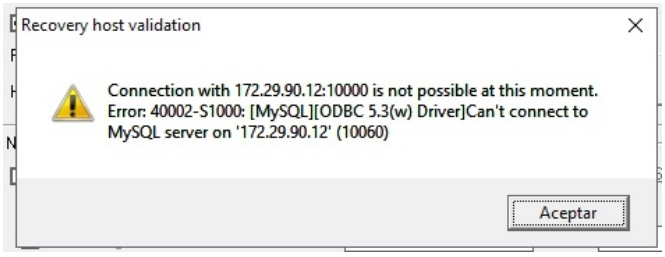
your **technical service** to try to reconnect.

4. If the icon  is **red**, means that connection with the **database** has failed. Then, contact your **technical service** to try to reconnect.

If any of icons is **green**, this **means that the failure has not been produced because of the master server and the recovery procedure should NOT be performed.** Instead, it is likely that an error in the connection between the slave server and the master server has happened.

B) **From the slave server**

1. Run **edinnM2_Console.exe** as **administrator** in the **slave** server.
2. Go to the **Supervisor** tab and, on the **Recovery** section, click **Test** to test the connection with master server.
3. Press "*Yes*" to proceed. If any error occurs, it will appear an error message like the one that follows:



If the connection can be stablished, a success message will appear like this "*Connection with HostI:Port successful!*". **This means that the master server is running, and the recovery procedure should NOT be performed.**

If, by the previous **A) AND B),** you have definitely confirmed that there is a failure in the master server, if you need to switch all terminals and applications to the backup (slave) server, follow this procedure.

2. Emails and Event log

When the supervisor service of the slave server detects a failure, sends an **ERROR eMAIL** and writes in the **EVENT LOG.**

There are **3 possible types**:

Some examples:

- If it is correctly working:
  - **Email:** "Run 100% updated"
  - **Event log:** will not show neither an error nor a warning.
- When the supervisor service is waiting to connect to the master database to obtain the status but the connection takes time:
  - **Email:** "connecting"
  - **Event log:** writes a connecting **warning** entrance.
- When there is not **consistency** between the master and the slave **database**:
  - **Email:** "Operations error: #### Could not execute Update_rows"
  - **Event log:** writes a **warning.**
- When the connection with the master **timed out.**
  - **Email:** "IO error: #### reconnecting to master"
  - **Event log:** writes an **error.**

How to proceed

WARNING: Please remember that the slave server will automatically become master in case the master server has fallen or it is not connected to the network.Therefore, in the majority of the situations, you do not need to perform this procedure. Proceed only if you want to **manually turn the slave server into the master server**.

WARNING: Recovery is an important process and can lead to data losses. Therefore, before starting the recovery process, **it is very important to be sure that there is a real failure in the master server**.

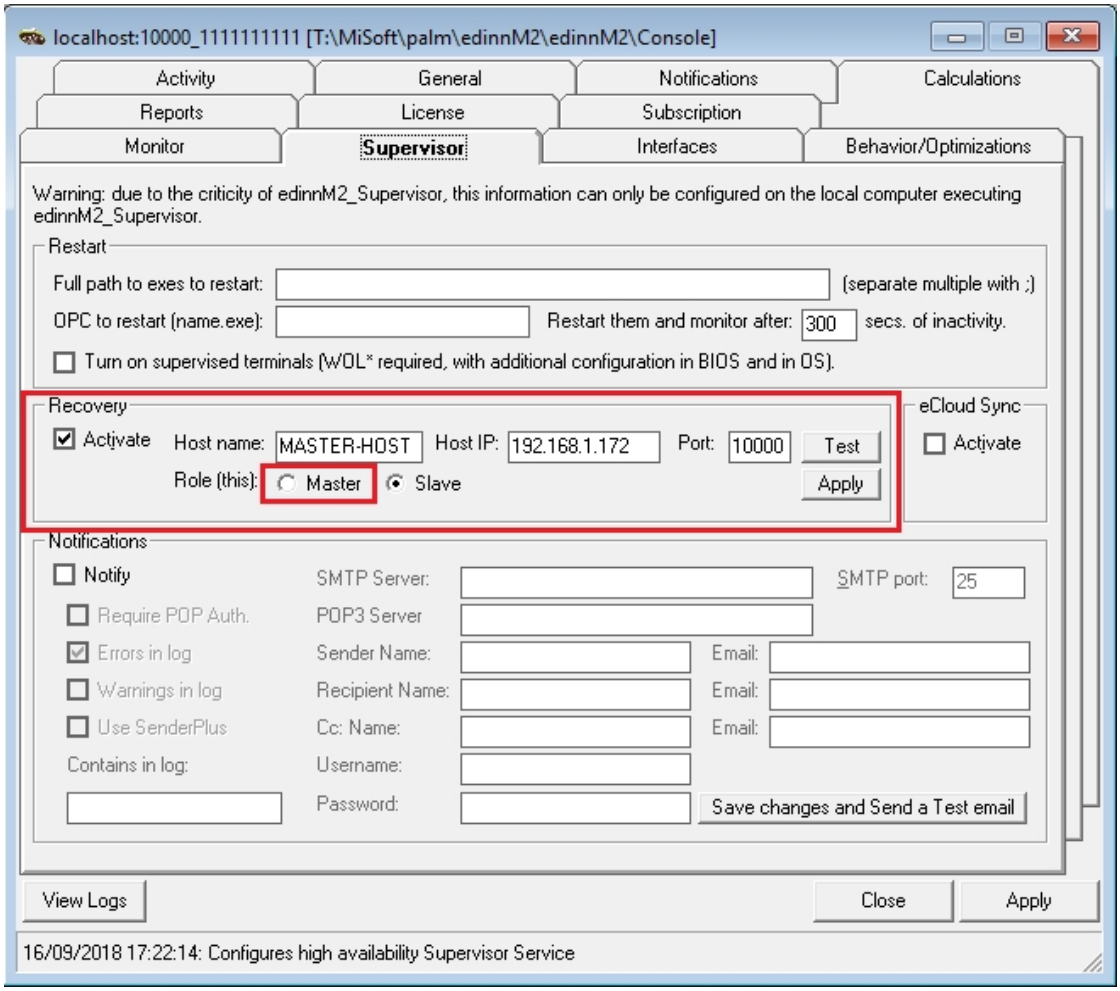
Once a failure has been detected, follow these steps to recover the normal operation:

Step 1. On the slave server, change the rol to master

The first step is to change the rol of the **slave server** to become a master, in order to replace the functions of the fallen master server:

1. In the **slave server**, run **edinnM2_Console.exe** with **administrator permissions.**
2. You can re-confirm that the master is not available, by testing the connection to the master server, in the **Supervisor** tab, **Recovery** section, by clicking the button **[Test] connection**.

   If failure of the master server is confirmed, continue with this procedure.
3. Open the **Supervisor** tab and change the **rol** from slave to **master**.



4. Press the **[Apply]** button of the Recovery section.

Step 2. On the slave server, disable Recovery and start all edinn services

1. In the same section, **disable** the Recovery (uncheck **[Activate]**), since you will change the slave to be a master, and there will not be any other slave.

2. Click on the bottom right **[Apply]** button of the Recovery section to apply the changes.

3. In the **Activity** tab, restart all the services by pressing [**Start service**] in **each of the services**, going from top to bottom (starting with the **Central**).

WARNING: Since the master server is now fallen, you do not have any recovery server available and it is recommended to **restore** the Master Server and **reconfigure** the Recovery again as soon as possible.

Once your master server will be correctly working and will be connected to the data network again, it will not be enough to just restore the role of the slave server to slave and reconnect operation terminals and applications to the master server, as **doing only this, could cause data loss**. Instead, you should proceed with the procedure to restore the master server.

Step 3. Some terminals could automatically connect to the slave (now playing as master) server. But for those users who report connection problems, inform them to connect to the slave server on their operation terminals and applications

Inform users on how to connect to the slave server and provide them with the **IP address and port** of the **slave server.**

END OF PROCEDURE

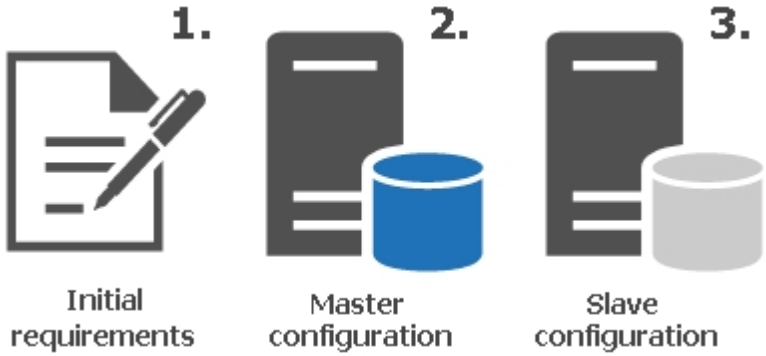
Setting up Recovery

In this guide the **configuration** for the recovery procedure will be explained.

NOTE: to understand the significance of the **Recovery procedure**, please see: Introduction

You will configure 2 **identical servers** so that one acts as a **master**, and another as a **slave.**

All these steps must be done for the recovery functionality to work properly. Complete the following steps in the following order:



1. Initial requirements

Before configuring the master and the slave server, you must **verify** that the following files are configured for **BOTH** servers:

Step 1. Check the "Binary Logging"

Binary logging is the method that the master uses to store the queries that the slave needs to replicate.

1. Go to pah
   C:\Program Files\edinn\edinnM2\Server\MySQL
   (it could be C:\Program Files (x86)\edinn\edinnM2\Server\MySQL in computers with 32 bits processor architecture).
2. Open the file **my.ini** in a text editor as administrator, for example: Notepad++.
3. Search in the file (CTRL+F) the method "***# Binary Logging***" and check the correspondance. If it does not exist, create it by copying the following parameters:

   ```
   # Binary Logging
   log-bin="log-bin"
   log-slave-updates=0
   binlog-format=MIXED
   log_bin_trust_function_creators=1
   expire_logs_days=7
   max_binlog_size=1G
   ```

4. Include this new entry in the **my.ini** file, putting **1 in the master and 2 in the slave**:

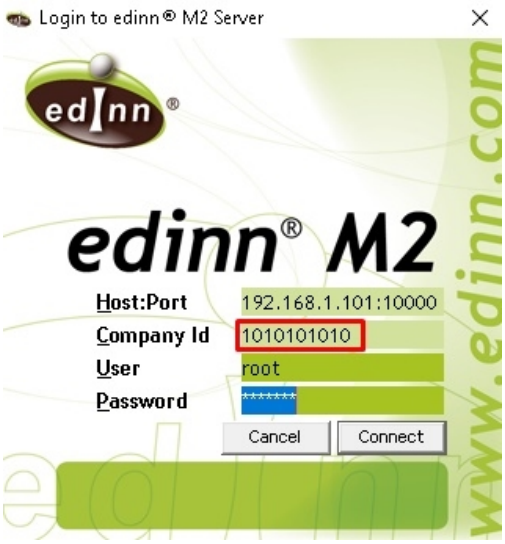
   ```
   # Server Id: Put 1 for the master and 2 for the slave
   server-id=1
   ```

5. Save the changes to the **my.ini** file.

Step 2. Check the "Replication" restrictions

To reduce the amount of data written in binary log and to prevent replication of data which needs to be unique for each server, configure the following parameters:

1. Go to path **C:\Program Files (x86)\edinn\edinnM2\Server** and execute, with administrator permissions, the program **edinnM2_Console.exe** (you can copy and paste to your desktop **a link** to this application as you will use it later, remember to mark it to be executed with **administrator permissions**). Copy the **Company Id** for later use.



2. Go to path
   C:\Program Files\edinn\edinnM2\Server\MySQL

3. Open the file **my.ini** in a text editor as administrator, for example: Notepad++.

4. Search in the file (CTRL+F) the method "***# Replication***". If it does not exist, create it by copying the following parameters. **Please remember to replace the #CompanyID# text by the company Id number copied in step 1**:

```
# Replication
slave_net_timeout=300
replicate-do-db=edinnm2_#CompanyID#
replicate-ignore-table=edinnm2_#CompanyID#.m2conf
replicate-ignore-table=edinnm2_#CompanyID#.m2logs
replicate-ignore-table=edinnm2_#CompanyID#.m2molo
replicate-ignore-table=edinnm2_#CompanyID#.m2reca
slave-skip-errors=1062
```

5. Save the changes to the **my.ini** file.

Step 3. Restart the edinn database service

1. From the Windows search icon, type **“Services”** and right-click the app returned and select **“Run as administrator”**.

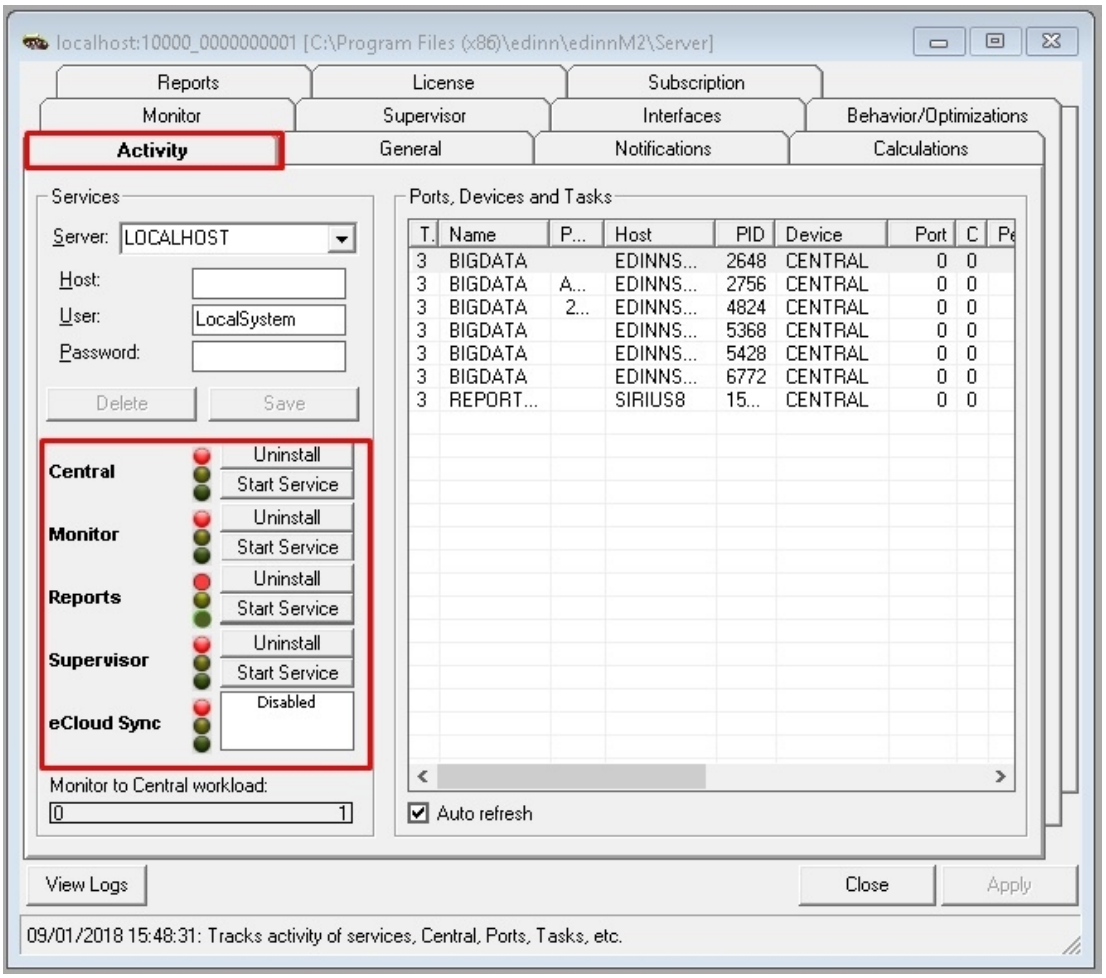
2. In the list, search "**edinnM2_MySQL**" and choose "**Restart**".

2. Master Configuration

Once completed the Initial requirements, the configuration of the **master** server will be explained in this section.

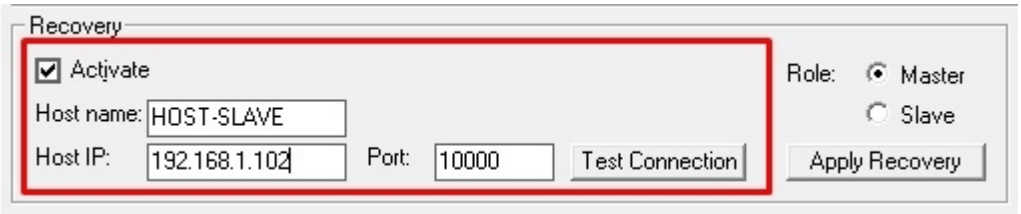
Step 4. Stop services in the master server

1. Go to path **C:\Program Files (x86)\edinn\edinnM2\Server** and execute, with administrator permissions, the program **edinnM2_Console.exe**.
2. In the **Activity** tab press **"Stop service"** in each of the services, going from bottom to top (starting with the **Supervisor**).



Step 5. Supervisor Configuration

Open the **Supervisor** tab. By default, recovery is inactive. Configure the following parameters:



Where:

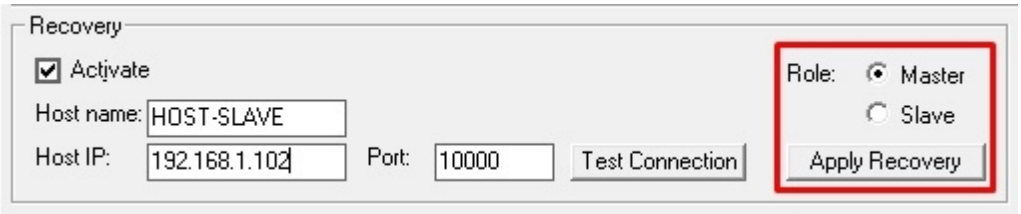
- **Host name:** is the remote host name. You can find it in the properties window of the slave server computer.
- **Host IP:** find it by writting the command "ipconfig" in cmd.exe in the slave computer.

Write here the information from the slave server:

1. **Switch on the Activate checkbox**: to enable the rest of the fields.
2. **Host name:** is the name of the slave server in the network, the one you are trying to connect to.
3. **Host IP:** is the IP of the slave server in the network.
4. **Port:** write '10000'.

Step 6. Apply Recovery

1. Select the **Master** role.
2. Press the "**Apply**" button of the Recovery section.

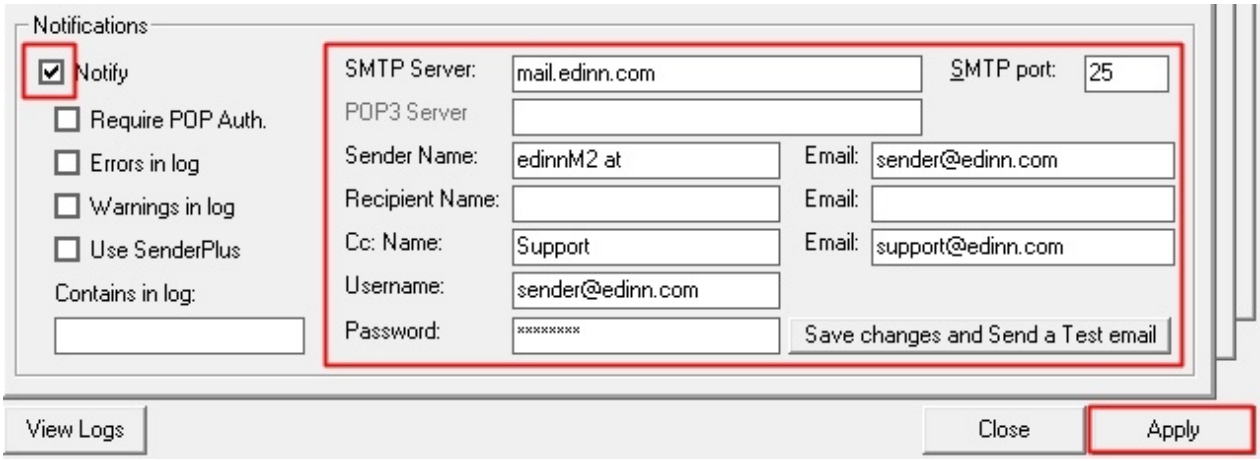


Step 7. Failure Reporting Configuration

To enable server to inform by e-mail about the failures and other issues, you must enable **Notifications** in the Notifications tab and fill in the fields as you can see in the image bellow.

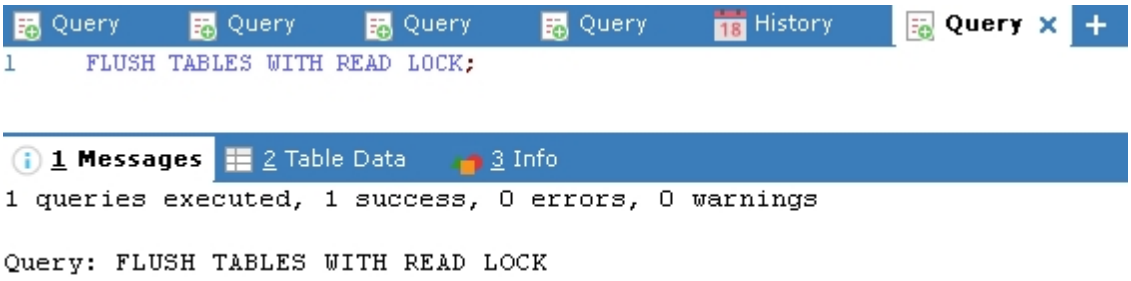
Remember to click on [**Apply**] button to save changes.

> **NOTE:** for more information of the Notifications configuration, please see: Notification.



Step 8. Lock access to database

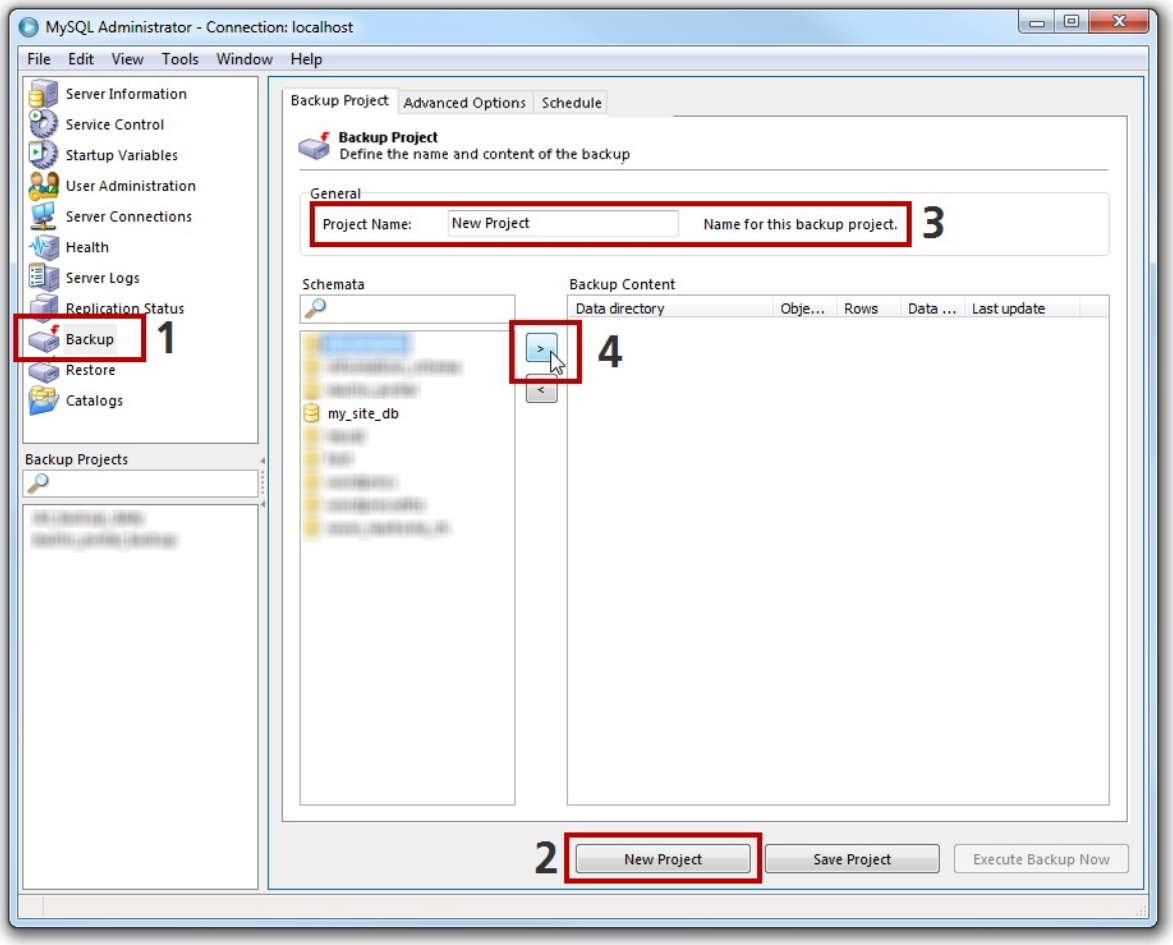
Run **SQLyog** as administrator and execute the following query: "**FLUSH TABLES WITH READ LOCK;**"



Step 9. Backup the database

Perform a database backup of the Server who is going to act as a Master:

1. Open **MySQL Administrator** as administrator.
2. Enter the backup tab, click on **“New Project”**, choose a Project Name and select the 2 databases from which you are going to perform the backup: the edinnm2 **and** the edinnm2_CompanyId database. Then click the > icon and **Save Project**.
3. Click "Execute Backup now" and wait for the backup to terminate. Copy the generated backup file and paste it in the Server who is going to act as a **Slave**. **Remember to not start the edinn services on the master server until indicated later**.



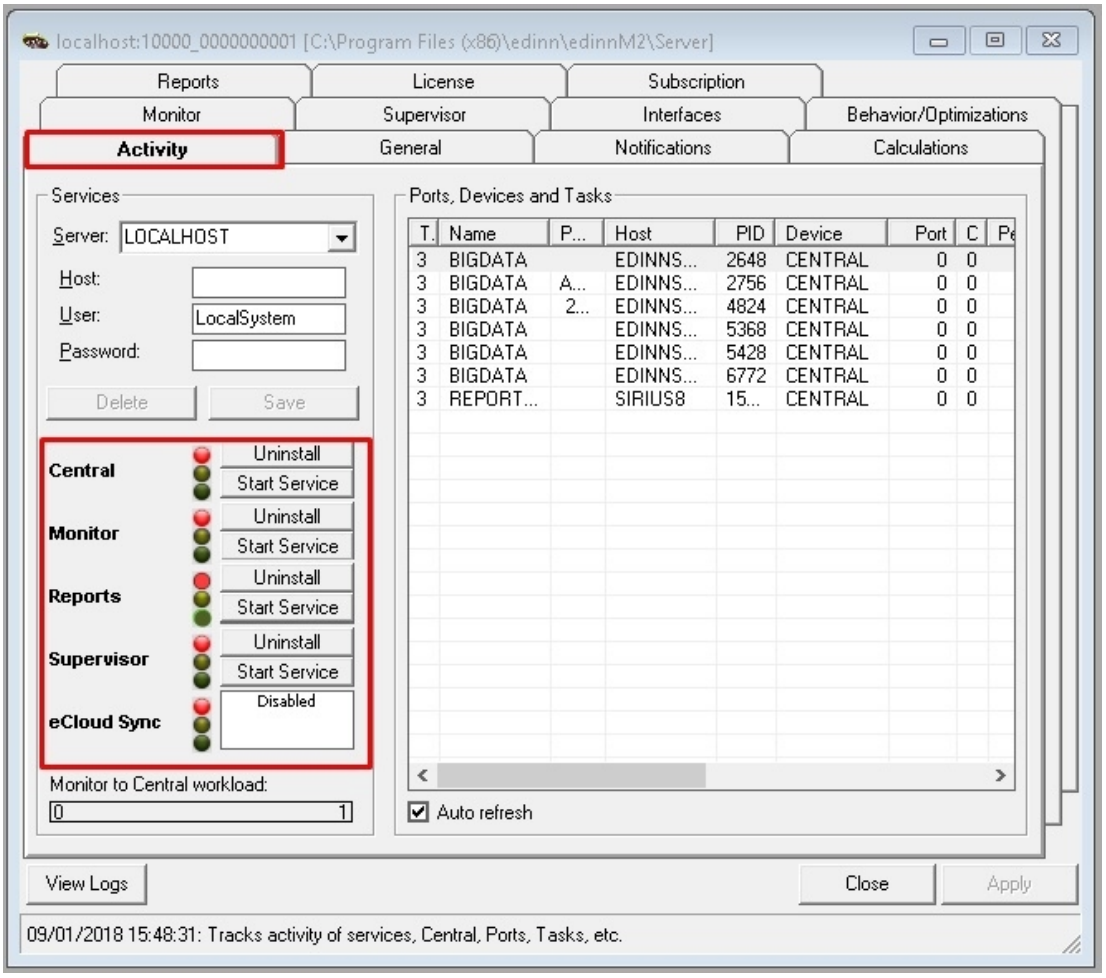
3. Slave Configuration

Once completed the Initial requirements and the configuration of the **master**, you can proceed with the configuration of the **slave** server which is explained in this section.

The slave server, as it is an edinn server, must be installed as an edinn local server following all the steps which are explained in the technical guide "**edinn Server Installation and Maintenance Guide**", which is available for installers in the edinn **Academy**.

Step 10. Stop services in the Slave server

1. Go to path **C:\Program Files (x86)\edinn\edinnM2\Server** and execute, with administrator permissions, the program **edinnM2_Console.exe**.
2. In the **Activity** tab press **"Stop service"** in each of the services, going from bottom to top (starting with the **Supervisor**)



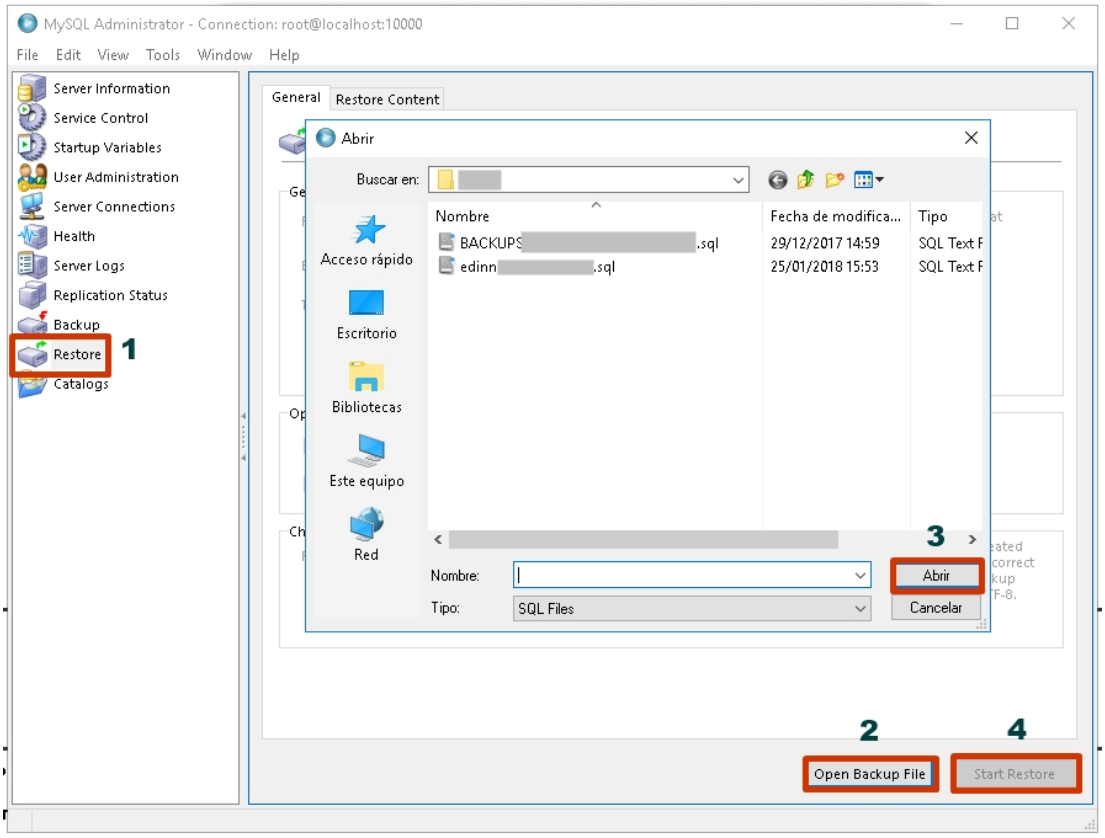
3. In the **License** tab, copy the license to a notepad or file, to be used later.

WARNING: do not forget to copy the License field content to a notepad of file. **If you do not provide this valid license later for this slave server, Recovery procedure will not work as this slave server will not run as a master server when needed**.

Step 11. Database restore.

Perform a database restore of the Master Backup that you have done in the previous step:

1. From the Windows search icon, type **"MySQL Administrator"** and right-click the app returned and select **"Run as administrator"**.
2. Enter the Restore tab, click on "**Open Backup File**", choose a Backup file and then click "**Open**"
3. Click on "**Start Restore**"

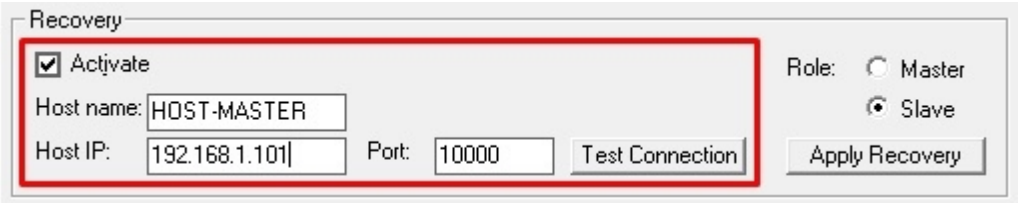


4. Once restored, use a database editor (for example the SQLYog application recommended before), and edit the table edinnm2.comp and **change, in all fields, the IP address of the master server by the IP address of this slave server.**

Step 12. Supervisor Configuration

If you have the edinnM2_Console.exe running, **close it**. Then **run again edinnM2_Console.exe** as **administrator** and:

1. Open the **Supervisor** tab. By default, recovery is inactive. Configure the following parameters:



Where:

- **Host name:** is the remote host name. You can find it in the properties window of the master server computer.
- **Host IP:** find it by writing the command "ipconfig" in cmd.exe in the master server computer.

Write here the information from the master server:

1. **Switch on the Activate checkbox**: to enable the rest of the fields.
2. **Host name:** is the name of the master server in the network, the one you are trying to connect to.
3. **Host IP:** is the IP of the master server in the network.
4. **Port:** write '10000'.

2. Open the **General** tab and in the Database configuration section, type the Update user (usually: root) and type the password of that user on the field below.
3. After performing the restore of the database in the previous step, the edinn license of this server has been replaced by the license of the master server. Please **paste in the License tab, in the License field, the license from the notepad or file where you copied it in the previous step**, or request for a new license to edinn.

WARNING: if you do not provide a valid license for this slave server, Recovery procedure will not work as this slave server will not run as a master server when needed.

Step 13. Test connection

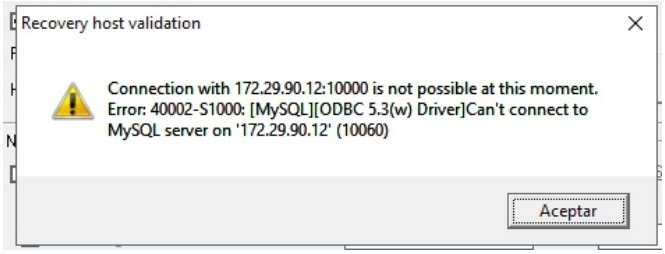
After parameters are filled in, click on the button "**Test Connection**" to test the connectivity between the servers. **Do this step from both the slave and the master server**.

WARNING: Remember to do it for both servers because we need to guarantee that the servers can connect to each other and that they have the appropriate permissions.

If you forget to fill any field or you don't do it rightly, a message box will appear showing you the reason of the error when testing the connection:

- *"Host IP is mandatory and must have a valid IP address."*
- *"Host port is mandatory and must have a value between 0 and 65535."*
- *"Host name is mandatory."*

When field are good filled, you will be asked if you want to proceed to the test. Press "*Yes*" to proceed. If any error occurs, it will appear an error message like the one that follows:



If the connection can be stablished, a success message will appear: "*Connection with HostI:Port successful!*"

Step 14. Apply Recovery

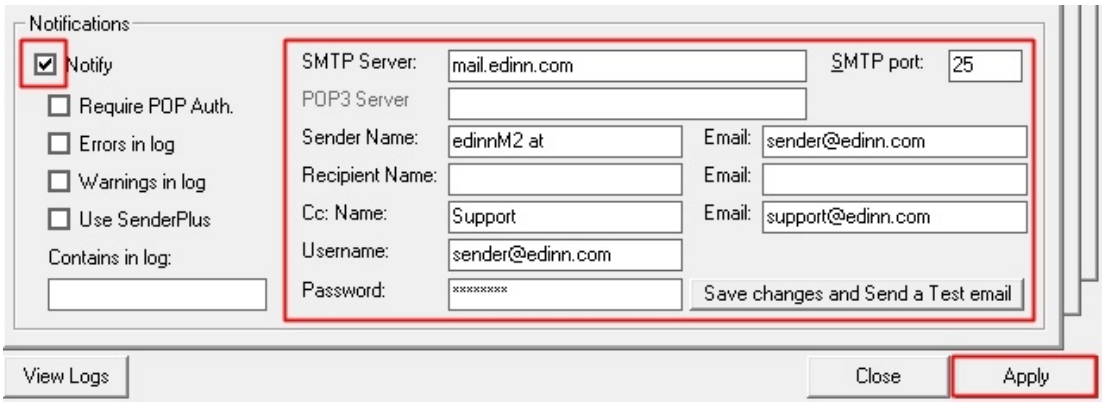
1. In the **Console**, **Supervisor** tab, **Recovery** section, select the **Slave** role.
2. Press the "**Apply**" button of the Recovery section.

Step 15. Failure Reporting Configuration

To enables server to inform by e-mail about the failures and other issues, you must enable **Notify** in Notifications frame and fill the fields as you can see in the image bellow.

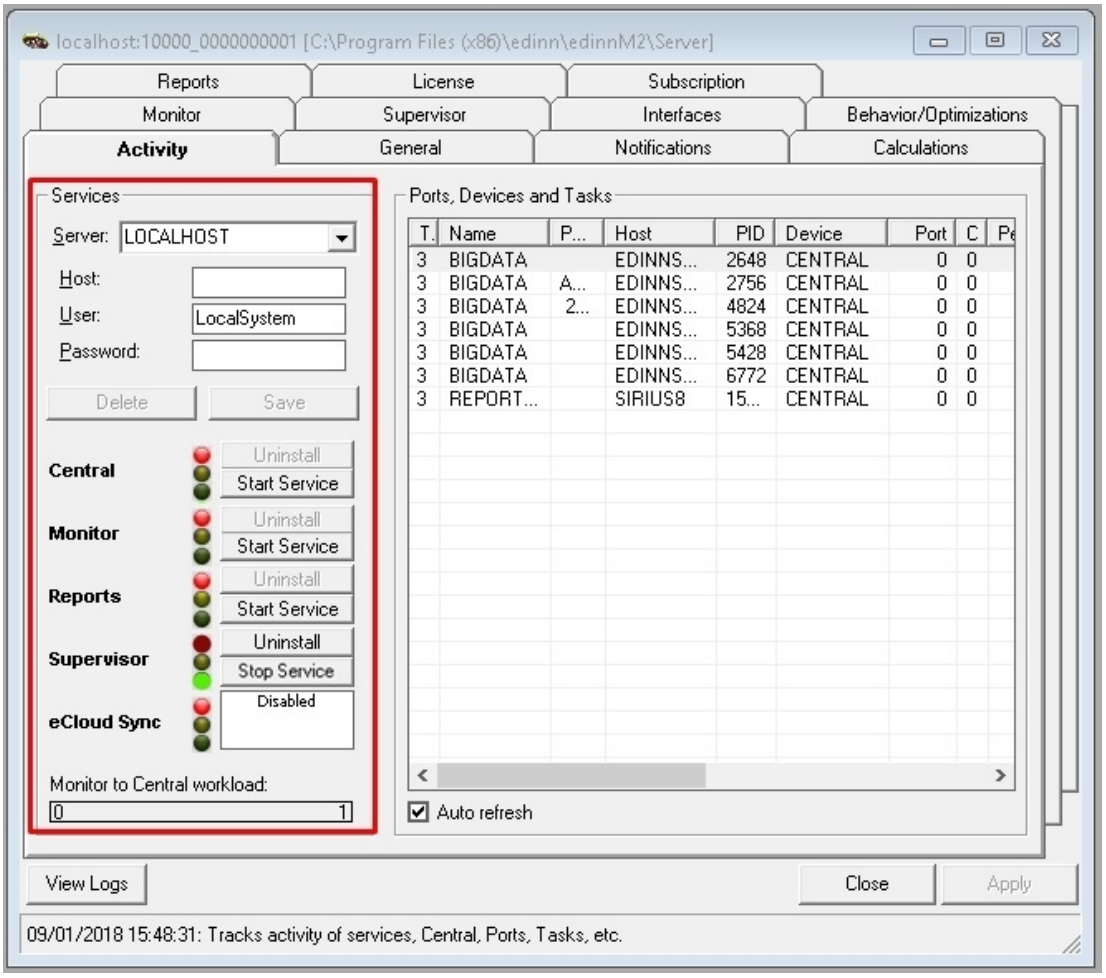
Remember to click on **"Apply"** button to save changes.

> NOTE: for more information of the Notifications configuration, please see: Notification.



Step 16. Start Supervisor service in the Slave server

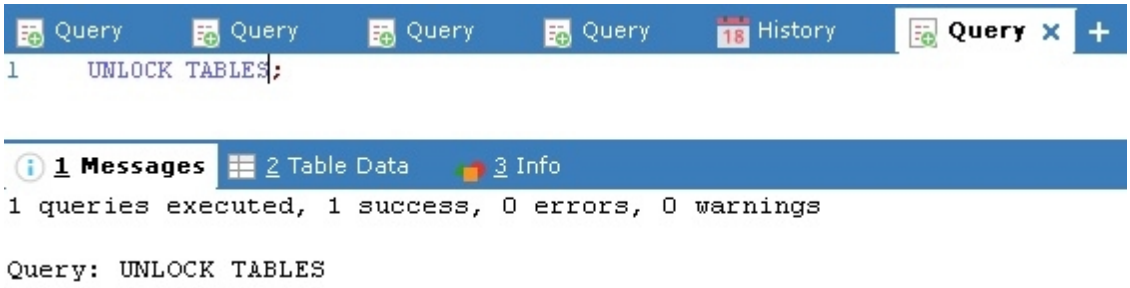
1. Go to path **C:\Program Files (x86)\edinn\edinnM2\Server** and execute, with administrator permissions, the program **edinnM2_Console.exe**.
2. In the **Activity** tab press **"Start service" ONLY for the Supervisor service.**



4. Final procedure

Step 17. Unlock database tables

Run **SQLyog** as administrator in the **master server** and execute the following query: "**UNLOCK TABLES;**"

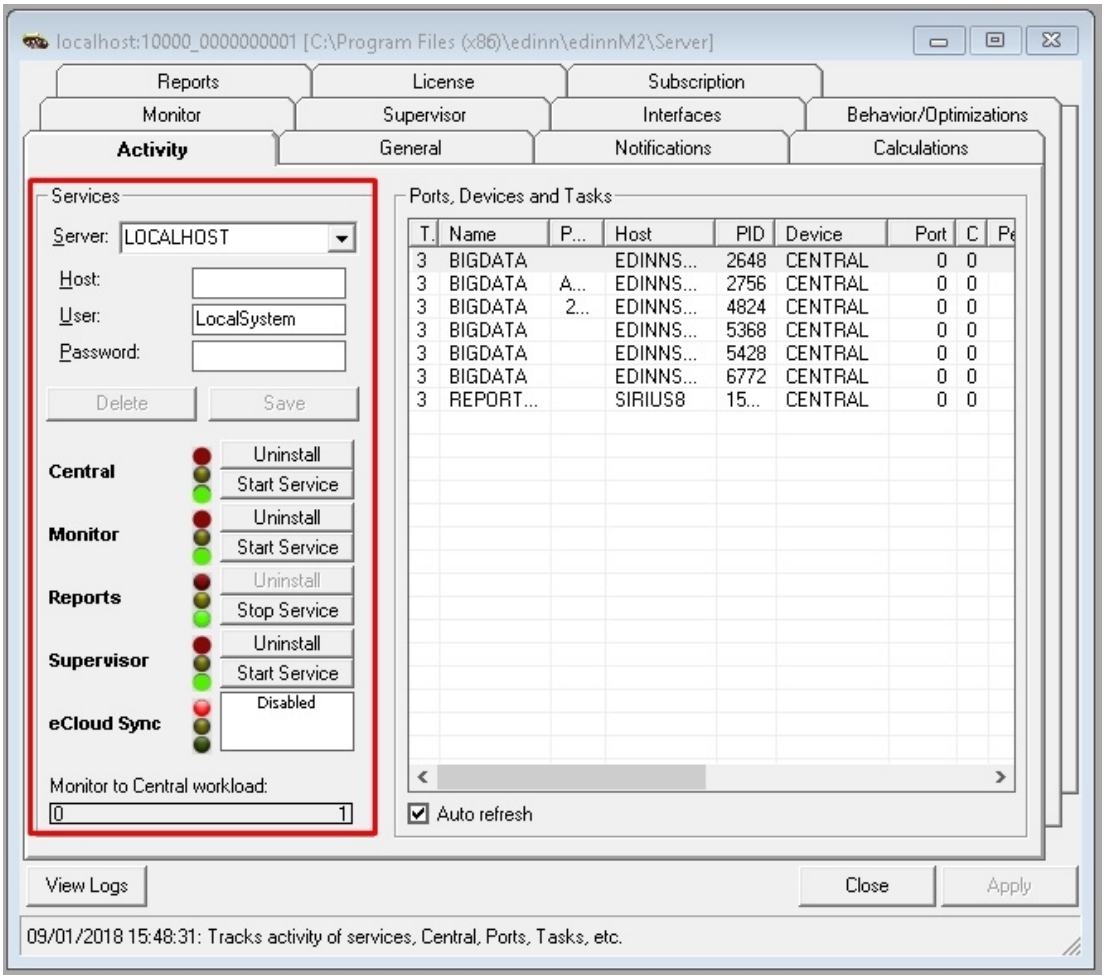


Step 18. Start services in the Master server

1. Go to path **C:\Program Files (x86)\edinn\edinnM2\Server** and execute, with

administrator permissions, the program **edinnM2_Console.exe**.

2. In the **Activity** tab press **"Start service"** in each of the services, going from top to bottom (starting with the **Central**)



Possible errors

There are some erros that you could find in the operative system event viewer:

- **Error: 1593-Fatal error: The slave I/O thr at**

  Solution: Possibly the server-id parameter was not specified in master or slave "my.ini" files. Please refer to step 1.

- **Error (or in general, to restart the whole process): Slave SQL for channel '': Error 'Duplicate entry**

  Solution: Possibly there was an error in procedure from the backup to the restore of the database from master to slave. The solution is, or in general, if you need to restart the whole process:

1. Stop edinn supervisor service in slave server.
2. Stop all edinn services in master server.
3. Stop the edinnM2_MySQL service in master server.
4. Delete, from the master server, the binary log files present in the data directory of the database server MySQL. Usually: log-bin.index and log.bin.000000X
5. Start the edinnM2_MySQL service in master server.
6. Redo all procedure, from step 2. Master Configuration.

- **Error: Central service does not start on slave server**

  Solution: Possibly the license was not saved in this step, and now actual license is not valid. License string is possiby lost, therefore please request to edinn a new license for this server.

Restoring master server

WARNING: this procedure is under revision and therefore is here just for information purposes. Use at your own risk.

This procedure applies **only after you have performed the Recovery procedure** and **you are using the slave server as your principal master server**.

You can proceed with **this procedure only when you have confirmed that your master server is working fine again and is connected to the network**.

Step 1. Change role of your old master server (do this step as soon as possible)

1. In the master server, run **edinnM2_Console.exe** as **administrator**.
2. Open the **Supervisor** tab and change the **rol** from master to **slave**. This should have automatically stopped all services, which can be verified in the **Activity** tab.

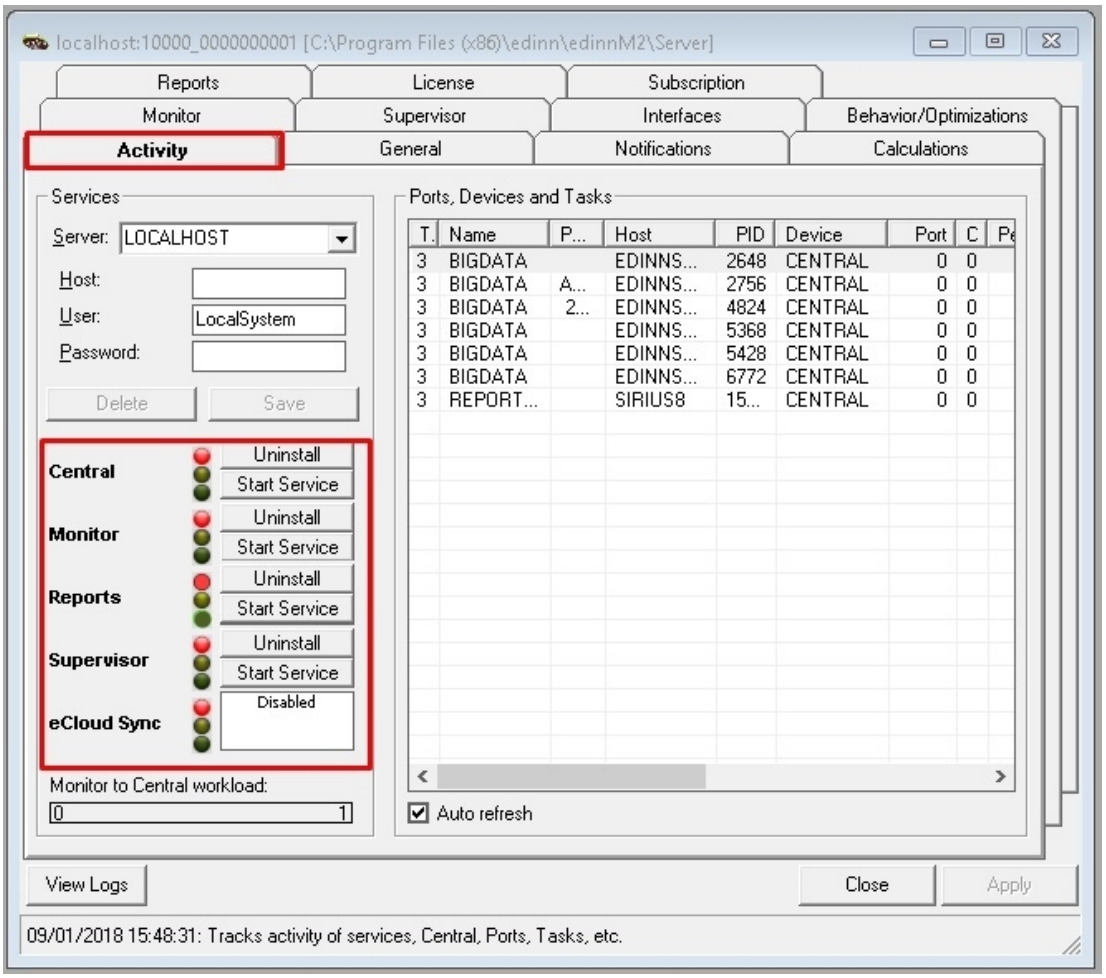
This will block access from operation terminals and applications. This is because, now your slave server is working as master, and your master server is switched on again and probably already connected to the data network. In this situation, maybe some user terminals or applications would connect to the recently restored master server. This would cause that some users are connected and operating with the slave server, which now is acting as the master server, and some other users with the restored master server, and therefore, that would result in data loss for those operating with the recently restored master server.

Step 2. Stop all edinn M2 services in slave server (which is acting now as master

server)

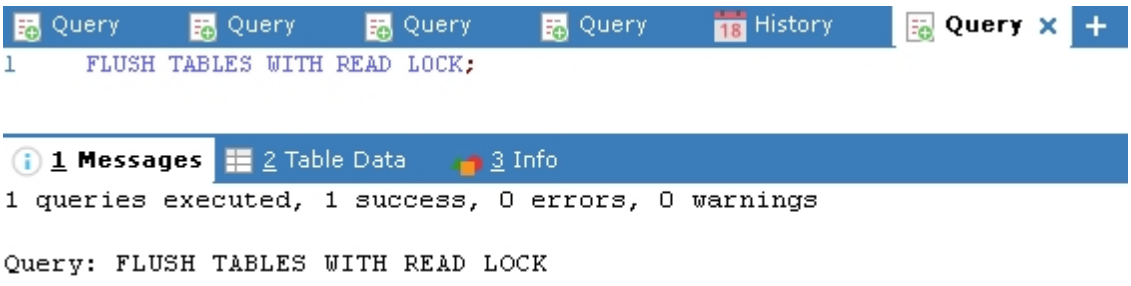
1. Go to path **C:\Program Files (x86)\edinn\edinnM2\Server** and execute, with administrator permissions, the program **edinnM2_Console.exe**.

2. In the **Activity** tab press **"Stop service"** in each of the services, going from bottom to top (starting with the **Supervisor**)



Step 3. Block access to the database of the slave server (which is acting now as master server)

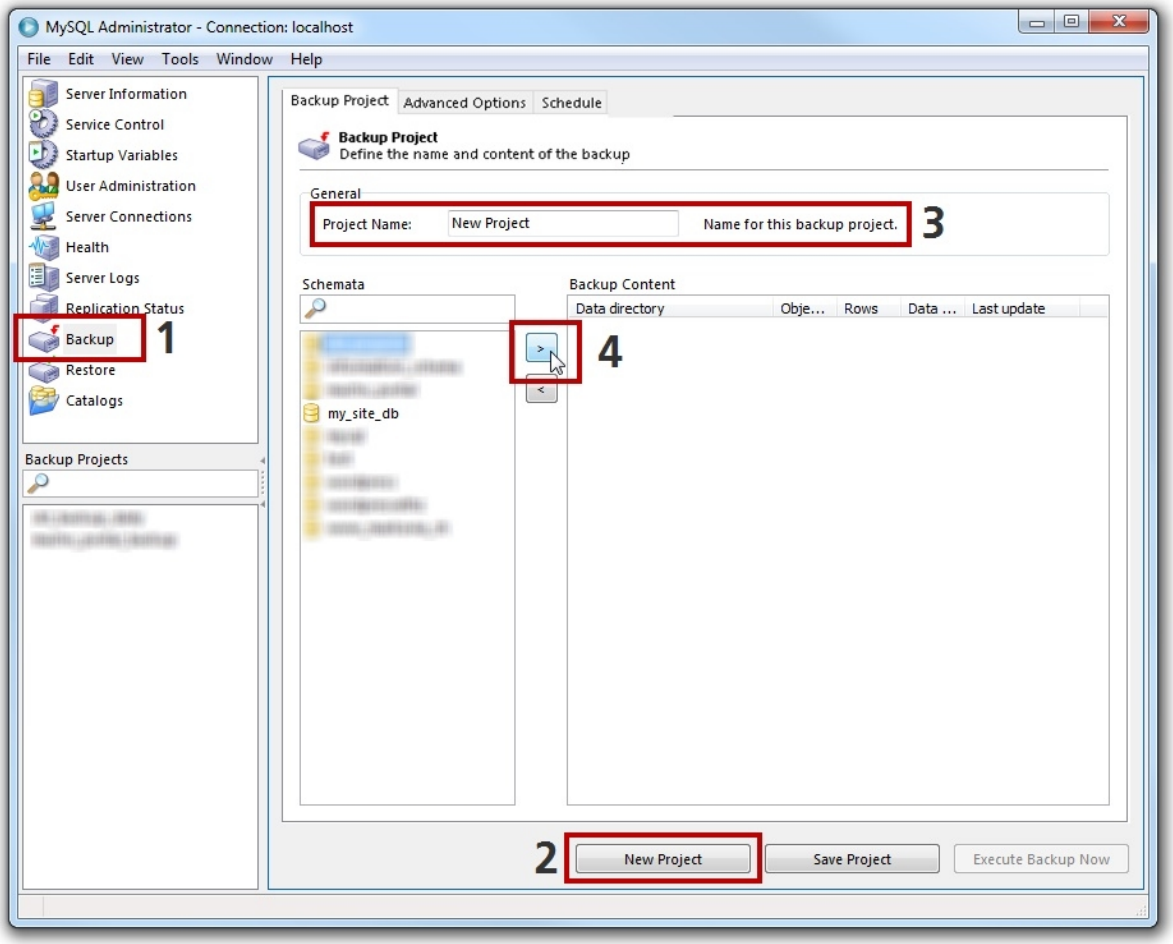
On the slave server (which is acting now as master server), run **SQLyog** as **administrator** and execute the following query: "**FLUSH TABLES WITH READ LOCK;**"



Step 4. Backup the database of the slave server (which is acting now as master server)

As users and applications where interacting with the slave server, now we need to restore all this data to the master server. To do this, perform a database backup of the slave server (which is acting now as master):

1. Open **MySQL Administrator** as **administrator**.
2. Enter the backup tab, click on **"New Project"**, choose a Project Name and select the 2 databases from which you are going to perform the backup: the edinnm2 **and** the edinnm2_CompanyId database. Then click the > icon and **Save Project**.
3. Click "Execute Backup now" and wait for the backup to terminate. Copy the generated backup file and paste it in the Server who is going to act as a **Slave**. **Remember to not start the edinn services on the master server until indicated later**.



Step 5. Copy, and retain for later, the license of the master server

1. Go to path **C:\Program Files (x86)\edinn\edinnM2\Server** and execute, with administrator permissions, the program **edinnM2_Console.exe**.

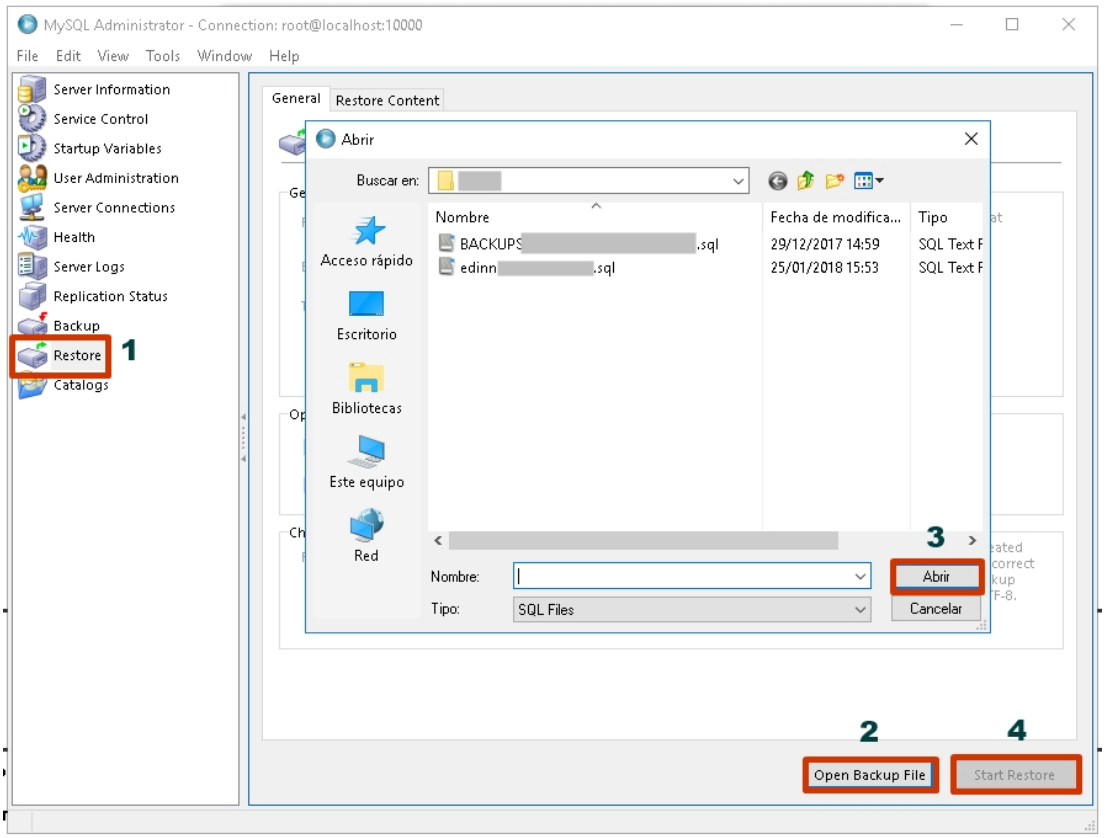
2. In the **License** tab, copy the license to a notepad or file, to be used later.

WARNING: do not forget to copy the License field content to a notepad of file. **If you do not provide this valid license later for this master server, the master server will not work**.

Step 6. Restore database in master server

Perform a database restore of the Backup that you generated in the previous step:

1. From the Windows search icon, type **"MySQL Administrator"** and right-click the app returned and select **"Run as administrator"**.
2. Enter the Restore tab, click on "**Open Backup File**", choose the backup file generated in the previous step, and then click "**Open**".
3. Click on "**Start Restore**"

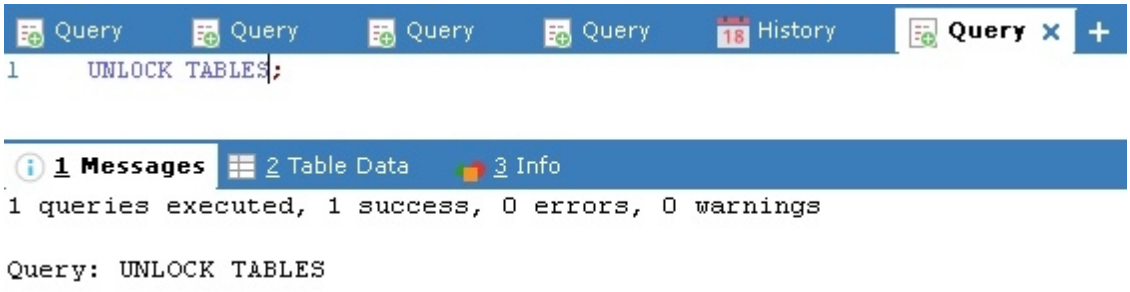


4. Once restored, use a database editor (for example the SQLYog application recommended before), and edit the table edinnm2.comp and **change, in all fields, the IP address of the master server by the IP address of this slave server**.
5. After performing the restore of the database, the edinn license of this server has been replaced by the license of the master server. Please **paste the license from the notepad or file where you copied it in the previous step**, or request for a new license to edinn

WARNING: if you do not provide a valid license for this master server, it will not work.

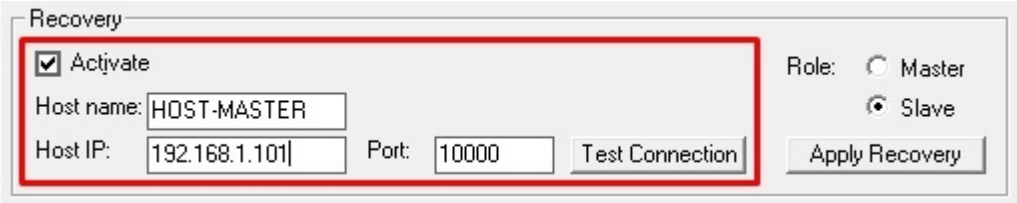
Step 7. Unlock database tables

In the **slave server**, run **SQLyog** as administrator and execute the following query: "**UNLOCK TABLES;**"



Step 8. Supervisor Configuration on the slave server

If you have the edinnM2_Console.exe running, **close it**. Then **run again edinnM2_Console.exe** as **administrator** and open the **Supervisor** tab. Confirm the following parameters:



Where:

- **Host name:** is the remote host name. You can find it in the properties window of the master server computer.
- **Host IP:** find it by writing the command "ipconfig" in cmd.exe in the master server computer.

Write here the information from the master server:

1. **Switch on the Activate checkbox**: to enable the rest of the fields.
2. **Host name:** is the name of the master server in the network, the one you are trying to connect to.
3. **Host IP:** is the IP of the master server in the network.
4. **Port:** write '10000'.

Step 9. Test connection with master server

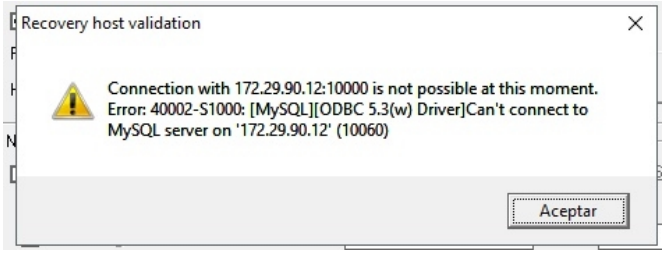
After parameters are filled in, click on the button "**Test Connection**" to test the connectivity between the servers. **Do this step from both the slave and the master server.**

> **WARNING:** Remember to do it for both servers because we need to guarantee that the servers can connect to each other and that they have the appropriate permissions.

If you forget to fill any field or you don't do it rightly, a message box will appear showing you the reason of the error when testing the connection:

- *"Host IP is mandatory and must have a valid IP address."*
- *"Host port is mandatory and must have a value between 0 and 65535."*
- *"Host name is mandatory."*

When field are good filled, you will be asked if you want to proceed to the test. Press "*Yes*" to proceed. If any error occurs, it will appear an error message like the one that follows:



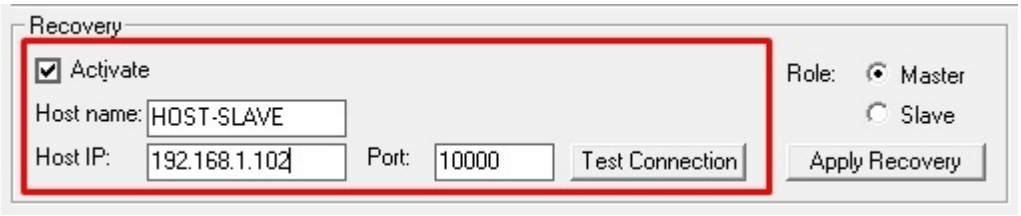
If the connection can be stablished, a success message will appear: "*Connection with HostI:Port successful!*"

Step 10. Apply Recovery on slave server

1. In the **Console**, **Supervisor** tab, **Recovery** section, select the **Slave** role.
2. Press the "**Apply**" button of the Recovery section.

Step 11. Supervisor Configuration on the master server

Open the **Supervisor** tab. Confirm the following parameters:



Where:

- **Host name:** is the remote host name. You can find it in the properties window of the slave server computer.

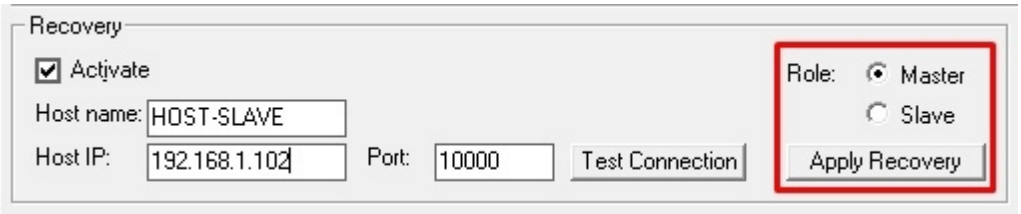
- **Host IP:** find it by writting the command "ipconfig" in cmd.exe in the slave computer.

Write here the information from the slave server:

1. **Switch on the Activate checkbox**: to enable the rest of the fields.
2. **Host name:** is the name of the slave server in the network, the one you are trying to connect to.
3. **Host IP:** is the IP of the slave server in the network.
4. **Port:** write '10000'.

Step 12. Apply Recovery on the master server

1. Select the **Master** role.
2. Press the "**Apply**" button of the Recovery section.

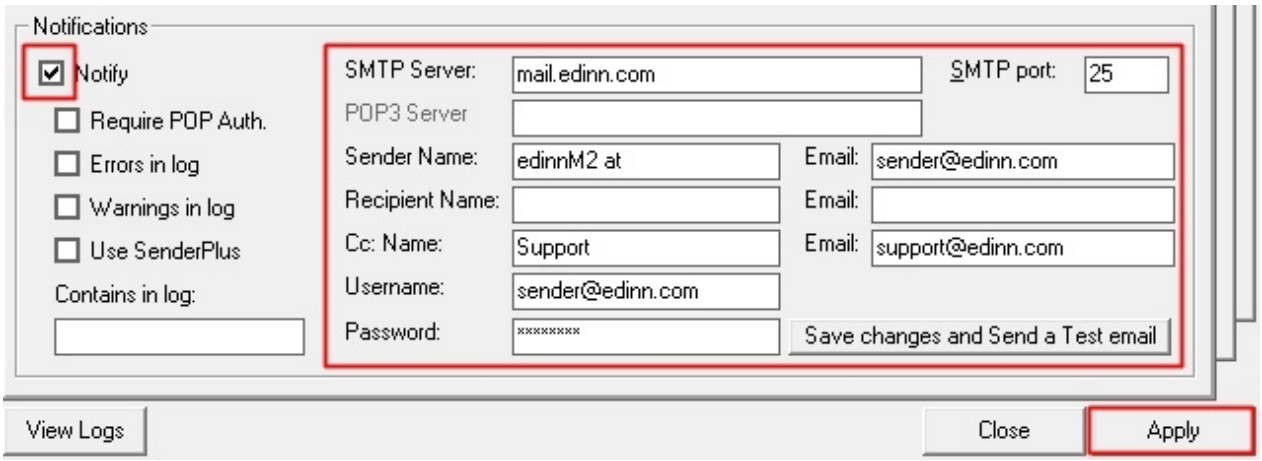


Step 13. Failure Reporting Configuration

To enable server to inform by e-mail about the failures and other issues, you must enable **Notifications** in the Notifications tab and fill in the fields as you can see in the image bellow.

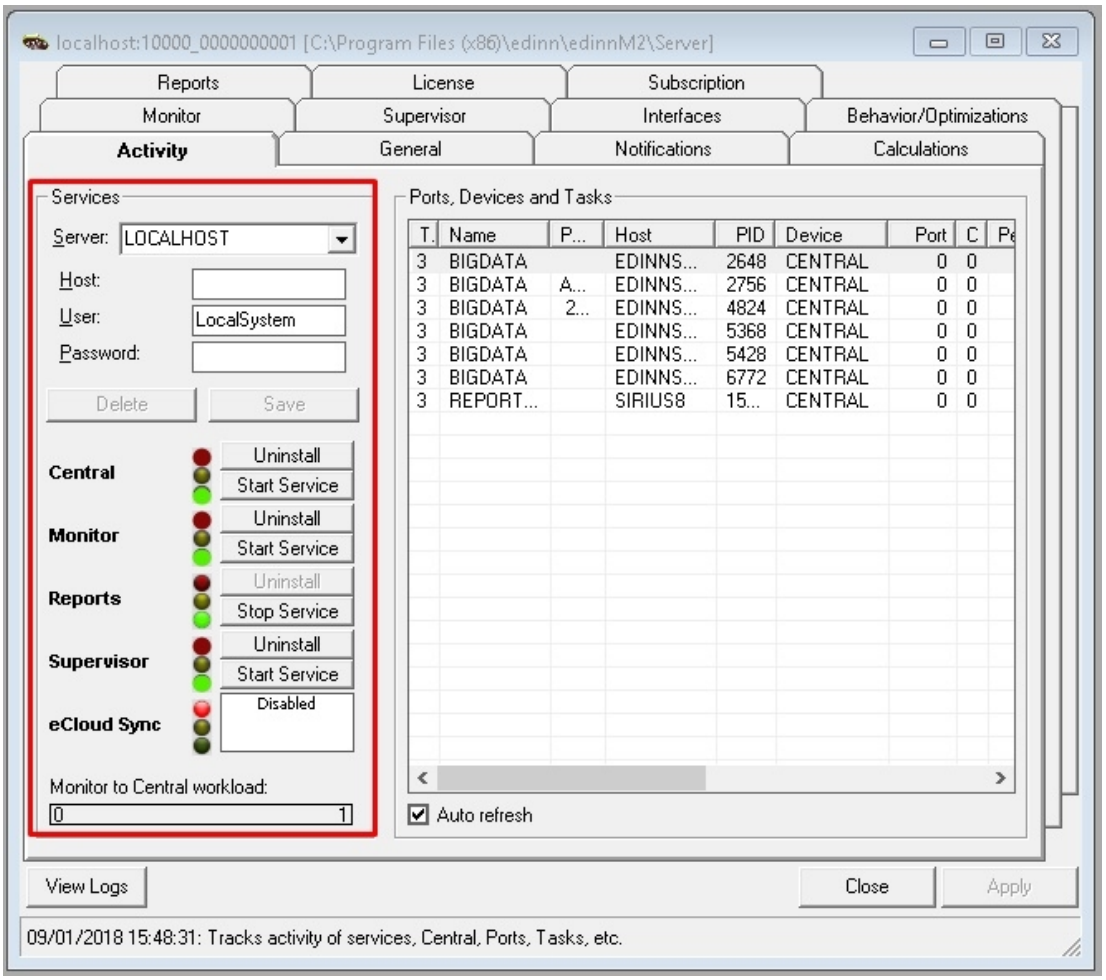
Remember to click on [**Apply**] button to save changes.

> **NOTE:** for more information of the Notifications configuration, please see: Notification.



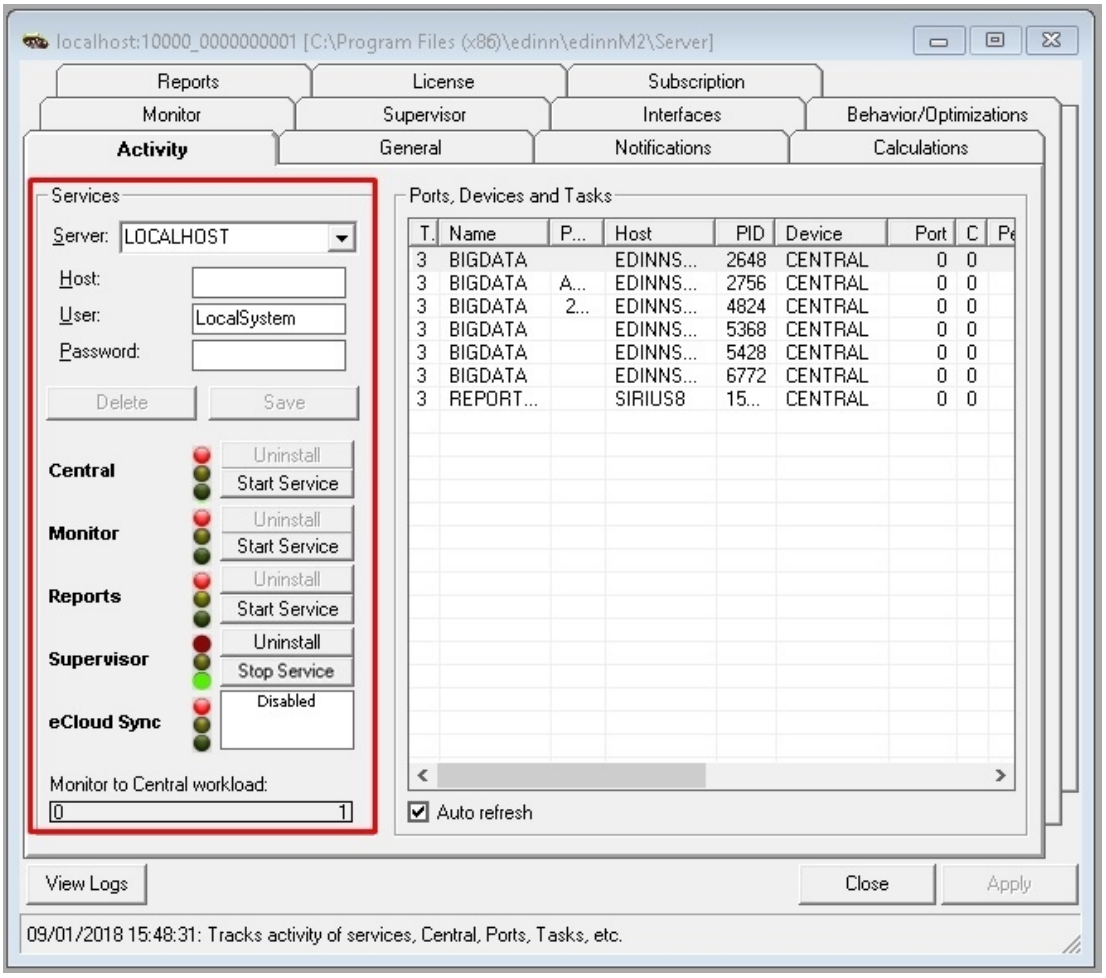
Step 14. Start services in the master server

1. Go to path **C:\Program Files (x86)\edinn\edinnM2\Server** and execute, with administrator permissions, the program **edinnM2_Console.exe**.
2. In the **Activity** tab press **"Start service"** in each of the services, going from top to bottom (starting with the **Central**)



Step 15. Start Supervisor service in the Slave server

1. Go to path **C:\Program Files (x86)\edinn\edinnM2\Server** and execute, with administrator permissions, the program **edinnM2_Console.exe**.
2. In the **Activity** tab press **"Start service" ONLY for the Supervisor service.**



Step 16. Inform users to reconnect to the master server on their operation terminals and applications

Inform all users on how to connect to the master server and provide them with the **IP address and port** of the **master server**.

END OF PROCEDURE

END OF PROCEDURE

Configure URLs

URL fields exist for the **configuration of URLs** in various points of the configuration like process, status, results, etc.

These fields allow the user to open a document or application when clicking on the button that appears after having clicked on a certain element in the data grids that are in almost every window of the application.

In order to configure the URLs it is necessary to follow this syntax:

COMPLETE_PATH_TO_DOCUMENT_OR_APPLICATION1 = CLICABLE_SHOW_TEXT1,
COMPLETE_PATH_TO_DOCUMENT_OR_APPLICATION2 = CLICABLE_SHOW_TEXT2,
COMPLETE_PATH_TO_DOCUMENT_OR_APPLICATIONn = CLICABLE_SHOW_TEXTn

Where:

- *COMPLETE_PATH_TO_DOCUMENT_OR_APPLICATION* is the complete text so that the computer which executes the terminal can open the document or device, for example:
  Z:\MANUALS\MACHINE_MAINTENANCE_MANUAL.PDF
- CLICABLE_SHOW_TEXT is the text that will appear and where the user must click in order to open the URL. Various separated URLs can be indicated by comma.

Example: **Z:\MANUALS\MACHINE_MAINTENANCE_MANUAL.PDF = Maintenance Manual**

How to verify URLs?

To verify if the URLs configured in a specific device work, we recommend to follow these steps:

1. Go to the device which you want to verify. Continue the following steps physically from that device.
2. With edinn administrator permission, open the configuration of the URL (processes, statuses, results, etc.) which is not working. Copy to the windows clipboard the part of the URL which corresponds to the path to the file, application, web page, files folder, etc., that you wish to open.
3. Now we will try to execute, from the device, what you copied in the previous step: you can use the key combination **Windows + R** and paste, **or** use the **Windows Tasks Administrator** (its menú option appears with the combined keys **CTRL+ALT+SUPR**) and select File and Execute New Task and then paste what you copied in the previous step. Click Enter to execute.
4. If what you wanted to open, does not show up, you should find out why because it is not a problem of the edinn platform. Otherwise, please report it to our support indicating that you have already followed this procedure.

Password recovery

To recover your password:

1. Open the reports and, from the login dialog, click on the link Forgotten Password.
2. Type your company and user id.
3. Click on **Send** and the password will be sent to the email 1 that was configured for your

user.

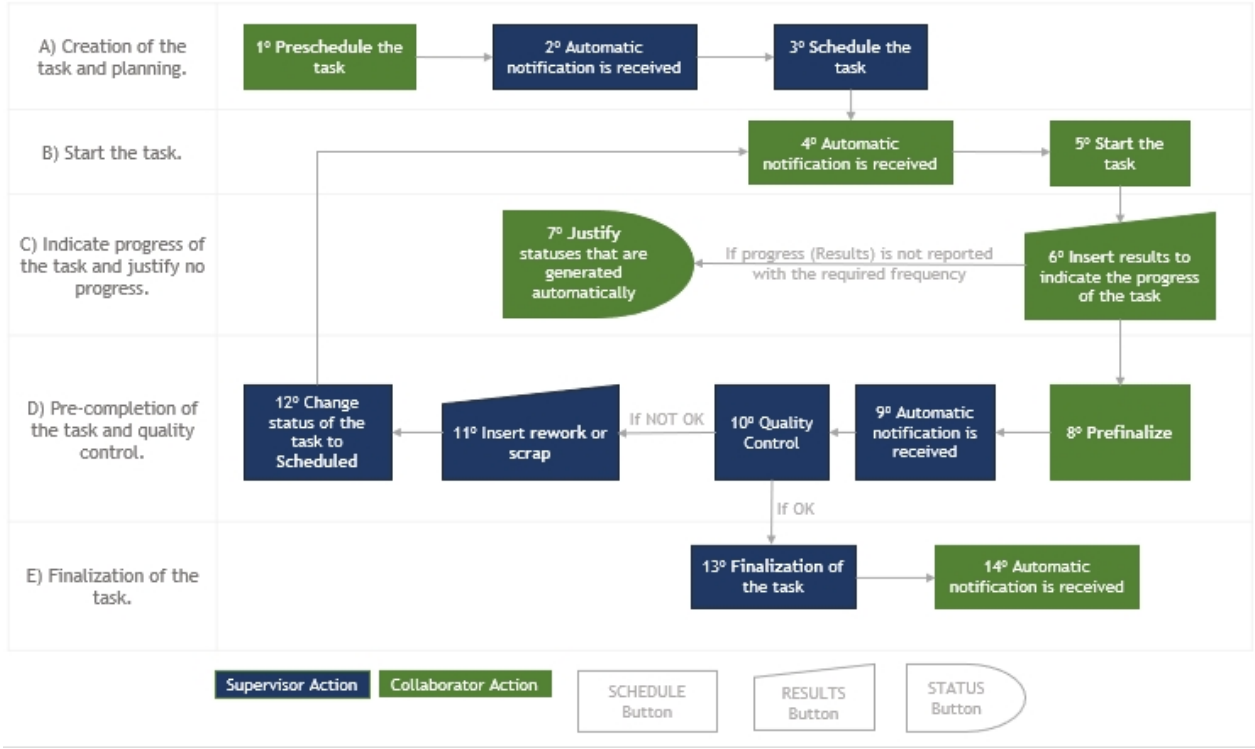
In a moment you will receive, in the indicated email, instructions to proceed.

Work Management

To obtain the benefits of the Work Management module:

- **Create a process for each user** and configure the required daily operation time. We recommend you to **add a photo of the person** for each process.
- **Configure the time statuses** in order to generate the work calendar. Please take into account that you should configure correctly the classes of the statuses:
  - Labour periods will be the real time minus the times of statuses marked with classes weekends or holidays. However, the scheduler will consider, in the future, as labour time the real time minus the times of statuses marked with classes weekends, holidays, external dependencies and breaks.
  - The system will consider as worked the total time of the statuses marked with the productive class. This is useful, for example, to consider the brunch inside of the labour day, by marking this status as class productive.
  - The system reads weekends and holidays from the calendar (planned statuses) and not from the statuses window, therefore you should not allow users to insert or justify weekends and holidays. On the other side, the system reads vacations from the statuses window and therefore you should allow users to insert or justify their vacations.

Now, according to this flow diagram:



- Train and make the working team:
  - **Create the tasks from edit schedule (step 1 in diagram).**
  - **Ask for them to be scheduled, from the change status button, so that the supervisors receive an automatic notification (step 2)** to check the times, finish dates, and characteristics, after which the supervisors can schedule **(step 3)**, with the same change status button, the task so the worker will receive a notification **(step 4)** and may begin working on them.
  - **Start (step 5) the tasks and insert the advancement of results (step 6) for them.**
  - **Justify the not productive statuses (step 7)** when the system requires it due to not having input a status, or failing to input sufficient results.
  - **Prefinalize the tasks (step 8)** when they are finished, which will generate and send an **automatic notification (step 9)** to the supervisors so they may check the quality **(step 10)** of the results of the given task.
  - **Supervisors should report low quality results (step 11),** when they have been detected in the tests before finalizing a task. An objective and easy way of doing this is by giving the maximum possible detail based on subtasks, gathering these subtasks in a document which is linked to the edinn task or order as an URL, and in the quality evaluation performed by the supervisor, he/she will insert the % of no quality based on the subtasks of the task or order that were not completed successfully.

        - **Supervisors must then finalize the tasks (step 13)** after reviewing the quality of results, which will generate an automatic notification to the worker **(step 14)**; or, return the tasks to the **scheduled (step 12)** state so their colleagues may continue working on them until reaching the desired results.

    - Visualize or receive alerts:
        - **Projects or tasks that are going to be delayed or have already been delayed.**
        - Personnel which should:
            - **Report results.**
            - **Justify their time spent working.**
            - **Perform periodic tasks or procedures.**
            - Other.
    - Obtain **reports:**
        - Indicators of personnel performance**, including an adapted OEE:**
            - **Availability:** percentage of completion of their designated work hours.
            - **Velocity:** percentage of completion of tasks within their designated deadline.
            - **Quality:** percentage of high quality results compared to all given results.
        - Projects and pending tasks, to whom they are assigned, anticipated deadline and completion dates, the state of each one, percentage of advancement, etc.
        - Total time dedicated per project and task:
            - Clic on the records to see the detail in time in Status\Status.
            - To know what was done in detail, clic on the records to see the detail in Status\Status and then switch to the Results\Results report, type the "To order" field with the same task code that is present in the "From order", mark show fields and then view report.

> Note: in the previous reports, to have the detail of what was done in each task, you have 2 options:
>
> 1. Each task performed should have a separated task into the system. Additionally, to be able to know the total hours of the project, all tasks belonging to the same project should belong

to the same area.

2. If you use a generic task to include different real tasks, this task should include a result with a field of description mandatory if empty.

- **Total time dedicated per project, task, and state**.
- **Summary of monthly hours** worked.
- **Results** that have been generated for a given task.

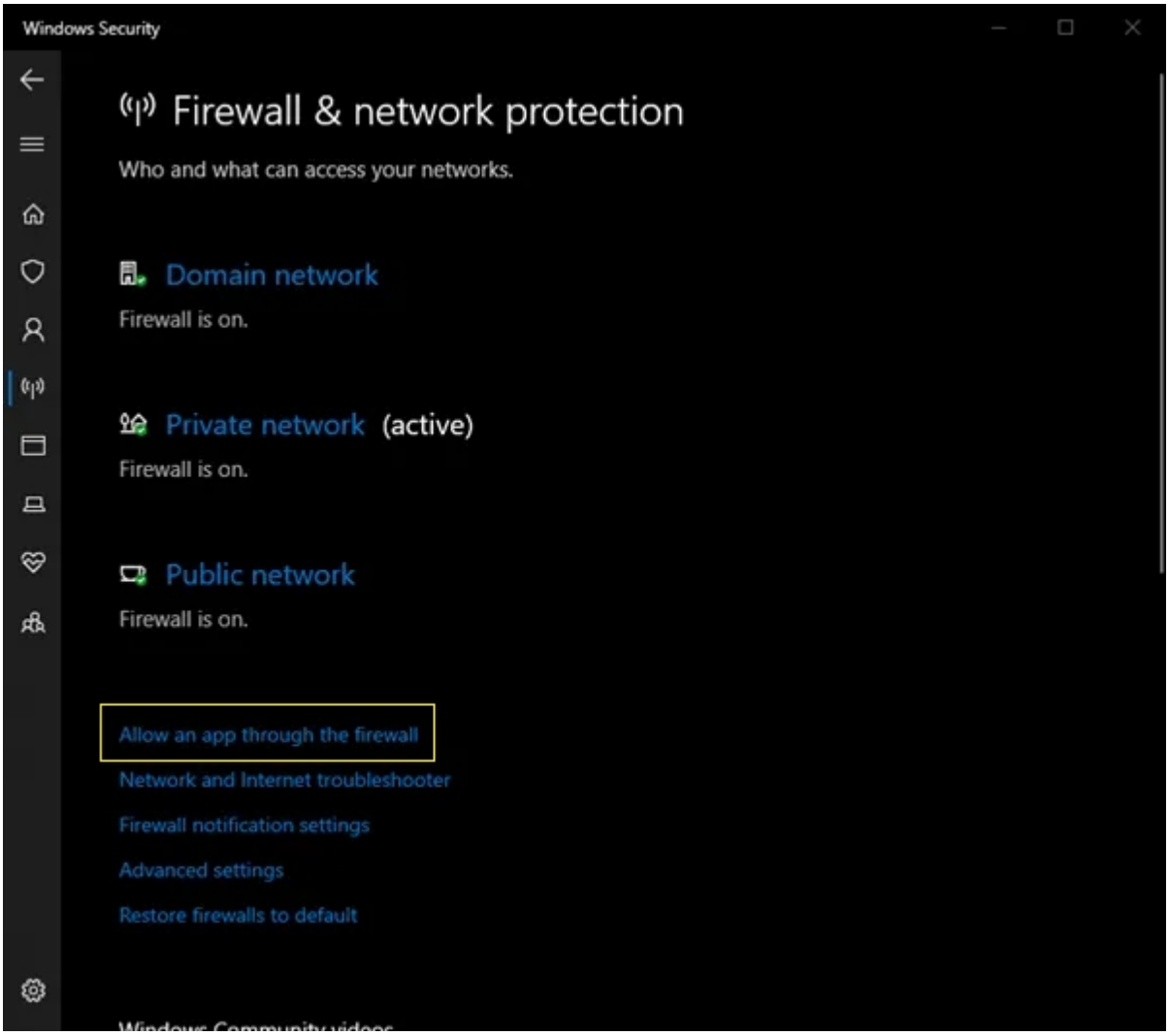
Manual update of Terminal

If this web page showed up automatically, probably your win32 terminal cannot update automatically.

The operation terminal for Windows (known as the win32 version), when correctly configured here, should update automatically from your server when there is a new version available. You can also force it to update automatically.

If your terminal does not update automatically, the most frequent solutions are:

1. Wait, if the updater **edinn M2 Updater** (process **M2Updater.exe**) is working, because some times the updater is working but its window is behind of other windows. Check, with **ALT+TAB** or with the **task manager,** if the updater **edinn M2 Updater** (process **M2Updater.exe**) is working and advancing. If, wether is not working, or is not advancing, keep reading this document.
2. If the updater **edinn M2 Updater** (process **M2Updater.exe**) is working, terminate it with the **task manager**.
3. Make sure that the application **edinn M2 Updater** has permissions in the firewall of your PC. To do this, use the **Windows key** and type "**Firewall & network protection**". Then clic on "**Allow an app through the firewall**", as shown in the following image:



In the **applications list** that will be shown, select **edinn ® M2 Updater** and make sure that this app is allowed to access **Private** and **Public** networks. For this, you may need to click on the button above "**Change settings**" and, after activating **Private** and **Public** for the indicated application, accept the changes by clicking **Ok** below.

4. Execute the **edinn M2** app with **Administrator rights**. This is typically done by typing "**edinn**" in the Windows Start Menu, and then, over it, with the right button of the mouse, a menu should pop up and then you should choose "**Run as administrator**".
5. **After doing all the previous steps**, retry the automatic update as indicated here.
6. If the previous steps did not update your terminal, continue reading this document.

If after the previous steps the automatic update failed, you can proceed with any of these options:

1. Perform the update manually, as indicated later in this document.
2. Reinstall the terminal application, as indicated in install the terminal.

Let us see here **Option 1: Perform the update manually**

WARNING: This procedure will only work in systems where there is a previous intallation of the Windows edinn M2 terminal (win32).

1. Download the zipped file edinnM2_client_offline_update_win32.zip

WARNING: You must download the following file from your edinn server, therefore, please get sure, by checking the URL of this help document (for example it should be similar to **http://[THE IP ADDRESS OF YOUR EDINN SERVER]/edinnM2/client/edinnM2_client_offline_update_win32.zip**), that you are accessing this documento from your edinn server. Otherwise, you might download an incompatible version which could not work or cause an issue.

Download this file: edinnM2_client_offline_update_win32.zip

2. Unzip it in your hard disk

As this is a standard ZIP zipped file, you should be able to easily uncompress it by **right clicking** on it with your **mouse** and selecting **Extract**.

3. Execute file _Update.bat with Administrator permissions

WARNING: DO NOT execute the file from the zipped file. Be sure that previous step was done and that **you are inside of a regular windows folder** and not inside of the zipped file opened with double click.

Once your files have been uncompressed, enter into the extracted folder (normally **edinnM2_client_offline_update_win32**), select the file **_Update.bat** and **right click** on it with your **mouse** and select **Execute as Administrator**. The procedure will require you to confirm if your terminal is installed in the standard path. **If your terminal is installed in the default path**, you should respond afirmatively by typing a **Y** and then the **enter key** to start the update. **If your terminal is not installed in the default path**, you should type **N**, **enter key** and then type the full path.

Please wait until the procedure finishes and then you should be able to execute successfully your updated software.

Still having errors after updating? In some cases, this has proven to solve them:

- Uninstall all versions of the edinn M2 terminal from the PC.
- Delete the folder of the edinn M2 terminal, normally present here: "C:\Program files (x86)\edinn"
- Delete the folder: "%userprofile%\appdata\edinn"
- Reinstall the edinn M2 terminal.

Need help? 

Project

This section explains how to connect your eBOXes and eHUB and how to inform edinn to continue with the installation and configuration.

This section also includes all the information to follow for the do-it-yourself (DIY) option.

This section refers frequently to other documents and files which are available inside the edinn Academy.

> **NOTE:** If you choose full service with an eHUB, you do not have to perform the steps marked in this guide with the icon .

Introduction

The edinn platform can be installed and configured from full service to do-it-yourself (DIY).

Option A) Full service installation using an eHUB device, these are the steps you need to perform:

1. Prepare, **those steps which are NOT marked with this icon** , of the monitoring section of this guide.
2. Inform, as indicated in the eHUB Quick Guide, to your edinn installation expert.

Option B) Do it yourself (DIY), these are the steps you need to perform:

1. Prepare monitoring
2. Create & Configure
3. If needed, support by our installation experts can be requested.

These tasks can be performed in parallel, although the 1st ("Prepare monitoring") must be completed before certain steps of the 2nd can be completed.

The edinn platform can be installed in your own local or cloud server. For users who have installer access in the edinn Academy, there is a guide named "edinn Server Installation and Maintenance Guide" which includes the necessary files and documentation.

NOTES:

- If you choose full service with an eHUB, you do not have to perform the steps of the documents with the icon .
- During the installation procedure a change of installation plan can be requested, from full service to do-it-yourself and vice versa.

Need help?

Previous preparation

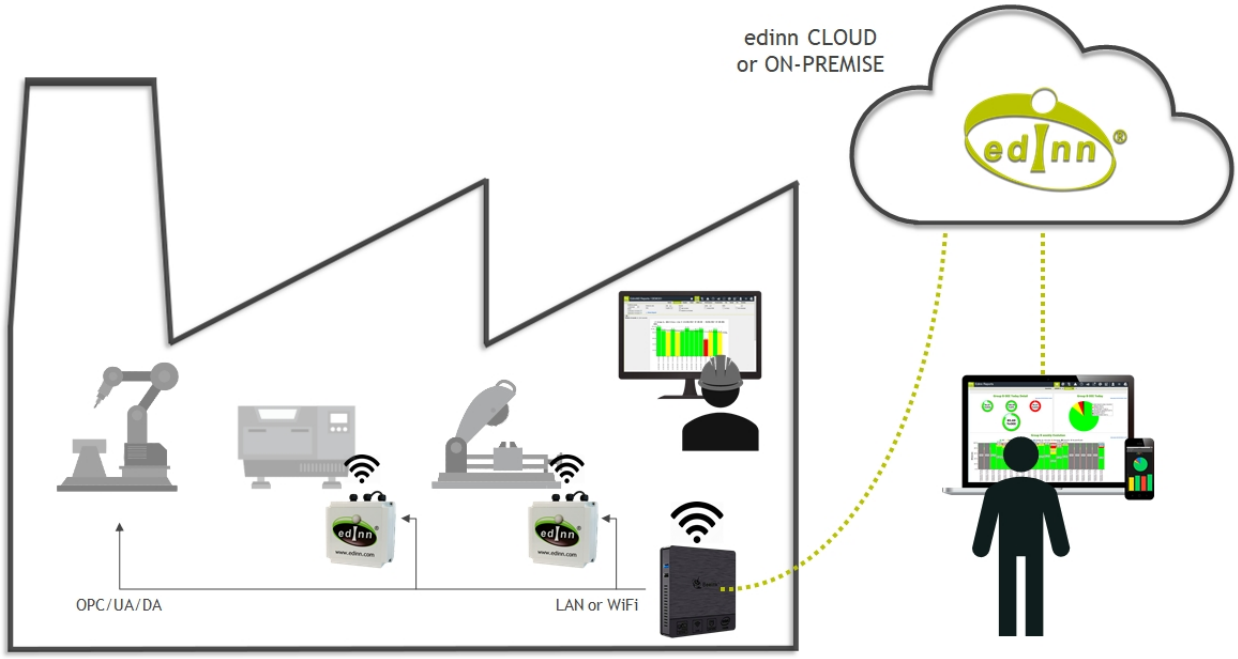
This section explains the previous tasks that you, as a user, should complete before proceeding with the installation of the platform, whether it is installed by an authorized installer or installed by yourself.

To continue, please check:

- Summary of checklist.
- Checklist.
- Factors for success.

If you have a question or doubt at any moment about what is indicated in this section, please contact us.

The following image shows a diagram with a typical installation.



Summary of checklist

This is a summary of the checklist that should be completed before an edinn installer starts the project:

- Have you decided **how many and which machines or production lines** do you need to monitor?
- Have you nominated (recommended as many as possible) professionals inside your organization who will be the **leaders of edinn**? And have you already requested and obtained their **users for support** in the EDINN support here? http://ecloud.edinn.com/support/
- Do you have an **Ethernet Data Network** to the machines you want to monitor?
- Have you requested, to the computer network responsible, the **fixed IP addresses** for the eBOXes and/or PLCs and the server?
- Is there **power supply (220V)** for the eBOXes and the operation terminals, near the machine.
- Do you have **internet access** with the minimum requirements?
- Do you have a **server and operation terminals** with the minimum requirements?
- **Monitoring**: do you plan to read data from the machinery through OPC? Please check in this document the Prepare Monitoring section.

- Have you created and configured your company here? http://ecloud.edinn.com/edinnM2/wiz/register.php?lang=en

> **Note:** Remember to check the critical factors for the success of the installation.

Checklist

- Have you decided **how many and which machines or production lines** do you need to monitor?
- Do you have your **Payment Code** or **Transaction Id**? You will need it to create your company in edinn.
- If you need local service:
  - Have you signed and stamped the **Installation Purchase Order** and/or the **Service Conditions** and accomplished the indicated conditions?
  - **Access and/or work documentation:** if the edinn system installer is external to your company, and you require any type of documentation (Labor Risks Prevention or similar) and/or authorization to enter and/or work in your facilities: have you sent to the edinn installer this documentation to be able to enter and/or work in your facilities?
  - Has the edinn installer returned to you this documentation and it is all correct?
- Have you nominated **professionals inside your organization who will be the leaders of edinn**? Their role will be to:
  - Take care of edinn (like if it was another machine of the facility) to assure that it is working properly, containing correct data, generating valuable information and satisfying the users.
  - Become the unique point of contact with edinn.
  - Please remember to nominate as many professionals as possible if your organization has a high personnel rotation.
- **Ethernet Data Network:**

  You need an ethernet computer data network to connect the machinery and the operation terminals with the EDINN server.
  - It is recommended that you separate your “office” network from the “production” (workshop) network. There are many ways to do this, but a simple way is to just create a different data network for the workshop. For example,

the “office” data network could be 192.168.1.X, and the “production” data network could be 192.168.2.X. The easiest way to do this is by creating 2 separated data networks, installing 2 network cards to the edinn server and connecting each of these cards to each network, making this server the unique common point between these 2 data networks.

    - Have you requested, to the computer network responsible, the **fixed IP addresses** for the eBOXes and/or PLCs and the server?
    - Do you have available an RJ45 plug, or a Wi-Fi access point with RJ45 (not recommended to be USB), to supply data network to each of the operation terminals?
    - Do you have available an RJ45 plug, or a Wi-Fi access point with RJ45 (not recommended to be USB), to supply data network to each of the monitoring devices (eBOXes or PLCs)?

- **Power supply:**
    - Do you have available a plug to supply electricity power to each of the operation terminals?
    - Do you have available a plug to supply electricity power to each of the monitoring devices (eBOX or PLC)?
- **Minimum requirements of the Internet connection:**
    - When your **edinn server is cloud:**
        - Have you tested that the minimum download and upload bandwidth are achieved?
            - Upload: 3600 bits / second (normally is half of this).
            - Download: up to 350 K bits / second (normally is half of this).
    - When your **edinn server is local**:
        - If you require access to your edinn web reporting from outside your organization: have you opened the necessary ports in your firewall (normally is port number 80)?
        - If you require using the edinn eCloud Sync to be able to access your edinn web reporting tool from outside your organization and to assure a backup copy of all the data external to the organization: have you opened in your firewall the necessary ports? (please request to your installer what will be this port number).
- **Hardware minimum requirements. All hardware requested to edinn will comply with these minimum requirements:**
    - For terminals or monitoring elements like sensors and PLCs: have you checked

that these elements have the required minimum specifications?

- Have you informed to the edinn installer about the work load that you estimate for your edinn server? Including:
  - The number of users that will use the system: number of terminals and number of users of the web reporting tool.
  - Software modules that will be activated, or that you think will be activated in the future.
- If you are going to use your own monitoring elements (eBOXes or PLCs): have you assigned fixed IP addresses to these elements?
- When your **edinn server is local**:
  - Minimum requirements for an edinn Server can be found in the Terms and Conditions.
  - Are you aware that the backup copies, the availability of the server and the possible redundant elements to guarantee its availability are under your responsibility?
    - Have you decided where will you store the files of the backup copies? It should be a safe storage unit outside of the server, and if possible, outside of the work center.
  - Have you considered the elements (Uninterruptible power supply, redundancy, etc.) that you require guaranteeing the maximum availability of the server?
  - Have you assigned a fixed IP address to the server?
  - The edinn support team uses its own tool (ISL) to provide remote on screen support:
    - Do you authorize the use of this tool, or should the edinn support team use your own remote support tool?
    - Have you opened the TCP/IP ports which are necessary for this remote support tool (for the edinn's support tool, ISL, these ports are TCP 7615, 80 and 443) or the ports of your corporative remote support tool?

- **Operation Terminals. There are 3 possible applications to use the edinn platform:**
  - **Web reporting tool**: edinn web reports are multiplatform and therefore they should run on every explorer, as long as it is not very outdated.
  - **Edinn terminal** for Windows (**full functionality**): we normally provide or recommend robust touch Microsoft Windows PC terminals which are IP-65 certified (protected against dust and water), at least in the front, and, therefore, there is no need to put the PC terminal inside a cabinet. The

terminals can be placed on a table, or attached to walls or to metal cabinets. Please see picture below.

- Minimum requirements for an edinn PC terminal can be found in the Terms and Conditions.

- **Edinn terminal** for iOS, Android or Windows (**limited functionality**): PCs, mobile phones or Tablets can be used.

- **ERP Integration**: the edinn platform has available a module which makes it compliant with the ISA-95 Standard. Please request more information if you are interested in connecting your edinn platform with your ERP (Microsoft Navision, SAP, etc.).
  - You can find the integration manual in the following link (only in spanish): https://edinn.com/dwn/recursos_web/ACADEMY/Instalador/Manual_Integracion_edinnM2.pdf (In order to consult it you will need an access key; Please contact our staff to get it).

- **Production interruption**: in most cases edinn can be installed without having to stop the affected processes. But sometimes, this is not the case:
  - To minimize the impact on production it is necessary that you schedule, with your edinn authorized installer, the possible stops of the machines where the edinn system will be installed.

- **Training**: after installing the devices, verifying the installation and adjusting configuration, the training is performed to the people that will use the system, for them to know how to get the most out of it.
  - Have you designated the persons who should receive the training, considering these profiles?
    - **Operators and supervisors**: they will need approximately between 30 and 60

minutes of training.

- **Management and supervisors**: they will need approximately 60 minutes of training.
- **Administrators of the system**: they will need a minimum of 2 hours of training.

- Have you considered that these persons will not be able to work while they are in this training?

- **Support**: once the system is installed, for questions, issues or improvement suggestions, please use the edinn on-line support system available here: http://ecloud.edinn.com/support/

  Please do not use any other channel to request service, like email or telephone, as to do so will worsen our service to you, due to the fact that the follow up is much more difficult through these other channels.

  You need a user and password in order to use the edinn on-line remote support tool. If you do not have them yet, please request them by sending an email to support@edinn.com, indicating:

  EMAIL:
  NAME:
  LASTNAME:
  NAME OF THE COMPANY:
  LOCATION:
  MOBILE TELEPHONE NUMBER (as dialed from another country):
  FIXED TELEPHONE NUMBER (as dialed from another country):.

Factors for success

In our experience, based on hundreds of projects, these are the **factors, in descending order of importance**, for the success of an edinn project:

1. Once edinn is installed, and after an adaptation period (for example 3 months), force the work center where edinn was installed to have their ratios (OEE and other), and their trend, to be available remotely through the edinn, and to be correct according to the diagnosis and trends reports.
2. The work center is producing normally that which is about to be controlled and improved with edinn, at least, 1 month before installing edinn.
3. The work center incorporates the ratios provided by edinn into its daily management routines.

4. Correct management of those professionals which prefer not to be objectively evaluated and that do not want their superiors to have objective and real time information about their performance.
5. Minimize personnel rotation or insufficiently trained on edinn. If this is not possible, it is recommended to:
    1. In organizations with multiple work centers (multi plant), train a small, specialized and centralized team which can provide support and lead the installation and use in case any of the work centers could not because of personnel rotation.
    2. To nominate multiple edinn local leaders in each work center.
6. Sufficient training of the edinn local leaders in the work center.
7. In organizations with multiple work centers, to have a small, specialized and centralized team (at least 1 person, but it could be more depending on the needs) to be, if necessary, 100% dedicated to the edinn platform, and to be trained and capable to support the work centers.
8. Enough visibility of the edinn platform in terms of flat screens or screen savers in the terminals to get profit of the motivation that the edinn platform can provide.
9. Wide and successful experience installing edinn of the installer team, with no linguistic barriers with the local edinn leaders.
10. The edinn platform relies on multiple systems which must be fast and robust, because any slowness or failure of these systems will be attributed to edinn and its partners, although it is not being caused by the edinn standard. This happens because edinn is the visible part of many back-end systems, for example:
    - PLCs, OPC Servers and, in general, systems or developments for data acquisition.
    - Computerized infrastructure like servers, terminals, data networks, etc.
    - Customized developments (API and/or UDL).

Monitoring

These tasks prepare the environment so that the edinn platform can read data from it in real time.

What signals do we need to monitor?

Usually, the first step is to measure the **OEE** ratio, and to do this, it is necessary to measure its **3 components**:

AVAILABILITY:

To calculate this component, edinn needs a **real-time counter** of the **total results** produced by the process. For example: *a counter of total pieces produced, updated in real time, each time a piece is produced.* Usually it is recommended to use the same counter which is already being used for the daily counting of the process, and read this counter with the OPC Server.

With this counter, the edinn platform **will know** if the process is **working** or **stopped** and the **speed** of it, but it will not know **why it stopped**. For edinn to know **why** the process stopped, it will need **signals**, available through the OPC standard, which activate when a certain type of stop occurs. **If you do not** have these signals, which is something very frequent, then you will need **a user to justify the stops manually** from the terminal.

SPEED:

This component is calculated **automatically** based on the production counter on the edinn platform.

QUALITY:

To calculate this component, **real-time counters** will be needed for **discarded** or **reworked** parts of the process.

This means to set up sensors to count, in **real-time**, **discarded production** in those points where production can be discarded because of **quality problems**. For example*: where production is derived to a conveyor belt, or to a basket, when it is not meeting a certain quality requirement*. **Multiple discard counters**, of a general type, or one for each type of discard can be provided, according to the different quality measurements: *incorrect weight, detection of unknown elements by X-rays, etc.*

There is another way option the edinn platform to calculate the defective production, and it is **by difference of 2 counters**: one located at the entrance of the process and the other at the end of it. The edinn platform will calculate the discarded production by the difference between the first and the second counter. This option will not allow edinn to calculate the discarded production as frequently as with having real-time counters of discarded production.

Optionally, to have **additional functionalities** of the edinn platform (*Predictive Maintenance, Quality SPC, etc.*), more counters and signals can be provided to the system, for example:

- **Electrical signals** with production values (pressure, temperature, etc.) that you need to monitor continuously.

- **Output signals** to activate beacons or sirens.
- **Energy or resources consumption counters:** electricity, water, etc.

How a counter must be?

- **Count always forward.**
- **Not count empty cycles:** edinn should not receive production counts which do not correspond with real produced results.
- **Count all the results produced by the process, and not only the good ones.**
- **Not be resetted or, if it does, it should reset only at a fixed account:** in addition, in case it is resetted, this count should be large enough so that the edinn platform can be a while (for example, 10 minutes) without reading the counter and that it has not been resetted 2 times in that period, which could cause a data loss.
- **Recommended to be accessible through the OPC standard:** if you do not have this counter, you can connect the signal that serves to have this counter to an edinn eBOX (which already includes this counter internally) or one of your PLCs, and build this counter with the following characteristics (please, request if you need OPC UA).

> **NOTE:** for more information, please visit: http://opcfoundation.org/

Need help? 

Scenarios for new and current machines

2 possible scenarios/monitoring options:

1st scenario. We are going to acquire a new machine: what specifications do we need to know about the new machine in order to monitor it? We can follow the following steps:

1. Inform the machine manufacturer that you are using the Industry 4.0 edinn platform and that this system needs to **read**, in **real time**, and through the **OPC standard**, all the **data and variables** of the machine.
2. If the manufacturer is concerned about the possibility that the edinn platform can write or change some parameters of the machine, we must specify that the edinn platform, unless it is developed specifically, only uses **read mode**. If still the

manufacturer remains concerned, there is the possibility of creating a **copy of the data** in a memory space dedicated to edinn, so that the manufacturer's data would always be kept without access and without risk of change of any critical data.

3. Inform the manufacturer that it is necessary to access the **machine information** using the **OPC DA standard**.

> **NOTE:** for more information, please visit: http://opcfoundation.org/

4. The manufacturer must provide the **OPC Server software** for reading the machine data.

5. Inform the manufacturer that you need the **map or complete list of items within the PLC**, with its name OPC, type, size and any other particularity that must be configured for reading from the edinn platform. Therefore, in this option, you must receive from the manufacturer, an installable OPC Server software (check the conditions of the license) and the map or complete list of items within the PLC (check that the list is self explanatory).

2nd scenario. We already have the machine in our facilities: how do we monitor it? There are mainly 4 possible options:

a) Contact the machine **manufacturer** and follow the steps of the previous option (A). If it is not possible, then let's try the following options:

b) Find out the **PLC brand** (if we do not have eBOX) of the machine from which we want to read the data, and **find, install and evaluate** (some providers offer a free evaluation period) an **OPC Server** that is able to read the PLC data.

c) **Connect** the sensors and signals **available** on the machine to an **edinn eBOX** and continue with this step of this tutorial.

d) Install **new sensors** on your machine and connect them to an **edinn eBOX** and continue with this step of this tutorial.

Need help? 

eHUB

Edinn Hub, eHUB, is an industrial microcomputer that can act as:

1. A device that collects all the electrical signals.

2. A local edinn M2 server, the "control room" of all the edinn services.

Edinn HUB (eHUB) frontside view

Edinn HUB (eHUB) backside view

NOTE: For more information on the eHUB device search for the “eHUB specifications” in: https://edinn.com/hardware/

Quick guide

In this section we explain how to easily set up your eHUB and get started with the installation of edinn.

1. Install the eHUB and the eBOXes:

The eHUB device can be safely installed:

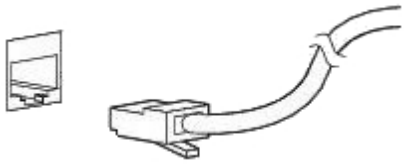
- Either on the plant’s shop floor.
- Or in the plant’s servers’ room.

WARNINGS:

- Both of these options need to ensure a continuous power supply and internet connection.

- The eHUB must always be connected to the same ethernet network as the eBOXes.

2. Connect the eHUB device:
    - To the power supply.
    - To the production data network, where the eBOXes are connected, using an RJ45 cable.
    - To the offices data network, where there is internet access, using an RJ45 cable.



RJ45 cable & socket

NOTES:

- Either connector from the eHUB can be used for the production and the offices data networks.

- For more relevant information on connectivity, please read “Power supply”, “Ethernet Data Network” section here in the Checklist.

1. Connect one or more eBOXes: for information on this topic read the Monitoring, Using eBOX help documents.
2. Reserve fixed IPs and inform edinn:
    a. Reserve fixed IPs for each eBOX (sold together with the eHUB or separately).
    b. Access your account in edinn Support at: https://ecloud.edinn.com/support
    c. Inform edinn, **through the support previously indicated**, that the eHUB is ready and **provide the list of eBOXes with this information**:
        - what machine, line or process they are attached to,
        - a reserved IP for each one.

    You can use the following table as a guide:

Machine/ Line/ Process	IP

MACH_01	IP_01
MACH_02	IP_02
...	

3. Work together with edinn to start and maintain your platform successfully.

Using eBOX

Edinn BOX (eBOX) is a device provided by edinn which allows, among other functionalities, to send data from electrical signals to an edinn server.

edinn BOX (eBOX).

Detail of the leds power and communication ok with the server, of the edinn BOX (eBOX).

Connect

WARNING: To avoid the **risk of electric shock**, only operate the eBOX only when unplugged from electricity.

WARNING: this document explains the steps to monitor with edinn **eBOX**. To monitor with other PLCs contact your supplier or request it here.

1. Select the signals to monitor

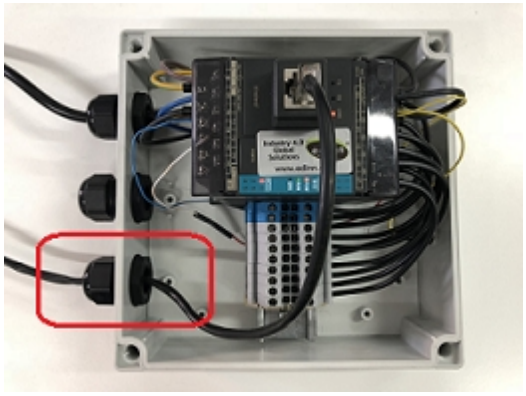
For this, see the section of this guide: What signals do we need to monitor?

2. eBOX basic connection

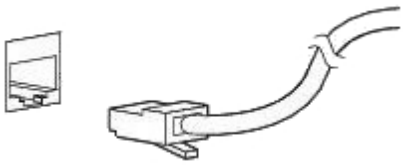
Let us connect the edinn **eBOX**:

1. Keeping the eBOX unplugged, unscrew the cover and removeit:

2. Pass the ethernet cable through the cables hole.



3. Plug the Ethernet cable (RJ45) to the eBOX to connect it to the Internet:



It is important to know what kind of RJ45 cable it is necessary to use:

- If the connection is made **from the PC to a switch**, and **from a switch to the PLC**, **straight RJ45 wires** are necessary for both connections.
- If the connection is made **directly from the PC, eHUB or server, directly to the eBOX, a crossover RJ45 wire** is needed. Usually, in modern equipments, this rule is not mandatory, but, if one of the network devices is too old, connection problems may arise.

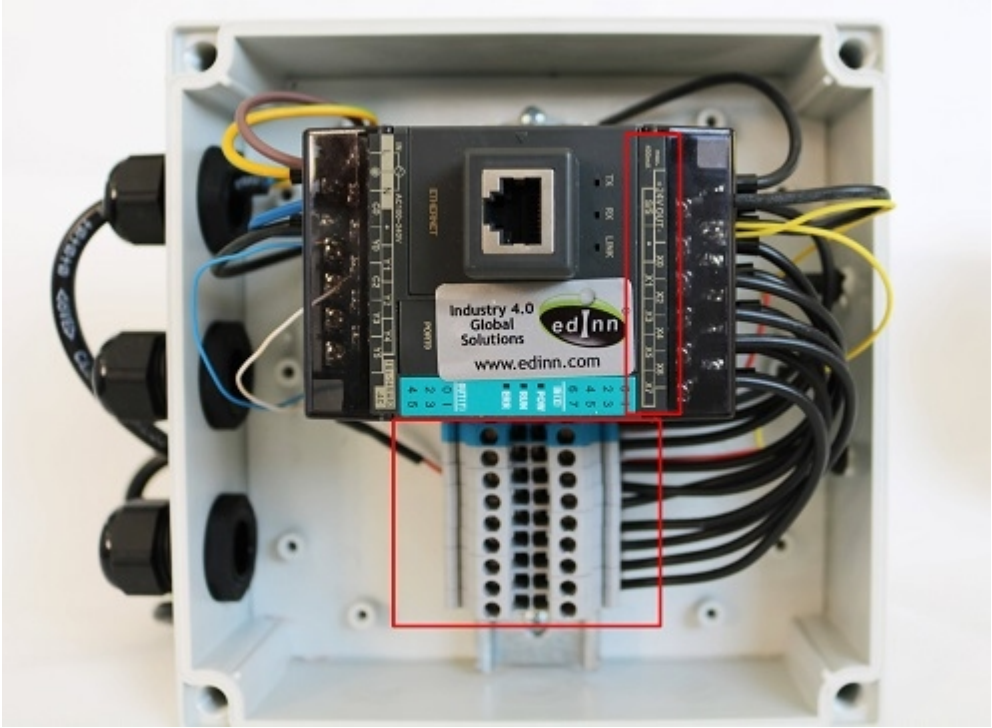
WARNING: Be sure that the RJ45 connectors and wires are in good condition and protected, otherwise, communication errors could occur.

4. Put the cover back, screw it and plug the eBOX to the electrical power:

3. Inputs

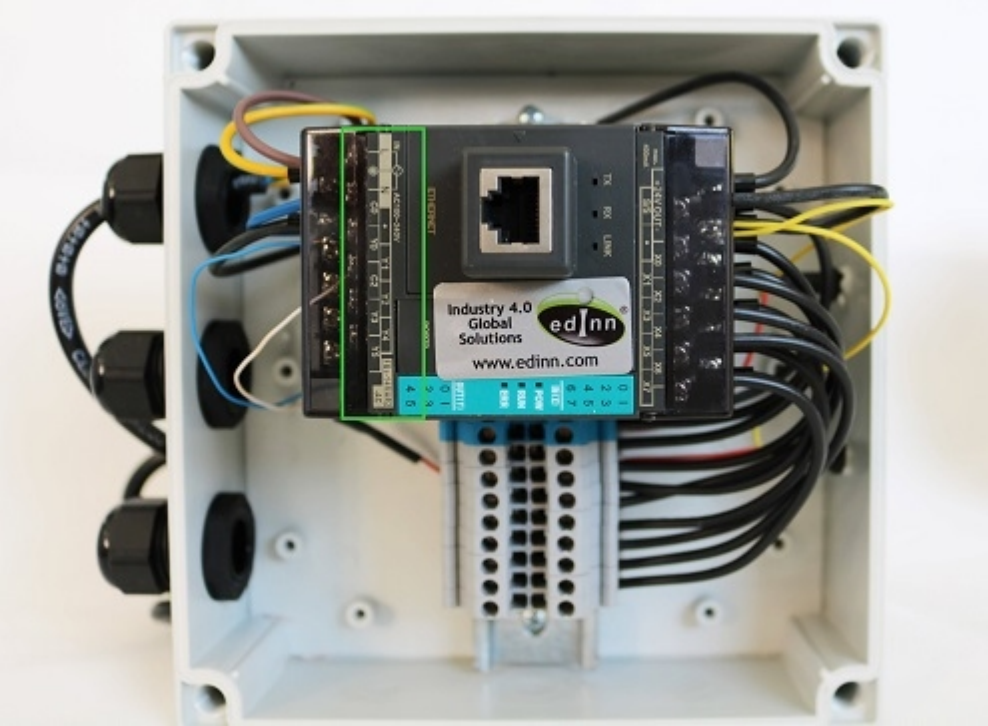
Each **input** of the eBOX (x0, x1, .., xN) is wired to the **DIN rail connector**. In this case, it has 8 inputs (it will depend on the acquired eBOX).

Wire and attach the selected signals from the external device to the correct connector on the **DIN rail**. Write down this relationship, since it will be needed later.



4. Outputs

In the case of needing to use outputs (*beacons, alarms, etc.*) you can use the outputs of the eBOX, marked in green on the image below.



For more information about the eBOX, check the document **eBOX_200B_specs_en.pdf** from the edinn Academy, IoT area here

Need help?

Input: sensor

WARNING: To avoid the **risk of electric shock**, only operate the eBOX only when unplugged from electricity.

In this procedure we will see how to connect a sensor to eBOX through an input.

1. Select the sensor

For this case, we will use this photoelectric sensor.

If you look at its legend, there are 5 connectors:

1. BW (brown cable) [Positive power supply L(V+)]: Common cable connected to +24 V
2. WH (white cable) [Normally opened Contact]: Cable that sends 24V to the PLC when the sensor is excited and 0V when not.
3. RED (red cable) [Voltage input to the sensor's circuit]: Positive input cable to the sensor's working voltage.
4. BK (black cable) [Normally closed contact]: Cable that sends 0V to the PLC when the sensor is excited and 24V when not.
5. BL (blue cable) [Neutral power supply N(V-)]: Common neutral input cable.

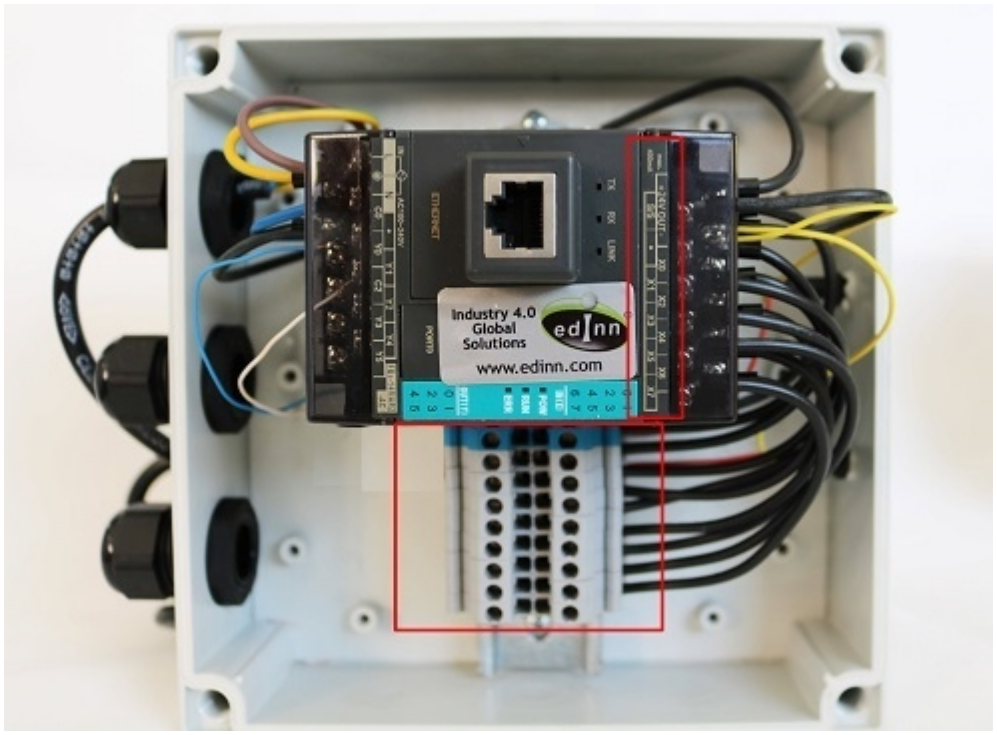
WARNING: Your sensor could have different colours for the same or different cables. Therefore, before starting the procedure, check and identify in the legend of the sensor to which cable of the indicated here correspond each one.

2. Basic sensor connection to eBOX

Let us connect the edinn **eBOX**:

1. Unscrew and remove the cover:

2. View, inside the eBOX, which connector is the +/-24V output and the XN signal. N being the number to which you want to connect the sensor:

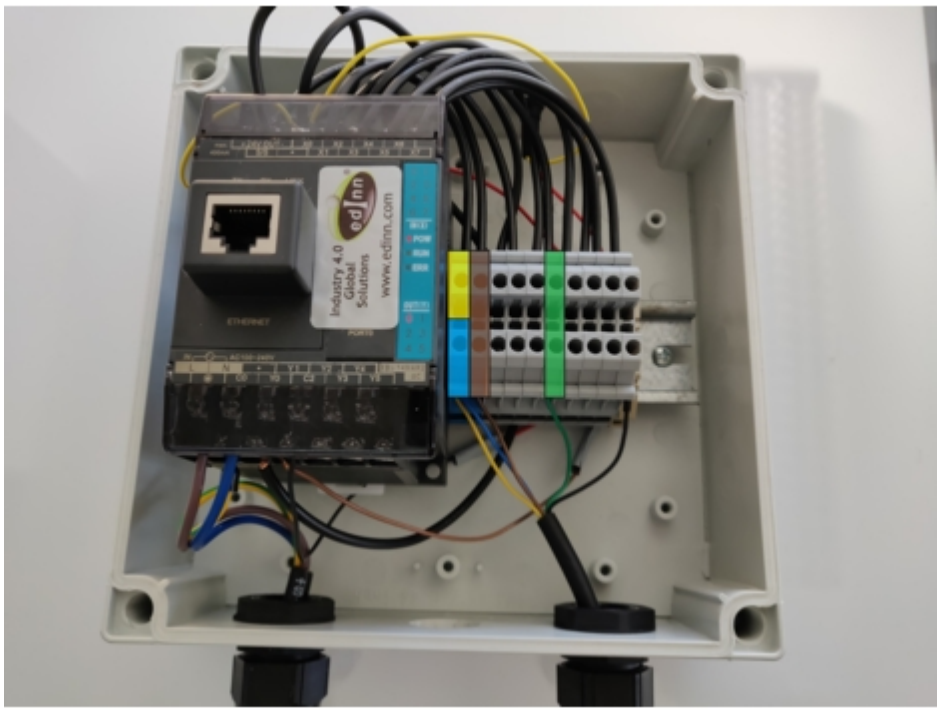


3. Pass the cables of the sensor, 5 in this case, through one of the holes (cable gland) of the eBox.

4. Connect the blue cable to connector marked in blue in the following image (output +24V) and the brown cable to connector marked in brown in the following image (output -24V):

5. Connect the yellow cable to the connector marked in blue in the following image (output +24V):

6. Connect the green cable to the entry to which you want to send what is captured by the sensor. In our case we choose the input X3 marked in green in the following image:



7. End of the procedure.

WARNING: Check with a multimeter that the eBOX is reciving +24V each time the sensor detects a pulse through the reflector.

For more information about the eBOX, check the document **eBOX_200B_specs_en.pdf** from the edinn Academy, IoT area here

Need help?

Ethernet

NOTE: If you choose full service with an eHUB, you do not have to perform the following steps.

Your eBOX must have an **internal fixed IP address** and **port** inside your local network**.**

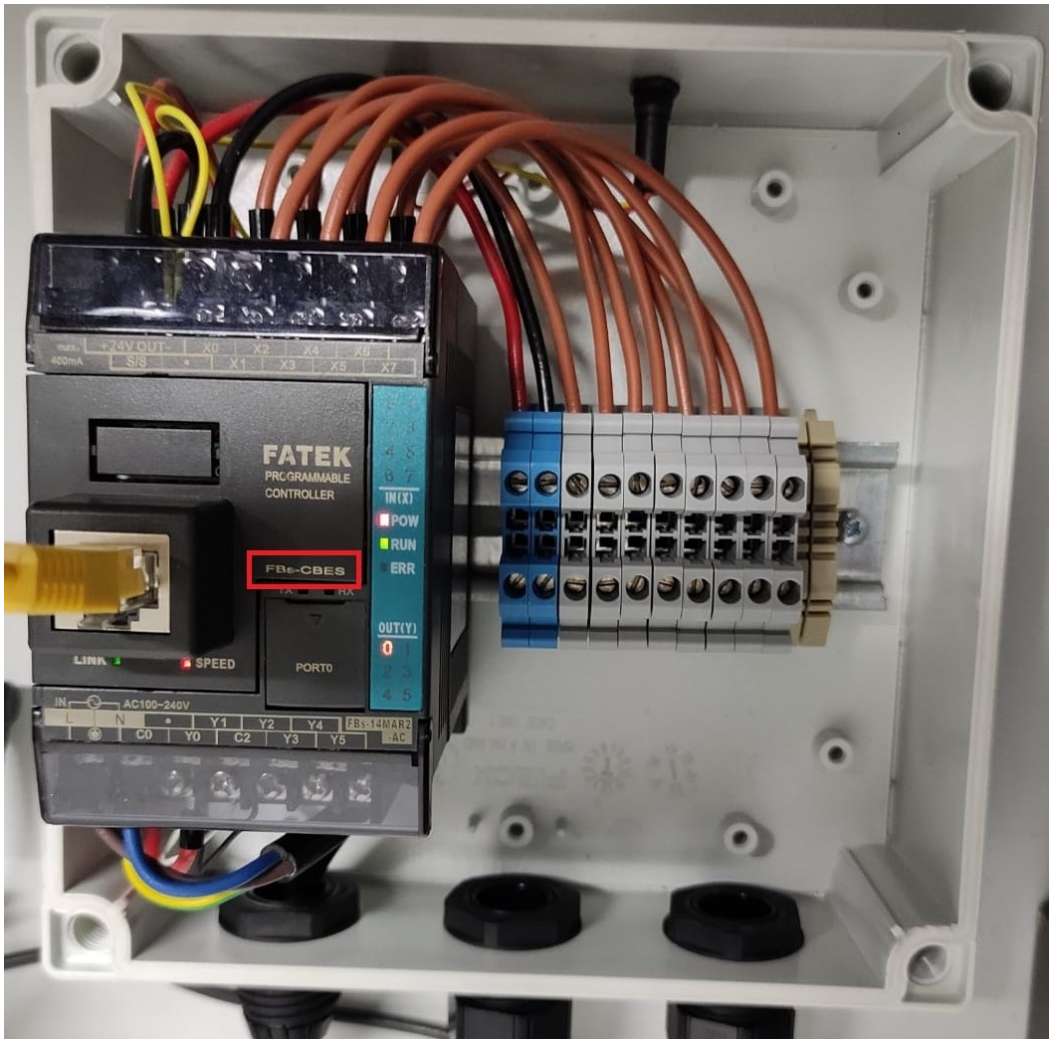
Therefore, you need to change the IP address and port of your eBOX to be compatible with your local network.

Edinn works mainly with 2 PLCs in its eBOX:

- Fatek FBs-CBE (From the beginning). It has fixed IP: 192.168.1.3

- Fatek FBs-CBES (Since 2023). Has fixed IP: 192.168.2.3

If you do not know the model of the PLC installed in your eBOX you can check it by opening it. It will appear in a place like the red box in the image below.



WARNING: This section is written for the old model, the Fatek FBs-CBE. However, for the new model (Fatek FBs-CBES) the process is the same taking into account the default IP (192.168.2.3) of the new PLC instead of the old one (192.168.1.3).

1. IP configuration

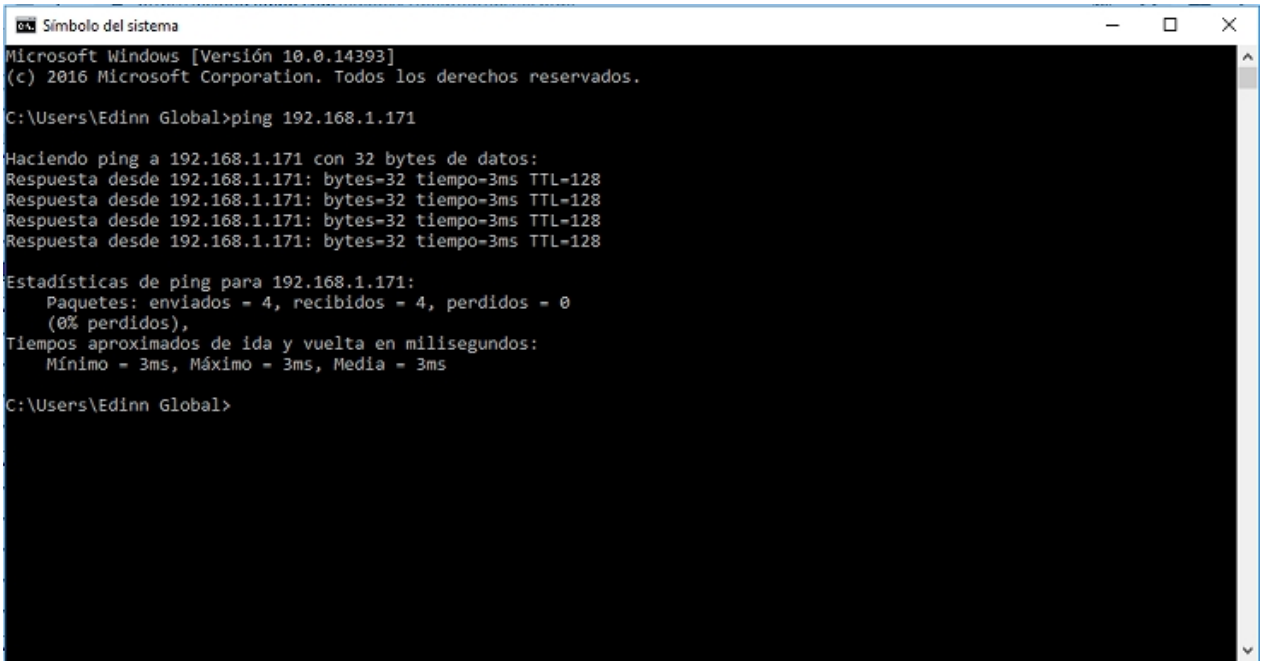
Once the connection is done, in order to allow the communication between the PC and the eBOX you should check the PC IP addresses.

- The PC IP has to be 192.168.1.X (Not 192.168.1.3 as it is the eBox IP)
- The Subnet Mask should be 255.255.255.0
- The Default Gateway should be 192.168.1.X (Not 192.168.1.3 as it is the eBox IP)

In order to assure that previous steps have been followed and to check the connectivity,

follow these steps:

1. Windows Start -> Execute cmd.exe
2. Write exactly the command: ping 192.168.1.3
3. Check that messages indicating that the ping to the ip is succesful



If a host error or a different one appear, it is highly recommended to deactivate Windows Firewall.

If all the previous steps have been followed but an error occurs you can create a ticket to solve your problem.

WARNING: WiFi connection must be deactivated, as it can cause a communication conflict from the PC to other devices.

WARNING: It is very important that the IP address 192.168.1.3 is free in our network, as it is the default one for the eBox.

2. Fatek Ethernet Module Configuration Tool

Downloads can be obtained from the edinn Academy, **Iot\eBOX\Ethernet Module Configurator** here

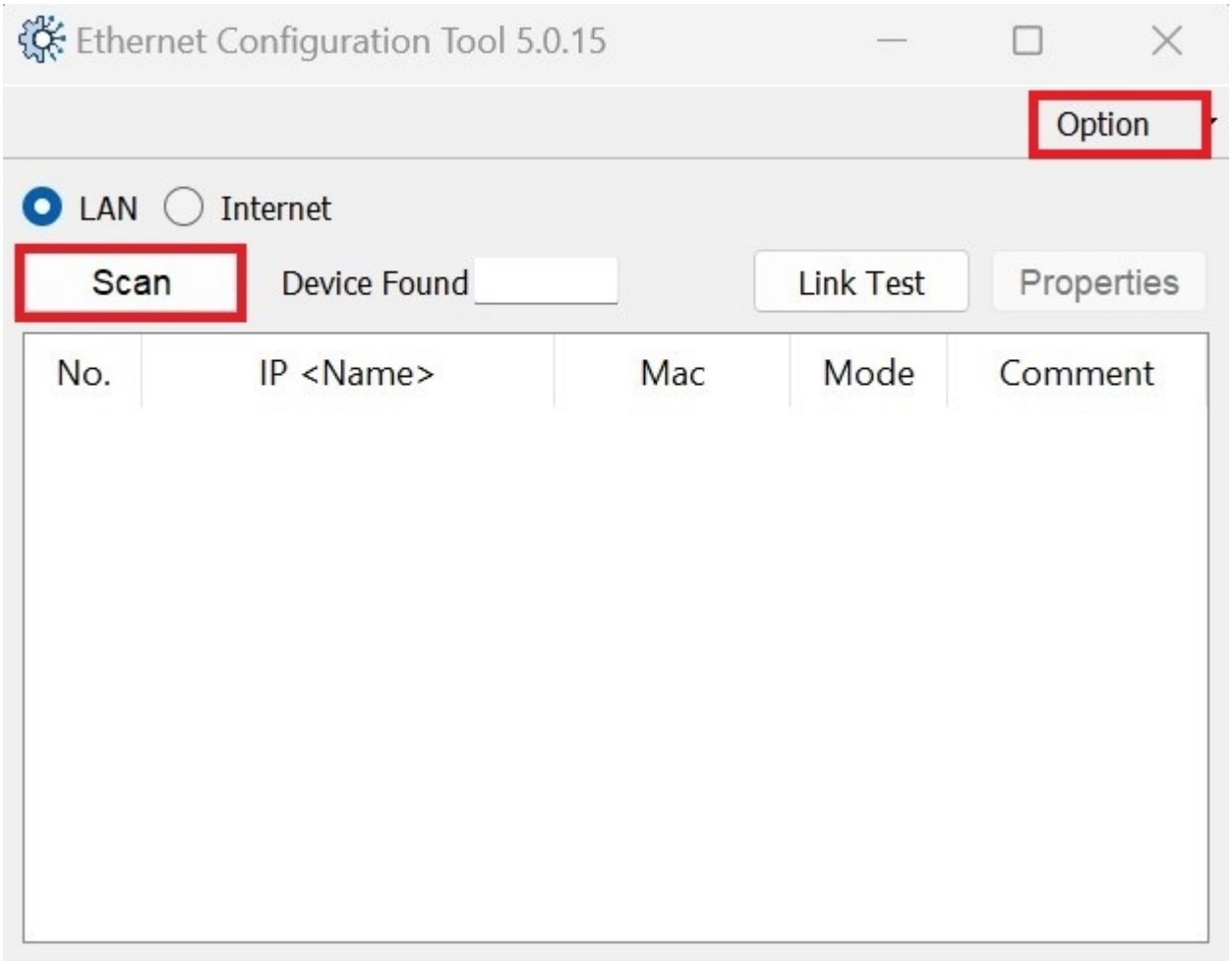
- Download and run with administrator permissions "Ethernet_Configuration_Tool_v5.0.15_installer" to install.
- Search and run with administrator permissions "Ethernet Configuration Tool".

3. eBOX detection in network

Connect the computer to the network where the eBOX is located in the **Option/Select Network** drop-down menu.

Choose the **LAN** option, which detects any eBOX connected to the network the computer is on, and click the **Scan** button. Note that eBOXes come with 192.168.1.3 as the default IP address, so in order to detect and change the eBOX IP, connect it on a local network with IP range 192.168.1.X, and no other devices on the network with IP address 192.168.1.3.

WARNING: it is advisable to disconnect the computer from any other network, to avoid IP conflicts.

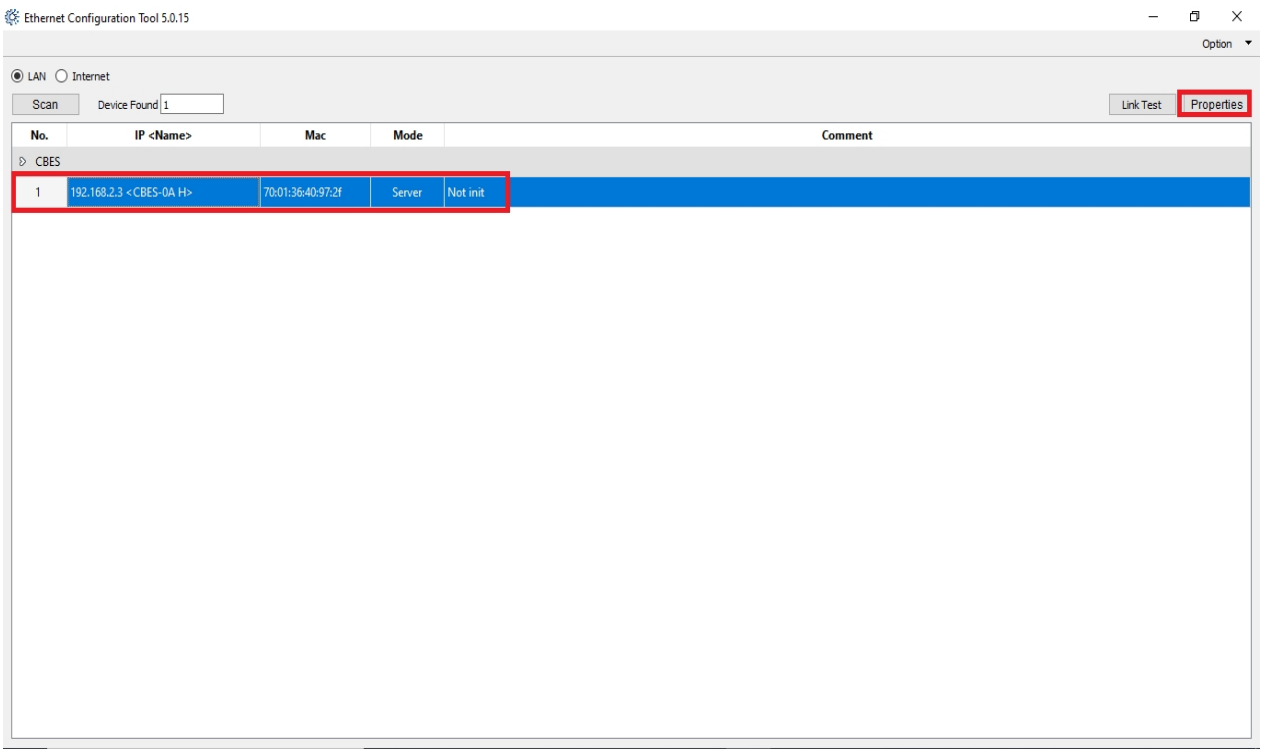


If you can not find the device, it is recommended to check the configuration of the computer IP, check that the PLC lights of the Link flashes on the Ethernet card, check the network cable and that the adapter is correctly installed in the PLC.

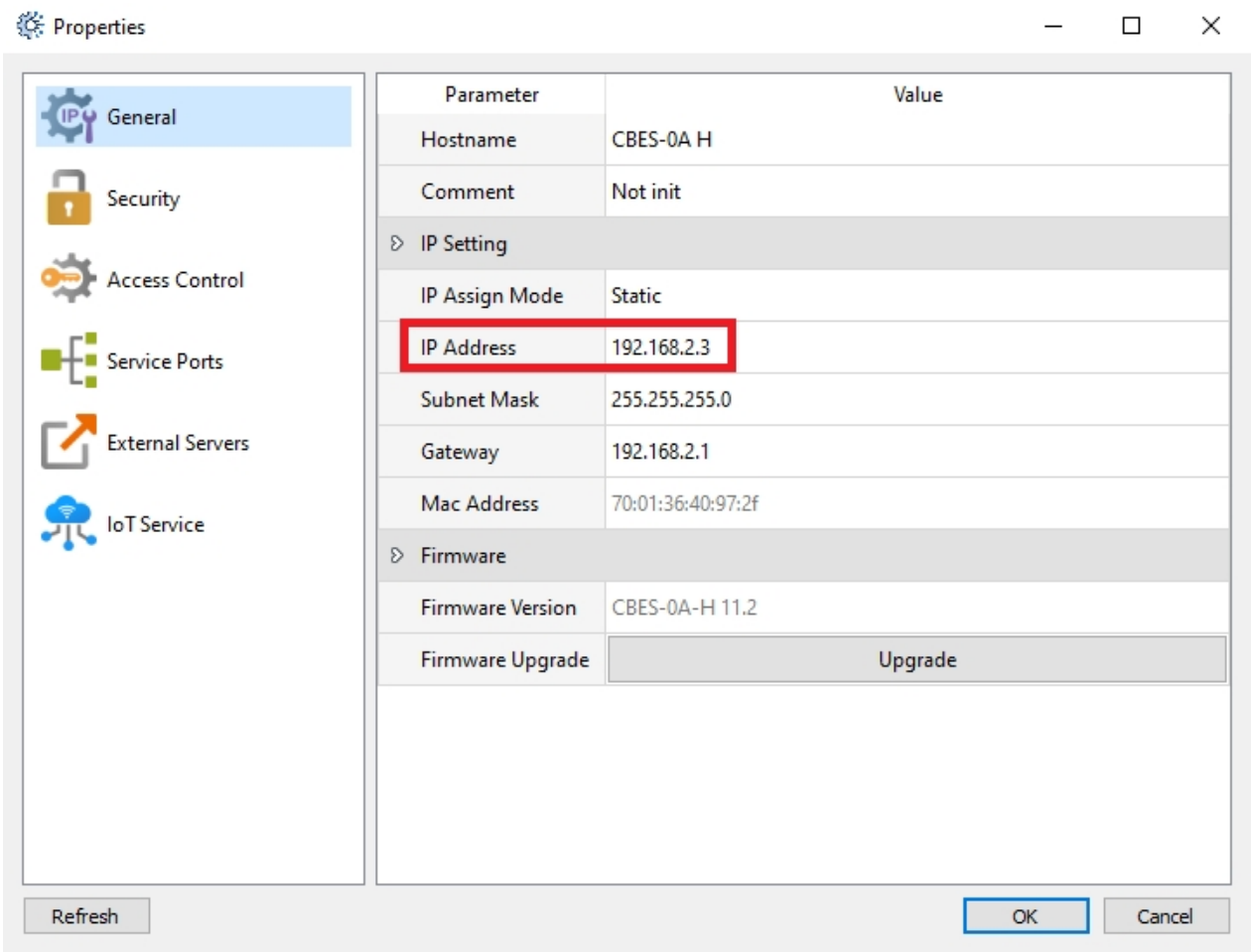
4. Select the eBOX

All eBOXs should appear in the list. Choose the one where you want to change the IP address and click the **Properties** button.

If a **password** is required, the default value is 1234.



5. Properties



You can change the following 4 fields (check them with your **Local Area Network administrator**):

- **IP Address**: where we will change the default IP to the fixed free IP necessary for our installation.
- **Subnet Mask number:** where we will define the logical division of the IP network in which the PLC is located.
- **GateWay**: where we will place the default Gateway address for our installation.
- **Host Name**: where we will define a name to the PLC, for example, the name of the process that it will monitor.
- **Comment:** if we want to add a comment on the PLC to identify the eBOX in the future.

Once configured, press **[Ok]** and then **[Exit]**

WARNING: changing any other parameter may cause the eBOX to NOT WORK and it might need to be sent back to edinn for reparation.

Need help? 

Using PLCs

NOTE: If you choose full service with an eHUB, you do not have to perform this step.

A PLC (Programmable Logic Control) is a device which allows, among other functionalities, to send data from electrical signals to an edinn server. If you need to prepare the monitoring for a specific PLC, the steps should be similar to those indicated in this manual for the eBOX.

Need help? 

Using simulation

The edinn platform is capable of receiving data from a simulation exactly as if it was the real thing. Simulations can be run with third party softwares like FlexSim or by your own application. More information will be given later in this manual.

Need help? 

Counting problems

When the count of produced results in a process does not match the total of results in edinn, these are the verifications to be performed based on the most common reasons:

1. **Check if the pulse is really arriving to the eBOX**: for example, if the sensor is photoelectric, some results could have a label which reflects the beam of the sensor and that prevents the sensor from counting. You can check if the PLC is really receiving the pulse by checking the LED on the input of the PLC (inside the eBOX) every time a result passes by the sensor. The typical fix to this problem is adjusting or changing the sensor to a most suitable one.
2. **Check if the frequency of pulses is higher than the one accepted by the eBOX or its program**: in some cases, when results' pulses are generated fast (typically faster than 20 /second), although you can see the LED blinking, some pulses are being lost. If frequency of production is higher or similar to 20/sec., the typical fix to this is to change the program of the eBOX and reduce the time filters, or to change the eBOX to a faster one.

> **NOTE:** If you choose full service with an eHUB, you do not have to perform the following steps.

1. **Check if the counter or counters meet with the minimum requirements of** how a counter must be and if its **reset value is correctly configured in the** monitor**.**
2. **Check if the OPC Server is counting as many as you are producing**: once confirmed the previous steps, open the OPC Server and check, during a long enough period, that the results produced are all counted by the OPC Server. If the previous steps were not cause of problems and if the OPC Server is working correctly, there should not be any difference to the real production.
3. **Check that the same count of the OPC Server has arrived to edinn in results**: this step should only be performed after the previous steps have been performed. Reasons to possible differences could be:
    - **The process was in a status which was blocking results**. Please see status configuration.
    - **The process was producing faster than the rate accepted by the % of minimum cycle time configured in the monitor**. Please see "minimum percentage of the cycle time".
    - **A user has deleted results**: this can be checked by searching for the word

"deleted" in the logs of the server in the Console (please see how to view the server logs). All user manual deletions of records are shown that way in the server Logs independently of the log detail, user roles, etc.

Need help?

Create & Configure

These tasks create and configure a company in the edinn Platform.

NOTE: If you choose full service with an eHUB, you do not have to perform these steps.

Using edinn OPC Bridge

Edinn OPC Bridge is a software which connects your own OPC Server or simulations with an edinn server in the cloud. It needs to be installed locally in one of your local servers where the OPC Server software is installed, and then it will send and receive data to and from an edinn server in the cloud.

NOTE: If you choose full service with an eHUB, you do not need to use this component.

Using edinn BOX

Edinn BOX (eBOX) is a device provided by edinn which allows, among other functionalities, to send and receive data to and from electrical signals, to and from an edinn server. For more information about the eBOX, please read the specifications document present in the edinn academy.

In this option, the Fatek OPC Server software is used.

> **NOTE:** If you choose full service with an eHUB, you do not have to perform the following steps.

Introduction

> **NOTE:** If you choose full service with an eHUB, you do not have to perform the following steps.

In this option of installation, we will need these components:

1. **A company in the edinn cloud (eCloud).**
2. **An edinn monitoring BOX (eBOX).** If you do not have any, please request it at marketing@edinn.com

3. **The edinn OPCBridge software.**

Need help?

Step 1. Create & configure company

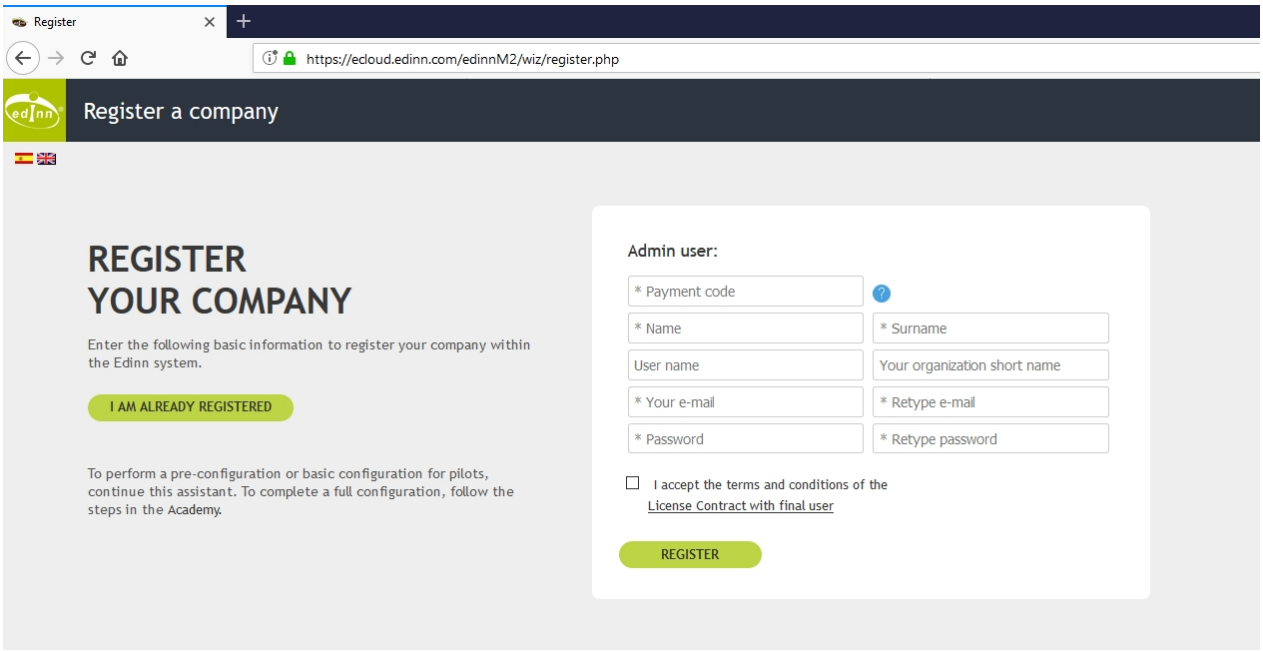
> **NOTE:** If you choose full service with an eHUB, you do not have to perform the following steps.

From an internet browser, visit:

- If you are a **Community** user, you will create your company in your local computer: http://127.0.0.1/edinnM2/wiz/register.php?lang=en
- If you have a **payment code**, you will create your company in the cloud: https://ecloud.edinn.com/edinnM2/wiz/register.php?lang=en

You can obtain your payment code here: https://edinn.com/online/?lang=en

In both cases, you should see a page like this:



Read carefully and follow the steps of the wizard **until the monitoring step**, and then continue on the next step of this procedure.

Need help?

Step 2. Install OPC Server & OPC Bridge

> **NOTE:** If you choose full service with an eHUB, you do not have to perform the following steps.

1. **Download & Install** the **Fatek OPC Server** from the edinn Academy, IoT area here 

   This OPC Server comes with a fully functional 30 days free evaluation period. It will require a valid license after that period and you should request it then to FATEK or to edinn.

WARNING: install the OPC Server on the same computer where you will install the edinn OPC Bridge, preferably on a local server or on a local PC that will be always running within the data network (Ethernet TCP/IP) where the eBOXes are connected, to have access to them and to not lose any monitoring data.

2. You have to **create** a blank database. To do this, go to **C:\Program Files (x86) \FATEK\FatekNetOPC** and execute **FatekOPC Configurator.exe.** Secondly, click on **"New"** at the top left corner and, without modifying anything, click on **"Save"** when the pop up window appears.

3. **Download** the **edinn OPC Bridge** from the edinn Academy, IoT area here 

4. **Unzip the file just downloaded and execute**, from the **edinn_OPC_Bridge_Setup folder**, the file **setup.exe** and follow the steps to install.

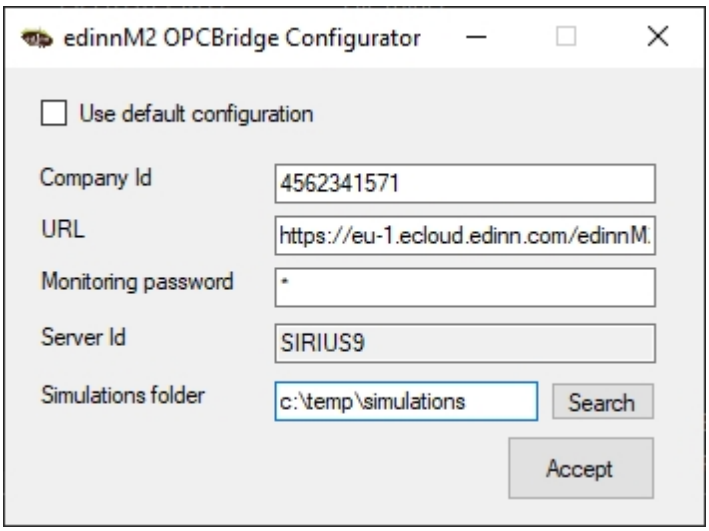
5. Once installation is finished, a pop-up window will show up to proceed with the OPC Bridge configuration. If this window would not show up, you can execute, with **administrator permissions,** the program **C:\Program Files (x86) \edinn\edinnM2\OPCBridge\edinnM2_OPCBridge_Configurator.exe** and follow next steps:

   A) Type the internal company Id of the company you created in Step 1: a 10-digit number available on the monitoring step of the configuration wizard and also on the e-mail that you received after creating your company.

   B) Type the URL based on the server where you created your company. For example, if you created your company in this server **eu-1.ecloud.edinn.com**, you should type: **https://eu-1.ecloud.edinn.com/edinnM2/monitor/**

   C) Type the password that you typed when you created the company in Step 1, which has also become the monitoring password, which can be different to your user password and can be changed in the monitoring step of the configuration wizard.

   It is not necessary to type anything in the "Simulations Folder" field. Accept the configuration.



If you have already installed **edinn OPC Bridge** and need to configure it again, click on your Windows Start button and type "edinn OPC" to find and execute the application **edinnM2_OPCBridge_Configurator**

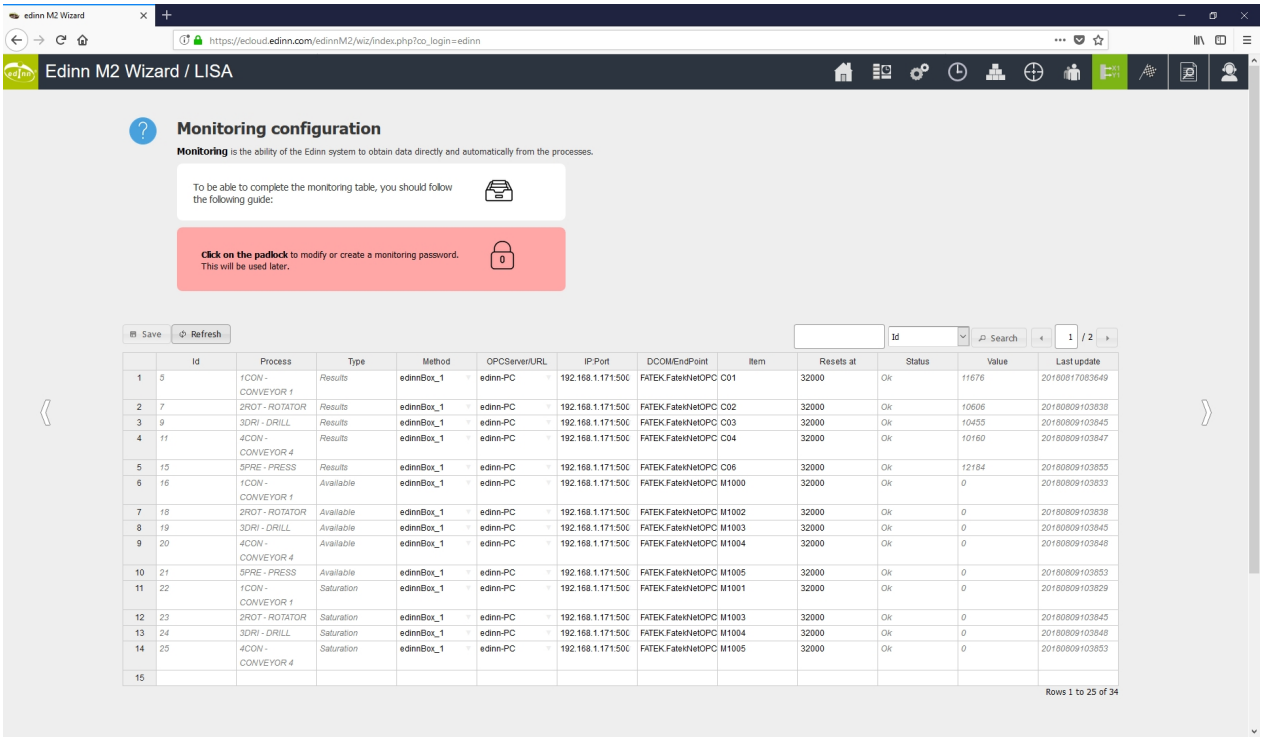
Need help?

Step 3. Connect and start

> **NOTE:** If you choose full service with an eHUB, you do not have to perform the following steps.

1. Go back to the configuration wizard, step of Monitoring Configuration.
2. Choose method **edinnBOX_1** for all the signals that you have connected **and save changes**. Please notice that only certain signals types are available from this wizard. To use all the signals types you need to access the console of the server, Monitor tab, or request help to edinn.
3. Click on **OPCServer/URL** to see the computers where **edinn OPC Bridges** are running and connected to this company. If **OPCServer/URL** is not showing the name of the computer where you installed the **edinn OPC Bridge**, be sure of having **saved the changes** in the previous step, click the Refresh button and try again. If the problem persists, please contact edinn.
4. At **IP:Port** column it is necessary to write the PLC IP address and Port in the following way: **XXX.XXX.XXX.XXX:YYY** where X are the 4 IP address digits and Y the Port digits. You can see an example in the image below.
5. Click on **DCOM/EndPoint** and select the **FATEK.FatekNetOPC.1** from the list of OPC Servers found by the **edinn OPC Bridge**.

6. Type the name of the item (also known as Tag) that you want to read. Please note that:
    - **Results**, **Scrap** and **Rework** items need counters, which Items (Tags) are like this **CNN**, where NN goes from 00 to 31.
    - The rest of the items (**Available**, **Saturation**, **Justify**, etc.) need signals, which items (Tags) are like this:
        - **XNN:** where NN goes from 00 to 31. These are to read **direct inputs** of the eBOX.
    - For example, if you connect the input number N from the eBOX to a certain signal, then that is:
        - **C0N**: if you want to use the counter which counts the pulses on that signal.
        - **X0N**: if you want to read the direct status of the input signal.
7. Click on the **Save** and **Refresh** buttons. Repeat after a few seconds until the **Value** shows the actual value of the item in the eBOX and the **Last Update** shows the actual date time. Once this happens, your system is fully configured:



Configuring the eBOX transmission LED

Monitoring eBOXes come with 2 LEDs: one indicating power on, and the other indicating that data is entering correcly into an edinn server.

Verify that the **two LEDs** on the **eBOX case** (power on and transmission) turn on.

The communication LED (transmission) will turn on automatically when using the **edinnBOX_1** monitoring method, or if the LifeBit signal is configured manually. This LED not only turns on by connecting the eBOX (PLC) to the data network, it also verifies that the monitored data is actually entering the server of the edinn platform and is being stored correctly.

If they **do not turn on**, please check:

- After connecting the RJ45 cable, it can take up to 1 minute for the light on the right to turn ON.
- eBOX is correctly powered on.
- There is internet connection on the RJ45 cable.
- In certain models, the TCP/IP port must be opened in your router and redirected to the eBOX internal IP address.

If still does not connect, please contact edinn.

Need help?

Using your own PLCs and OPC Server

Edinn can monitor any PLC by using the OPC standard. Please read here to find more information about OPC.

In this option, as an example, we will use an edinn eBOX as PLC and the Kontron OPC Server.

> **NOTE:** If you choose full service with an eHUB, you do not have to perform the following steps.

Introduction

> **NOTE:** If you choose full service with an eHUB, you do not have to perform the following steps.

In this option of installation, we will need these components:

1. **A company in the edinn cloud (eCloud).**
2. **A PLC which is compatible with an OPC Server.**

In this example, we will use an edinn monitoring BOX (eBOX). If you do not have any, please request yours at marketing@edinn.com

3. **An OPC Server Software.** In this example we will use the Kontron OPC Server.

4. **The edinn OPCBridge software.**

Need help?

Step 1. Create & configure company

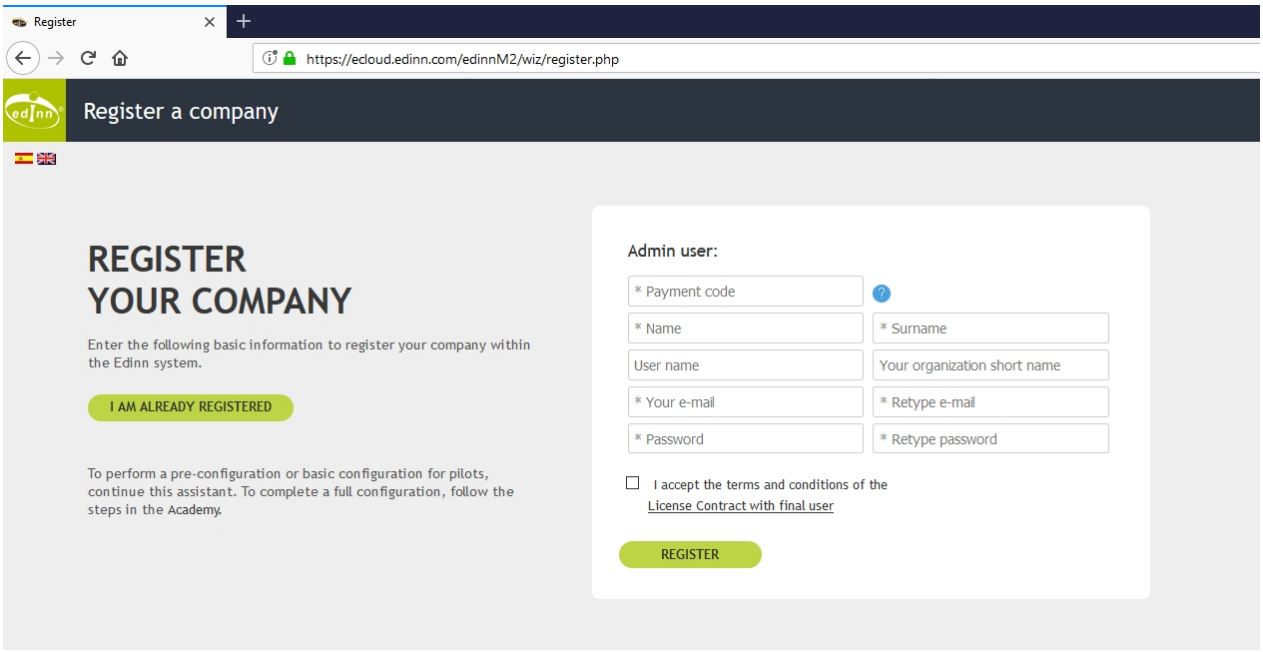
> **NOTE:** If you choose full service with an eHUB, you do not have to perform the following steps.

From an internet browser, visit:

- If you are a **Community** user, you will create your company in your local computer: http://127.0.0.1/edinnM2/wiz/register.php?lang=en
- If you have a **payment code**, you will create your company in the cloud: https://ecloud.edinn.com/edinnM2/wiz/register.php?lang=en

You can obtain your payment code here: https://edinn.com/online/?lang=en

In both cases, you should see a page like this:



Read carefully and follow the steps of the wizard **until the monitoring step**.

Need help?

Step 2. Install & configure OPC Server

> **NOTE:** If you choose full service with an eHUB, you do not have to perform the following steps.

1. **Download the Kontron OPC Server** from the edinn Academy, IoT area here 

2. **Open** the compressed file ***Kontron_Fatek_OPC_Server*** inside the OPC Server folder and execute with administrator permissions "***Fatek.EXE***", follow the instructions:

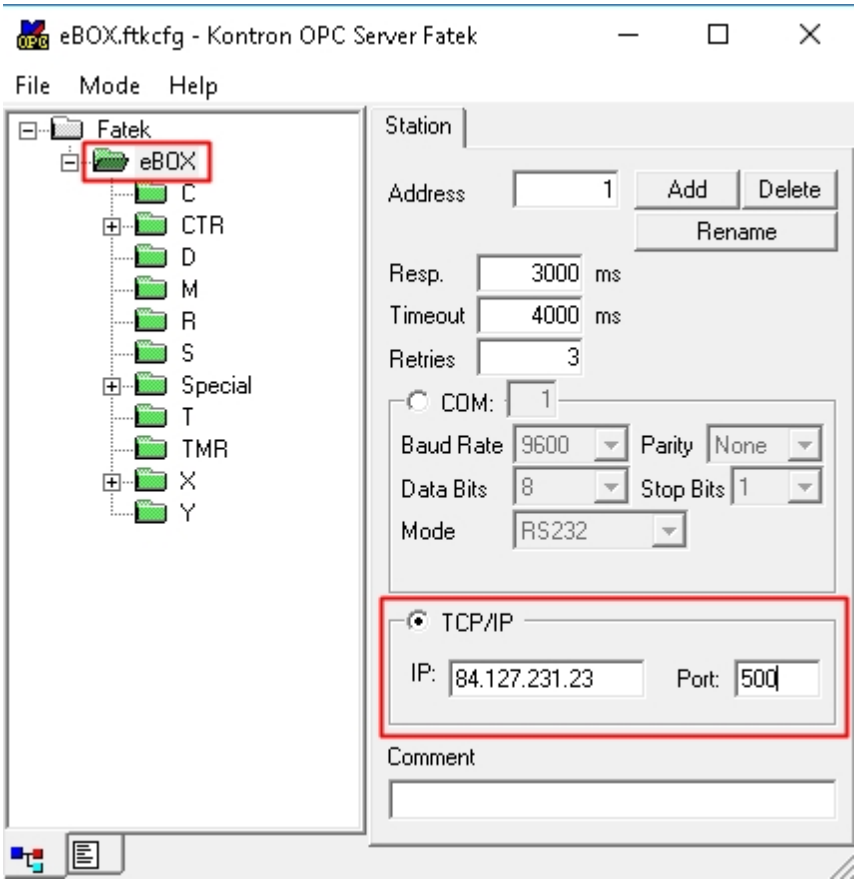
> **WARNING:** Install the OPC Server on the same computer where you will install the edinn OPC Bridge, preferably on a server or on a PC that will be always running within the data network (Ethernet TCP/IP) where the eBOXes are connected, to have access to them and to not lose any monitoring data.

> **WARNING:** This OPC Server is a proprietary software from a manufacturer named Kontron. This software has been discontinued by the manufacturer and edinn provides it here just as an example of an OPC Server suitable to connect with the edinn devices. If you have any other devices (PLCs), consult the necessary software with their manufacturer.

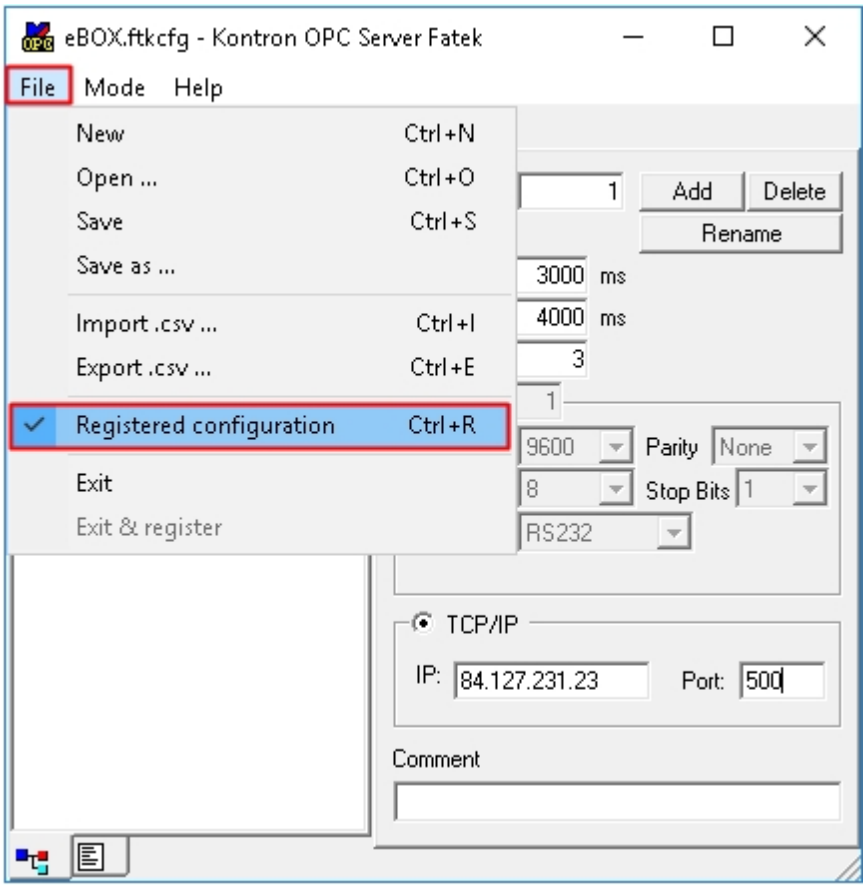
3. **Download and copy** the file ***Kontron_Fatek_OPC_Server_lic.exe*** to the location where the Kontron Fatek OPC Server was installed (typically C:\Program Files (x86) \KontronCzech\OPCServers\Fatek) and execute it with administrator permissions**,** *and then press "**Yes, I accept**".*

Execute the **KONTRON FATEK** application with administrator permission and open the file **[eBOX.ftkcfg]** from the **IoT\eBOX** folder. Once you have opened this file:

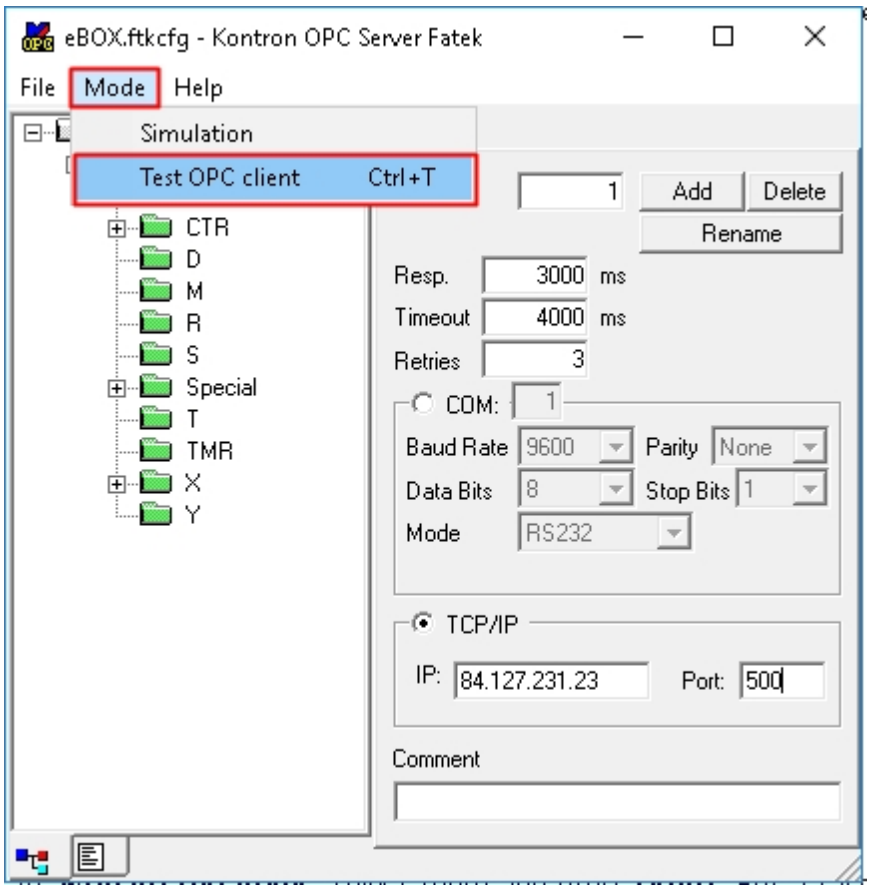
1. Enter in the [IP] field the IP that you defined in the Step 2: Ethernet configuration.



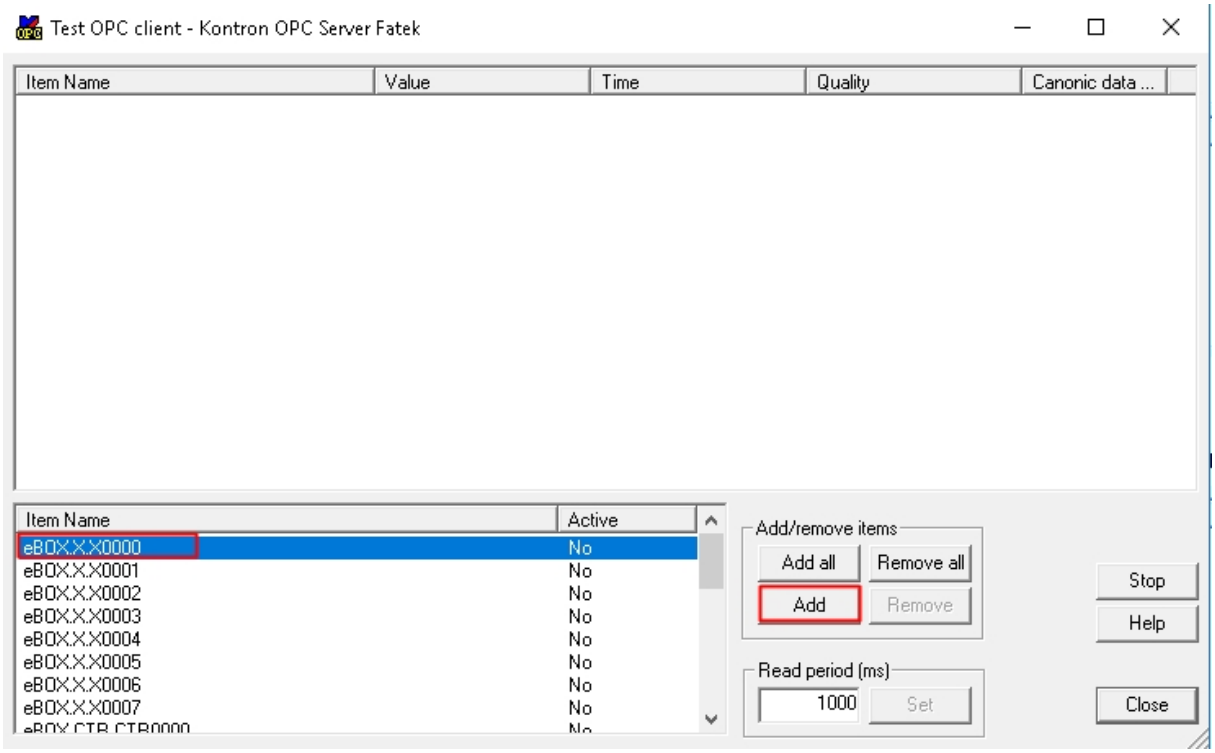
2. Register the configuration (this step will not work unless you executed the application with administrator permission):



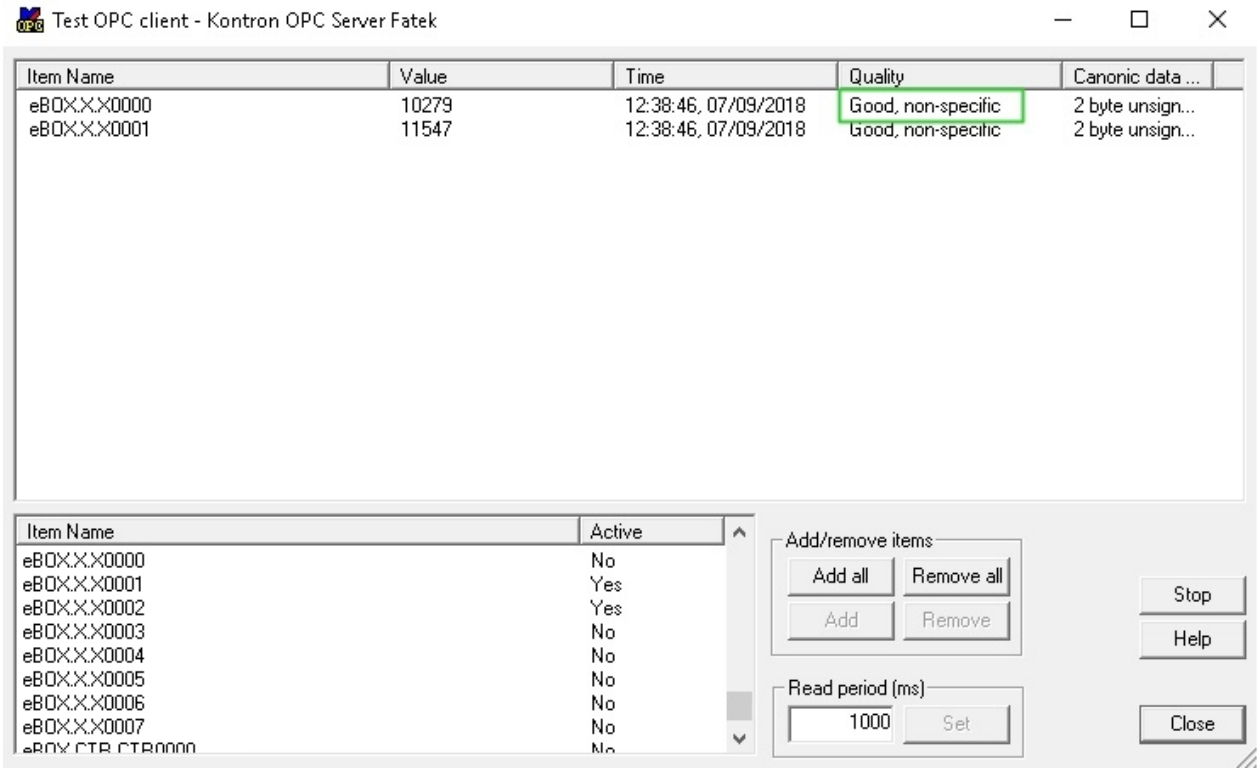
3. To know the name of the Items that, later, you will have to insert in the Configuration Wizard press **[Mode]** and click **[Test OPC client]** and the following window will open:



4. To **check the items,** select them and press **[Add].** For a faster activation, press **[Add all]**



5. Check that in the **Quality** column "***Good***" appears, to ensure that everything works correctly, otherwise, check that all connections are connected correctly.



Need help?

Step 3. Install & configure OPC Bridge

NOTE: If you choose full service with an eHUB, you do not have to perform the following steps.

1. **Download** the **edinn OPC Bridge folder** from the edinn Academy, **IoT area** here

2. **Unzip the file just downloaded and execute**, from the **edinn_OPC_Bridge_Setup folder**, the file **setup.exe** and follow the steps to install.

3. Once installation is finished, a pop-up window will show up to proceed with the OPC Bridge configuration. If this window would not show up, you can execute, with **administrator permissions,** the program **C:\Program Files (x86) \edinn\edinnM2\OPCBridge\edinnM2_OPCBridge_Configurator.exe** and follow next steps:

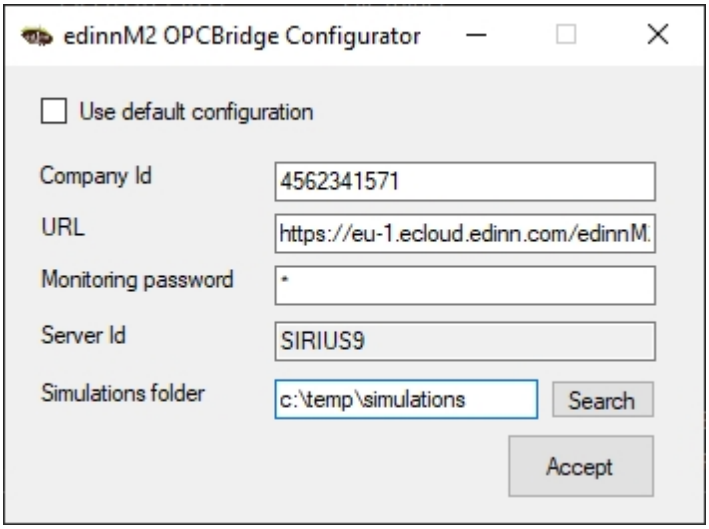
   A) Type the internal company Id of the company you created in Step 1 (a 10-digit number written on the e-mail you will receive after creating your company).

   B) Type the URL based on the server where you created your company. For example, if

you created your company in this server **eu-1.ecloud.edinn.com**, you should type: **https://eu-1.ecloud.edinn.com/edinnM2/monitor/?**

C) Type the password that you typed when you created the company in Step 1, which has also become the monitoring password, which can be different to your user password and can be changed in the monitoring step of the configuration wizard.

It is not necessary to type anything in the "Simulations Folder" field. Accept the configuration.



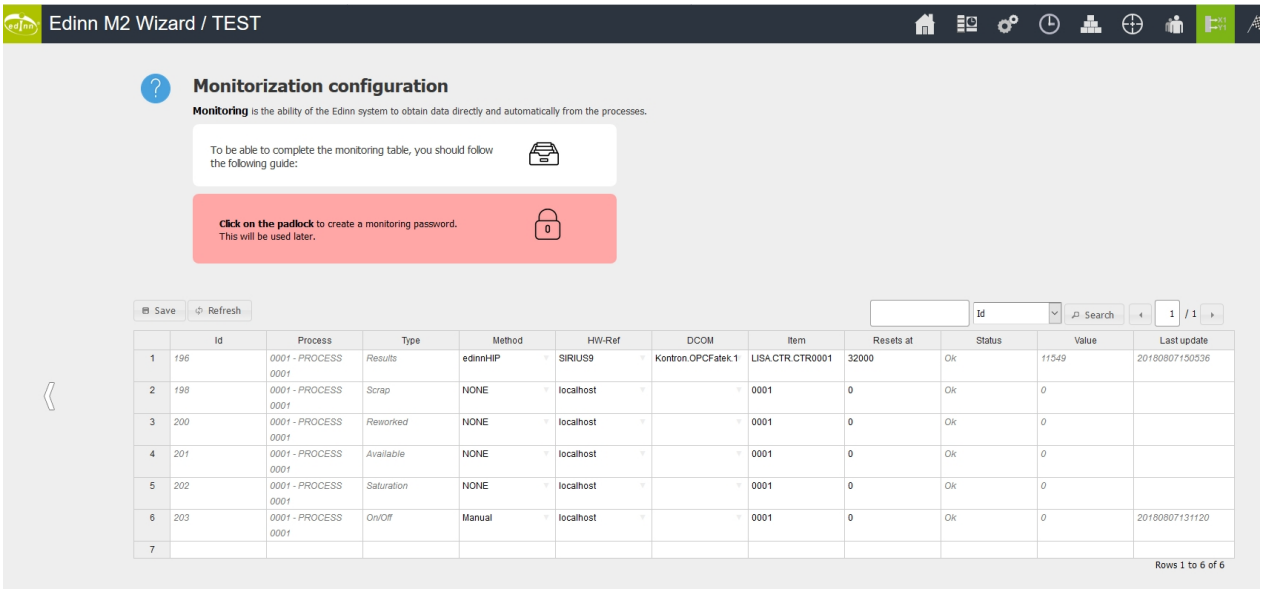
Need help?

Step 4. Connect and start

> **NOTE:** If you choose full service with an eHUB, you do not have to perform the following steps.

1. Go back to the configuration wizard, step of Monitoring Configuration.
2. Choose method edinnHIP for all the signals that you have connected. Please notice that only certain signals types are available from this wizard. To use all the signals types you have to access the server directly or request help to edinn.
3. Click on HW-Ref to see the computers where OPC Bridges are running and connected to this company. If HW-Ref is not showing the name of the computer where you installed the OPC Bridge, click the Refresh button and try again. If the problem persists, please contact edinn.
4. Click on DCOM and select the OPC Server from the list of OPC Servers found by the OPC Bridge.

5. Type the name of the item (also known as Tag) that you want to read.
6. Type the count on which that counter resets.
7. Click on the Save and Refresh buttons. Repeat after a few seconds until the Value and the Last Update show the actual value of the item. Once you see the actual value, the OPC Bridge is connected to the company. Please find below an example of the final result:



Configuring the eBOX transmission LED

Monitoring eBOXes come with 2 LEDs: one indicating power on, and the other indicating that data is entering correcly into an edinn server.

Verify that the **two LEDs** on the **eBOX case** (power on and transmission) have been turned on.

The communication LED (on the right) not only turns on by connecting the eBOX (PLC) to the data network, it also verifies that the monitored data is actually entering the server of the edinn platform and is being stored correctly.

If they **do not turn on**, please check:

- After connecting the RJ45 cable, it can take up to 1 minute for the light on the right to turn ON.
- eBOX is correctly powered on.
- There is internet connection on the RJ45 cable.
- In certain models, the TCP/IP port must be opened in your router and redirected to the eBOX internal IP address.

If still does not connect, please contact edinn.

Need help?

Using Other OPC Server

NOTE: If you choose full service with an eHUB, you do not have to perform the following steps.

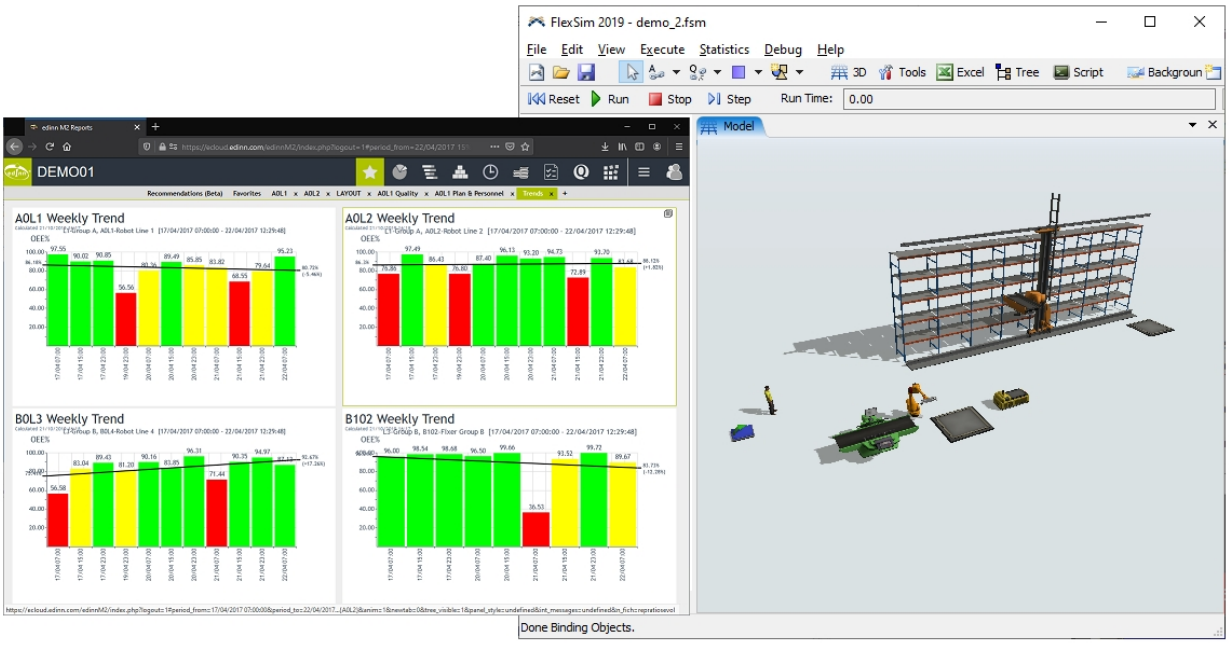
An edinn server can obtain data directly from any OPC Server, including the latest Universal Architecture (UA) OPC Servers.

Need help?

Using simulation

The edinn platform is capable of receiving data from a simulation exactly as if it was the real thing. Simulations can be run with third party softwares or by your own application. In this manual we will use FlexSim simulation software.

Need help?



Introduction

In this option of installation we will use these components:

1. edinn OPC Bridge installed on your local computer. We are using the one from version 2019-11 of the edinn platform.
2. FlexSim installed on your local computer. These procedure has been done with the free 19.2.2 version, although it should work correctly with higher versions.

3. An edinn organization created wether in the cloud or locally.
4. A FlexSim project file prepared to generate data which can be read by the edinn OPC Bridge.

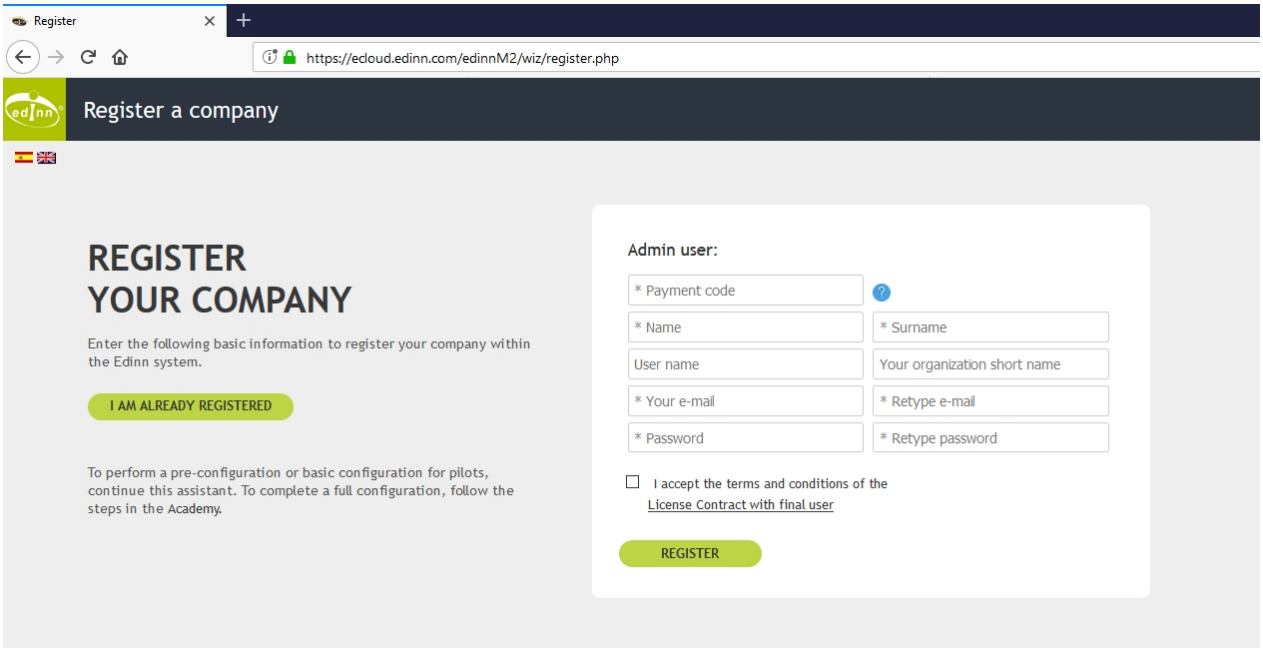
Step 1. Create & configure company

From an internet browser, visit:

- If you are a **Community** user, you will create your company in your local computer: http://127.0.0.1/edinnM2/wiz/register.php?lang=en
- If you have a **payment code**, you will create your company in the cloud: https://ecloud.edinn.com/edinnM2/wiz/register.php?lang=en

You can obtain your payment code here: https://edinn.com/online/?lang=en

In both cases, you should see a page like this:



In the next window, select:

- The server where you want to create the company. We recommend you to select the server which is closest to your users.
- The template which you want to use: in this case, select **Simulation 4 processes**.

Then follow the steps of the wizard **until the monitoring step**. Then continue on the next step of this procedure.

Need help?

Step 2. Install & configure OPC Bridge

1. **Download** the **edinn OPC Bridge folder** from the edinn Academy, **IoT area** here
2. **Unzip the file just downloaded and execute**, from the **edinn_OPC_Bridge_Setup folder**, the file **setup.exe** and follow the steps to install.
3. Once installation is finished, a pop-up window will show up to proceed with the OPC Bridge configuration. If this window would not show up, you can execute, with **administrator permissions,** the program **C:\Program Files (x86) \edinn\edinnM2\OPCBridge\edinnM2_OPCBridge_Configurator.exe** and follow next steps:

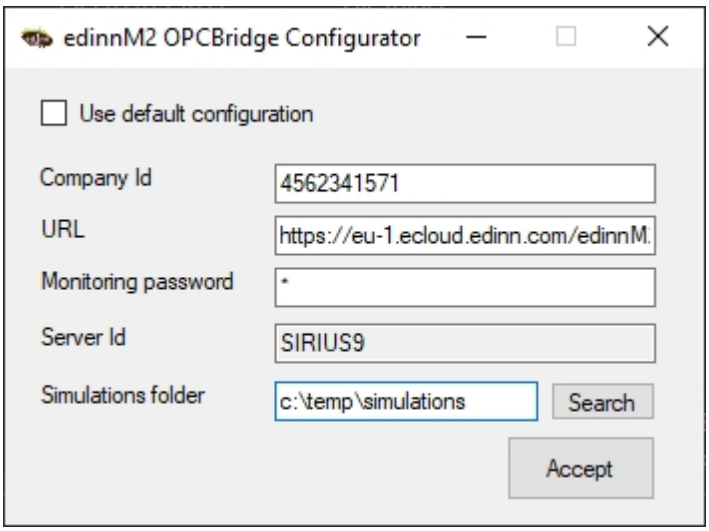
   A) Type the internal company Id of the company you created in Step 1: a 10-digit number that was shown to you in the web page after creating the company and written on the e-mail you will receive.

   B) Type the URL based on the server where you created your company. For example, if you created your company in this server **eu-1.ecloud.edinn.com**, you should type: **https://eu-1.ecloud.edinn.com/edinnM2/monitor/?**

   C) Type the user password that you typed when you created the company in Step 1, which has also become the monitoring password. Please remember that the user and monitoring passwords can be different. The monitoring password can be changed from the monitoring step of the web configuration wizard and from the console.

   D) In the field "Simulations folder" type, or search with the button, the complete path to where the FlexSim file **Simulation_4_processes.fsm** is, the one that you downloaded and unzipped in the previous 1 and 2 steps respectively.

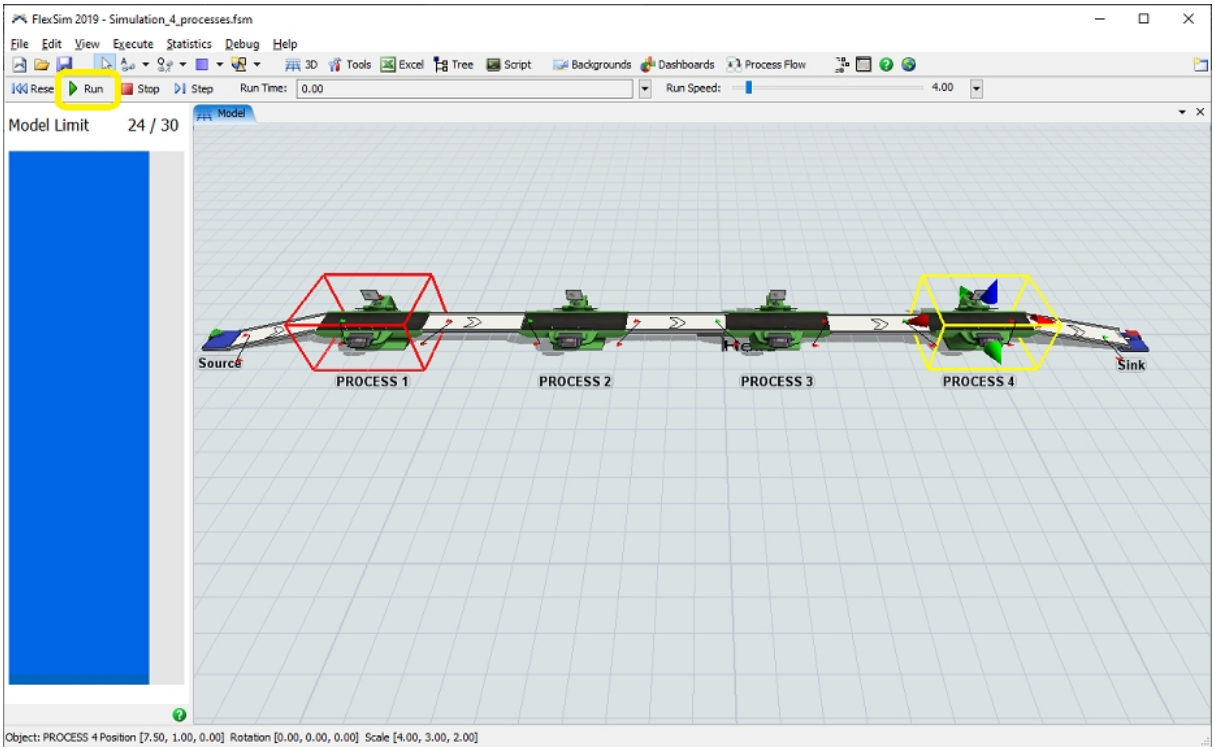
   Accept the configuration.



Need help?

Step 3. Simulate and start

1. **Download and install** the 19.2.2 (or higher and can be the free one) version from **FlexSim**.
2. **Double click** to open the file **Simulation_4_processes.fsm**, downloaded in the previous step. **FlexSim** should automatically open the file.
3. Run the FlexSim simulation by clicking the Run button as indicated in the image:



4. While the simulation is running, it will generate data which will be sent to your edinn server. Access the edinn web reports of your company to analyze the simulated processes and configure or operate with them through the operation edinn terminal.

Simulating failures

You can simulate failures by double clicking on the process and changing the **Process Time** to 999999. To restore the failure back to normal operation, you have to put the **original Process Time** and click the **Reset** button which is on the left of the Run button.

Need help?

New OPC Server item

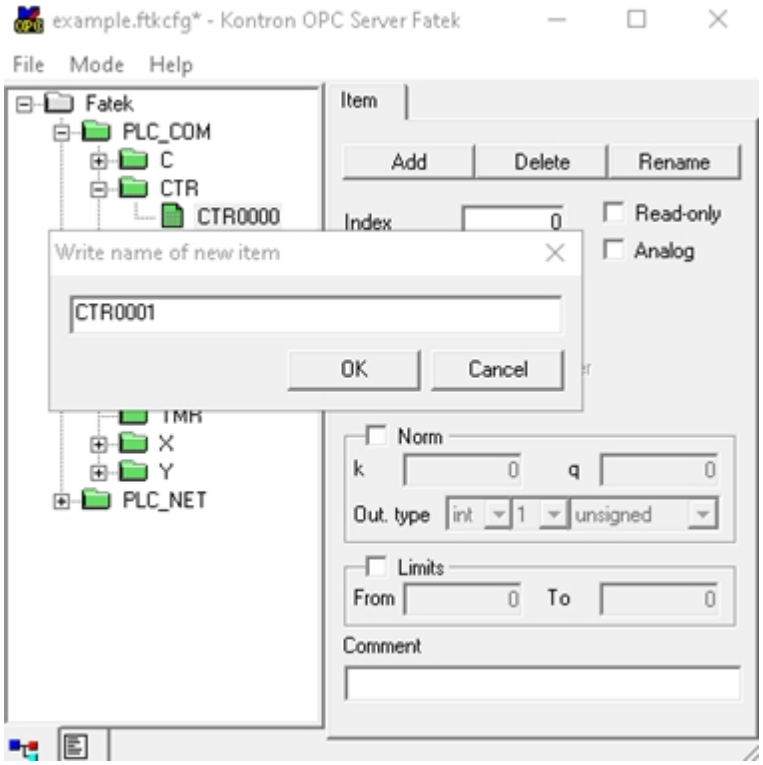
In this document we will see how to create a new item in an OPC Server software and project. As an example, we will create a counter item inside the Kontron OPC Server for Fatek.

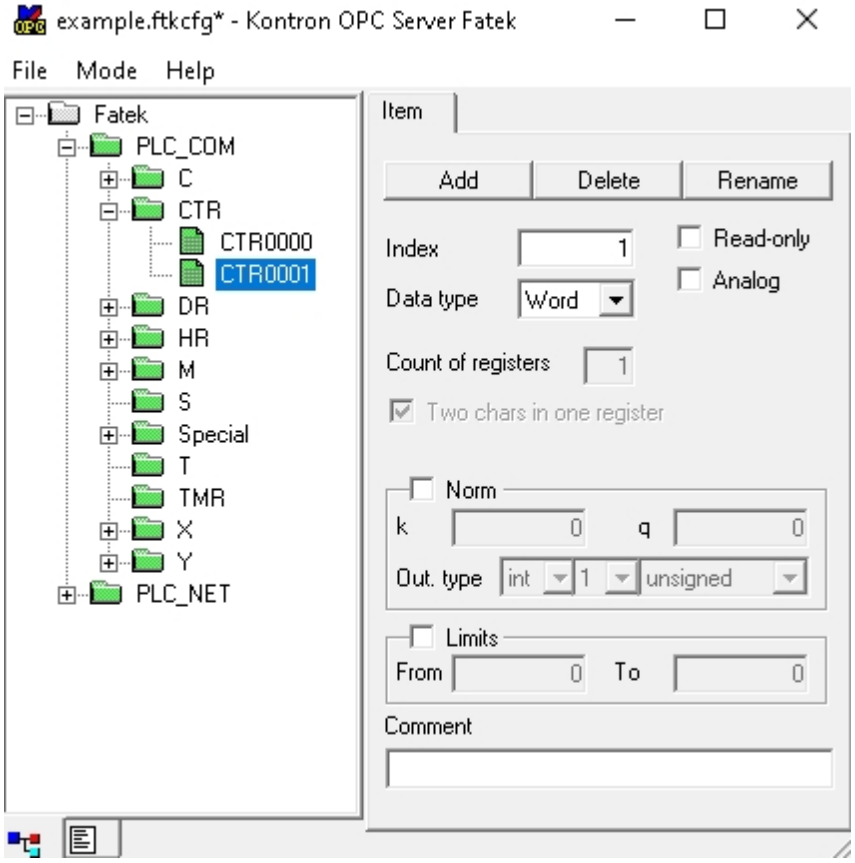
1. Execute the **KONTRON FATEK** application with administrator permission and open the **".ftkcfg"** file that you need to add the new item to.

2. Once you have opened the previous file, open the edinn M2 Console application, also with administrator permission, and close the edinnM2 Server's services in the right order (beginning from the Supervisor process, bottom to top).

3. Under the eBOX or PLC that you want to add the counter to, search in the list for the folder items and select the one desired. In this example, CTR, for counters.

4. Right mouse click on the folder, select **New item**, name it with respect to the names of the previous entries (in this example CTRXXX) and type its index with respect to the Plc item in the Ladder program, a counter in this case. For example, item CTR0001 must have an index of 1, CTR0002 an index of 2, and so on.





5. Save the file.
6. Close and re-open the **KONTRON FATEK** application to ensure that the newly configured counter has been added correctly.
7. Start the services in the edinn M2 Server in the correct order, from top to bottom, finishing on the Supervisor service.

Final verifications

Verify, before finishing the project, that the following points have been covered:

- **PROJECT CLOSE:**
  - Have photographs been taken of you and the users, with written permission from the users, to be stored in the project folder?
  - Has the quality and satisfaction department been informed that the project is finished and that the quality and satisfaction survey can be sent to the users?

- **LEADERS OF THE EDINN PLATFORM:**
  - Have the leaders (recommended to be more than one person) of the edinn system been appointed?
  - Have been informed so that all the support they need is channeled through the ticketing support system here?

    - Have they checked if the ratios and data offered by the edinn platform are correct?
    - Do they have the dashboard they need and it is being correctly and automatically updated?
    - Do they receive, by a weekly email, and understand, the Diagnosys Report?
    - Are the leaders autonomous and satisfied with the edinn platform?

- **AUTONOMY:**
    - Have users received enough training to be truly autonomous?
        - For operators, to operate.
        - For the project leaders, to configure.
        - For supervisors, to supervise: reports and scorecards.
        - For developers, to develop (optional: UDL and API).
    - Has it been verified if the users are really autonomous?
    - Do the users know how to request support and how to view this guide?

- **MOTIVATION AND IMPROVEMENT:**
    - Are the web dashboards available from outside the company?
    - Does the terminal show the dashboard visualization correctly?
    - Do the web dashboards automatically refresh? That is, have you verified that the caching of the reports is working correctly?
    - Does the client know the possibility of seeing customized dashboards on the devices (terminals), on web and on large format TVs in continuous carrousel?
    - Have you verified that the leaders and edinn receive by email the report Registry\Diagnosis in PDF format, at least monthly, in order to know if the edinn system is being correctly used and that performance (OEE) is improving?
    - Have you configured the devices so that on the terminals, after a little while without using them in the login screen, the screen saver activates showing dashboards which are useful for the users?

- **INFRASTRUCTURE (if the server is the user's responsibility):**
    - Is an automatic backup scheduled?
    - Does the client know the location of the backup for its management?
    - Has it been checked if the edinn system sends emails?
    - Has remote access (ISL) been verified to the edinn server and terminals for future remote support?
    - Has the web configuration wizard been disactivated?

> **Note:** Remember to check the critical factors for the success of the installation.

Satisfied?

Please suggest users to complete the satisfaction survey here.

Need help? 

Edinn support

If you have an active support contract, use the ticketing system to report, control and resolve your requests. The basic concepts for using the system are explained in the following guide.

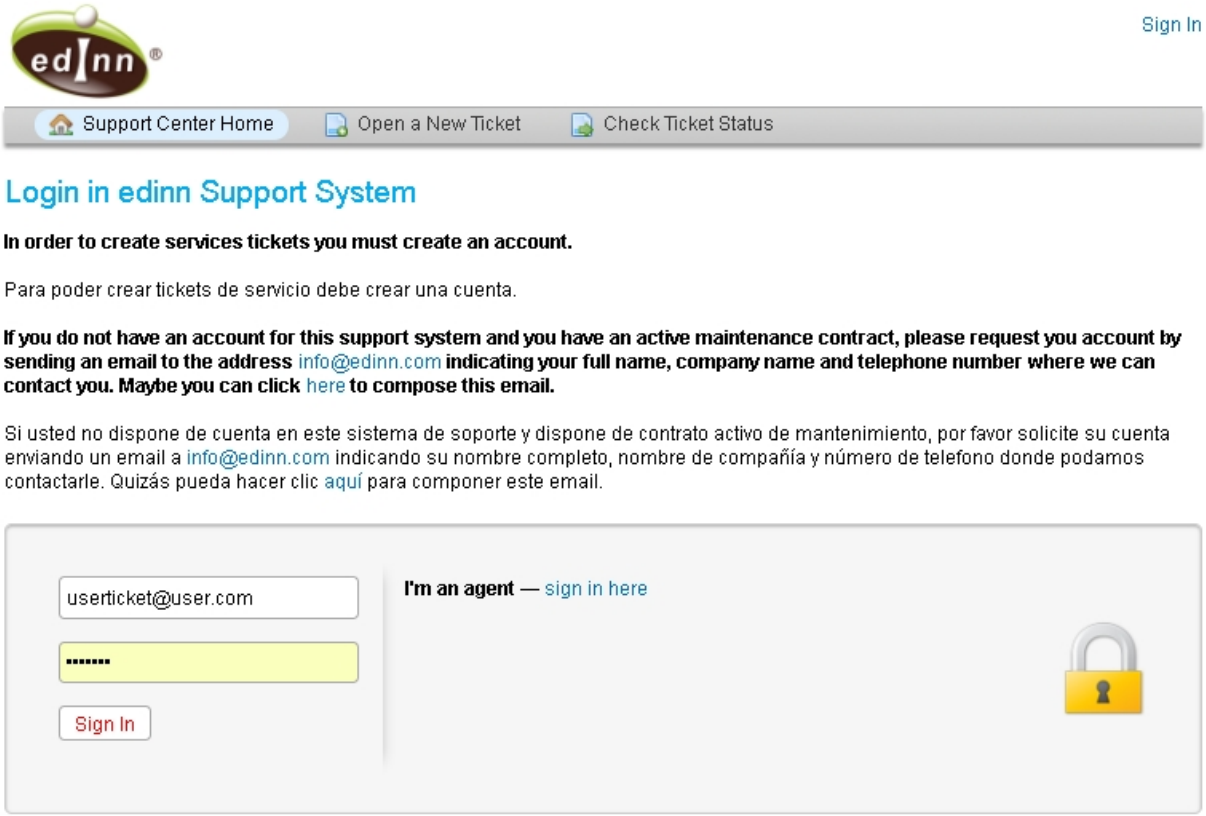
If you do not have an active support contract, you can use the Community Support Forum.

Please, additionally, check the Terms and Conditions in the sections which affect the edinn support.

Login edinn Support System

To login the edinn Support System please enter: https://ecloud.edinn.com/support/

Sign in with the user and password in the following window:

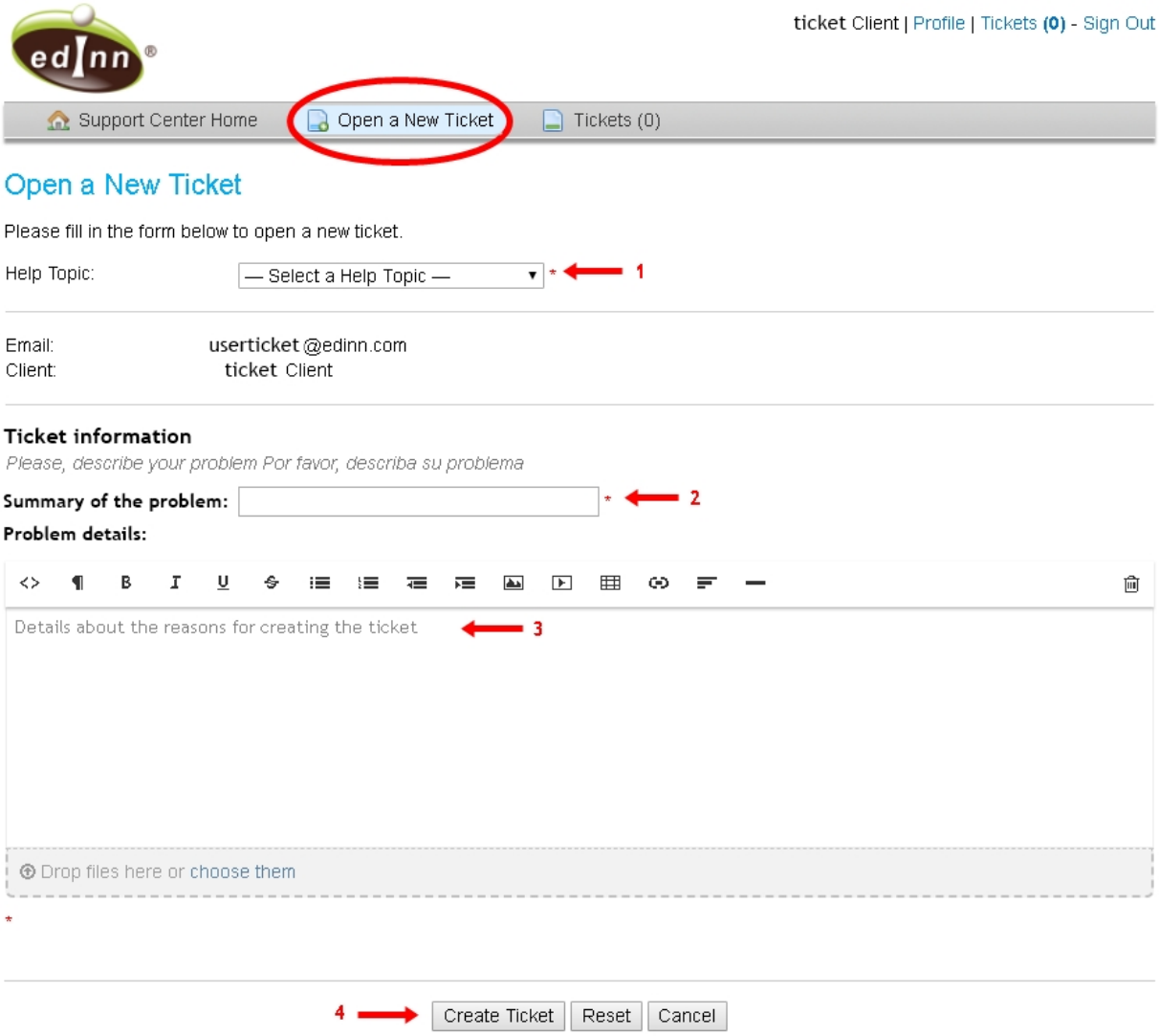


By default, edinn creates a ticket user with the edinn installation.

Need help? or in this case, send an email to support@edinn.com

Create a ticket from ticketing

To create a ticket, enter the "*Open a New Ticket*" window, the following has to be detailed:



1. Select a "Help Topic" depending on the incidence severity, choose between:
    - Can not work.
    - Urgent.
    - Difficult work.
    - Improvement.
    - General inquiry.

2. Describe the incidence into "Summary of the problem"
3. Give further information about the problem into "Problem details", you can also insert photos, videos, tables, links...

> **NOTE:** The more details are given, the better the communication between the client and edinn will be for detecting earlier the problem and solve it.

4. To send the ticket, click on "Create ticket"

Create a ticket from the email

Alternatively, you can create a ticket sending an email to ticket@edinn.com

The subject will be saved as the "Summary of the problem" and the body of the email as the "Problem details" . The "Help Topic" which defines the severity of the incidence, as default will be defined as Urgent, this can be changed from the Ticketing platform: https://ecloud.edinn.com/support/tickets.php

> **NOTE:** The more details are given, the better the communication between the client and edinn will be for detecting earlier the problem and solve it.

Copyright

Terms and Conditions

Last modification: 06/07/2021 (version 51)

These Terms and Conditions refer to the use of the edinn platform. Please also review and accept the Terms and Conditions (available in this link) referring to the commercialization, installation and maintenance of the edinn platform.

Thank you for using edinn! This Use License and Terms and Conditions Contract (the "Terms"), govern your access to and use of the **edinn** (edinn is the abbreviation of the incorporated company EDINN GLOBAL, S.L., with Spanish fiscal ID B-97423305, indicated here as "edinn", "we" or "our") software, hardware, websites and services (the "Products and Services"), so please read them carefully before using the Products and Services.

By using the Products and Services you agree to be bound by these Terms. If you are using the Products and Services on behalf of an organization, you are agreeing to these Terms for that organization and assuming that you have the authority to bind that organization to these terms. In that case, "you" and "your" will refer to that organization.

You may use the Products and Services only in compliance with these Terms. You may use the Products and Services only if you have the power to form a contract with edinn and are not barred under any applicable laws from doing so. The Products and Services may continue to change over time as we refine and add more features to them. We may stop, suspend, or modify the Products and Services at any time without prior notice to you. We may also remove any content or feature from our Products and Services at our discretion.

ACCOUNT SECURITY

You are responsible for safeguarding the password that you use to access the Products and Services and you agree not to disclose your password to any third party. You are responsible for any activity using your account, whether you authorized that activity or not. You should immediately notify edinn of any unauthorized use of your account. You acknowledge that if you wish to protect your transmission of data or files to edinn, it is your responsibility to use a secure encrypted connection to communicate with the Products and Services.

SOFTWARE AND UPDATES

Some use of our Service requires you to download a client software package ("Software"). Edinn hereby grants you a limited, nonexclusive, nontransferable, revocable license to use the Software, solely to access the Products and Services. Your license to use the Software is automatically revoked if you violate these Terms in a manner that implicates our intellectual property rights. We hereby reserve all rights not expressly granted in these Terms. You must not reverse engineer or decompile the Software, nor attempt to do so, nor assist and nor allow anyone else to do so. Our Products and Services may update in their on-line version, or automatically in the Software on your device, when a new version is available. Our prices and conditions may also change without prior notice. If for any reason your devices do not update automatically, and you keep using an older version of our Products and Services

which is not supported by edinn anymore, we may terminate this agreement and stop your Products and Services without prior notice.

EDINN PROPERTY AND FEEDBACK

These terms do not grant you any right, title, or interest in the Products and Services, Software, or the content in the Products and Services. While we appreciate it when users send us feedback, please be aware that we may use any feedback, comments, or suggestions you send us or post in our forums without any obligation to you. The Software and other technology we use to provide the Products and Services are protected by copyright, trademark, and other laws of both Spain and foreign countries. These Terms do not grant you any rights to use the edinn trademarks, logos, domain names, or other brand features.

COPYRIGHT

Edinn respects others' intellectual property and asks that you do too. We will respond to notices of alleged copyright infringement if they comply with the law and are properly provided to us. We reserve the right to delete or disable content alleged to be infringing and to terminate the access to the Products and Services of the infringers.

The Products and Services may contain links to third-party websites or resources. Edinn does not endorse and is not responsible or liable for their availability, accuracy, the related content, products, or Products and Services. You are solely responsible for your use of any such websites or resources. Also, if we provide you with any software under an open source license, there may be provisions in those licenses that expressly conflict with these Terms, in which case the open source provisions will apply.

YOUR STUFF AND YOUR PRIVACY

By using our Products and Services you provide us with data, information, files, and folders that you submit to edinn (together, "your stuff"). You retain full ownership to your stuff. We don't claim any ownership to any of it. These Terms do not grant us any rights to your stuff or intellectual property except for the limited rights that are needed to run the Products and Services, as explained below.

We will not share any of your information with any other third party, unless we have your authorization to do so, which is not granted by this agreement. But, by using our Products and Services, you accept that we may inform publicly that your organization is a customer of edinn. This authorization only includes naming the customer.

You are solely responsible for your conduct, the content of your files and folders, and your communications with others while using the Products and Services. For example, it is your responsibility to ensure that you have the rights or permission needed to comply with these Terms.

We may choose to review public content for compliance with our community guidelines, but you acknowledge that edinn has no obligation to monitor any information on the Products and Services. We are not responsible for the accuracy, completeness, appropriateness, or

legality of files, user posts, or any other information you may be able to access using the Products and Services.

Finally, by using the edinn Products and Services you also accept that personnel from edinn might remotely connect to the server machine that hosts your stuff for surveillance, monitoring, maintenance or support purposes.

SHARING YOUR STUFF

The Products and Services may provide features that allow you to share your stuff with others or to make it public. There are many things that users may do with that stuff (for example, copy it, modify it, re-share it). Please consider carefully what you choose to share or make public. Edinn has no responsibility for that activity.

YOUR RESPONSIBILITIES

Data, files and other content in the Products and Services may be protected by intellectual property rights of others. Please do not copy, upload, download, or share files unless you have the right to do so. You, not edinn, will be fully responsible and liable for what you copy, share, upload, download or otherwise use while using the Products and Services. You must not upload spyware or any other malicious software to the Service.

You, and not edinn, are responsible for maintaining and protecting all your stuff. Edinn will not be liable for any loss or corruption of your stuff, or for any costs or expenses associated with backing up or restoring any of your stuff.

If your contact information, or other information related to your account, changes, you must notify us promptly and keep your information current.

TERMINATION

Although we prefer that you become another long-term user of the Product and Services, you can really stop using our Products and Services at any time. Additionally, we reserve the right to suspend or end the Products and Services at any time, with or without cause, and with or without prior notice. For example, we may suspend or terminate your use if you are not satisfying the required payments to edinn, or not complying with these Terms, or use the Products and Services in any way that would cause us legal liability or disrupt others' use of the Products and Services. If we suspend or terminate your use, we will try to let you know in advance and help you retrieve your data, though there may be some cases (for example, repeatedly or flagrantly violating these Terms, a court order, or danger to other users) where we may suspend your access immediately.

EDINN IS AVAILABLE "AS-IS"

Though we want to provide a great service, there are certain things about the service we cannot promise. Because of this: THE SERVICES AND SOFTWARE ARE PROVIDED "AS IS", AT YOUR OWN RISK, WITHOUT EXPRESS OR IMPLIED WARRANTY OR CONDITION OF ANY KIND. WE ALSO DISCLAIM ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. Edinn will have no responsibility for any harm to your

computer system, loss or corruption of data, or other harm that results from your access to or use of the Products and Services. Some states do not allow the types of disclaimers in this paragraph, so they may not apply to you.

Additional notes for installations of the edinn solutions:

1. If monitoring kits or user terminals have not been included because existing elements in the customer's facility will be used, in the case of these elements would not be powerful enough or they might need any adaptation, any additional cost regarding these concepts should be carried out by the Customer.

2. In general, the power and Ethernet sockets (among others), which are available to the edinn team and its partners, will be used. It is the obligation of the Customer to indicate if any socket, connector or element should not be used during the project. Any additional cost regarding this should be carried out by the Customer.

3. It is necessary that production and consumption electrical signals, which are to be monitored in this project, are already identified and electrically accessible to the edinn team and its partners when the project starts. Any additional cost regarding this should be carried out by the Customer.

4. In the case of using the computer data network (cable or WiFi) of the Customer, it is assumed that it works correctly and with the necessary quality level. Any additional cost regarding this should be carried out by the Customer.

5. If, under the customer's request, edinn or its partners would install a Wireless (Wi-Fi for example) computer network, this network will be guaranteed (min. 1 year) on its hardware, but it will not be guaranteed regarding the quality of the transmission of the data. Any additional cost regarding this should be carried out by the Customer.

6. When referring to "working day", we are referring to working days in Spain. You can ask for the updated calendar at any time.

7. Until the cost of the software use licenses has been paid to edinn, the customer will have a temporary software use license. When this temporary software use license will expire, the edinn system will automatically stop and the Customer accepts to totally delete and remove it from his facility and hardware in less than 7 days. The customer will be allowed to use the edinn platform again only after paying the edinn software use licenses.

8. Edinn provides services to install the platform at the Customer's facility, but that installation is finally only responsibility of the Customer, as he only can provide access to his facilities, access to his systems, and configuration data, among others.

9. The edinn platform might not be able to detect certain microstops when the cycle times are very short. This is due to causes which are external to the edinn system: slow computer data network, eBOX and/or PLCs response time, hardware delays, etc. Although this, the Customer will have 100% detail of where his losses are, as this

not detected microstops will be included in the "Low speed and microstops losses" category.

10. Anticipated payments are needed due to the costs of initiating and preparing projects, and where applicable, to the need of hardware and software licenses. For this reason, the anticipated payments cannot be returned to the Customer.

11. If the customer, or any of his providers, is providing computers or servers to run the edinn software, edinn will not be responsible in any case, of the malfunction, deterioration in performance or data, etc. of any application or function that might be before, during or after, on the equipment running the edinn software.

12. The installation, by edinn or by its partners, of the edinn platform does not include any software or hardware development nor adaptation, except those indicated in written in the commercial proposal.

13. All developments which have been approved by edinn to be included in the standard versions of edinn, will be implemented inside the last available version and will be guaranteed if the Customer has an active maintenance contract. In addition, because these developments will be integrated into the standard, and although edinn will try to respect them in the future with the purpose of affecting the least possible to the Customer, edinn reserves the right to change the standard product and therefore the results of these developments, even without prior notice.

14. In case of an integration with other systems, these must be adapted to Standard ISA95. If this is not the case, the hours of integration support must be requoted again.

15. All training performed by edinn is given to trainers, and not to final users, unless otherwise stated in written in the commercial proposal.

Additional notes for the maintenance of the edinn solutions:

1. Hardware is guaranteed for a period of 2 years. When a hardware element fails, the Customer accepts to send it, at his costs, to an edinn office. The hardware will be then verified, and if the hardware is inside the guarantee period and if edinn determines that its failure is not due to an incorrect use, negligence or natural disaster, then an identical or better equipment will be sent to the Customer, free of any charge.

2. Support will be provided remotely through the edinn support system ("ticketing") and in a working day. If the customer needs 24x7 support, it should be required and acquired to edinn. In edinn support services, on every interaction with the Customer, a minimum of 15 minutes will be consumed.

3. Additional support service hours should be acquired before the support service is provided to the Customer. However, the Customer could have support hours, only if this is clearly indicated in his licenses or maintenance invoices.

4. Edinn support service only includes the analysis and resolution of issues only in case they are a defect of the standard edinn software. Therefore, this service does not cover issues of any other type, even if they were generated by edinn, like, for example, but without being a complete list: related to the physical IT infrastructure, wrong data, regional differences like the use of coma or point to group numbers, neither related to tailored (nonstandard in the edinn platform) developments, like, for example, those using the API nor the UDL modules.

5. In the edinn support services, if the edinn engineers confirm that an issue is a defect of the edinn standard software, the solution will be implemented only on the latest available version of the edinn platform. Therefore, by this clause, the customer implicitly accepts to upgrade his edinn platform to the latest available version when required by the edinn engineers for the purpose of solving an issue. If the Customer rejects to upgrade his edinn version, then the issue could not be solved and edinn will be not liable of any responsibility.

6. The edinn support services will only be available when the client is up to date in the payments of the maintenances of all his edinn active licenses.

7. If the Customer needs from the edinn team an urgent response to a support request, then the Customer accepts to notify edinn in written with the word "URGENT" in the subject of the edinn support system. These urgent issues will consume minimum 60 minutes from the support time, if they are proved to be caused by causes external to the edinn standard software.

8. If the customer has acquired support hours, they cannot be used for new developments. In addition, not consumed support service hours will not be refundable, transferable nor accumulative between support periods.

9. The Customer will have a 5% discount on his maintenance contracts if he is allowing at least 10 visits to his facility to show edinn to potential customers of edinn or its partners (never competitors of the Customer), or if the Customer is presenting the edinn logo, with a link to https://edinn.com, on the home page of his web page or on his products, in a visible place.

10. In the edinn support services, the provision of new versions of the edinn platform, includes only the provision of the necessary files and the procedures to update them, but does not include any service for the update nor correction, for example the conversion, recovery process or adaptation of any tailored developments (those in the API or UDL modules, for example).

11. It cannot be guaranteed, by any mean, that a failure or the update of the edinn platform will not affect the Customer in any way, and therefore, under no circumstance edinn will be responsible for the damages or losses caused by the use or lack of use of the edinn platform. If the Customer wants to test how a new version or update will affect him, he is recommended to request for a separated test environment where he will be able to try to anticipate any possible bad effect.

12. Tailored and particular developments, for example, but without being a complete list, those developed through the API or UDL modules; must always be implemented and served to the customer in the latest published version of the edinn platform. In addition, these particular developments are not guaranteed by edinn in any way, as they are not distributed by edinn in the standard package.

13. The use of the edinn platform is only allowed to the Customer if he has valid edinn software use licenses and if the Customer has realized all the pending payments to edinn as indicated in the invoices' payment terms. These edinn software use licenses should be renewed, at least, yearly.

14. In the version's updates, although in some cases it is possible to restore and come back to previous versions, the user accepts that it cannot be guaranteed in all cases that it will be possible to go back to previous versions.

15. These software use licenses do not provide the Customer with any property right on the edinn platform, neither in total nor in part (software, hardware nor documentation).

16. The Customer accepts to not use any version of the edinn platform which is not officially supported by edinn. In this case, the customer accepts to update his platform to a supported version or to definite and totally stop using the platform and to definitely delete and remove the edinn platform from his facilities in less than 7 days after receiving written notification from edinn.

17. When the Customer lacks of a valid use license of the edinn platform, as the use licenses did not provide the customer with any right of property on the edinn platform or any of its parts (software, hardware and documentation), the customer accepts to totally and definitely delete and remove the edinn platform from his facilities in less than 7 days after receiving written notification from edinn.

18. With the purpose of maintaining the quality of the products and services, edinn reserves the right to change, at any time and without prior notice, the prices (also named "costs") of hardware, licenses, services, maintenances and supports.

19. By using the edinn platform (software, documentation and or hardware), the Customer accepts: 1) That edinn publishes, in any format (verbal, written, or other formats), that the Customer is using edinn, but taking into account that this authorization only includes the name and the logo of the Customer; and 2) That edinn, or third-party companies authorized in written by edinn, might use the data of the Customer as long as it is in an aggregated and anonymously manner, respecting the Spanish Personal Data Protection Law and the Customer's confidentiality; and with the only purpose of providing the functionality, new products and services or improving the current ones, adapt the products and services to the Customer's preferences and/or to avoid an illegal or not licensed use of the product.

20. Although the edinn platform executes automatic optimizations to improve performance, with the passing of time, the performance of the platform could worsen, mainly due to the existence of more data records and/or more users and/or processes. If this was the case, it will be the responsibility of the Customer to provide the means to improve the performance and under no circumstance will be the responsibility of edinn as it is impossible to forecast all the possible situations of workload that could arise.

21. The services provided by edinn are configured as an obligation of means and not of results, so the subscription of this Contract does not guarantee the client to obtain results.

22. The costs of the servers in the cloud could vary in each renewal and without prior notice.

23. Edinn will not be responsible, neither will be covered by its support service, any issue related with equipment.

24. Edinn will not be responsible of, neither will be covered by its support service, any issue related with items like for example, and not being an exhaustive list, computers or servers, which do not meet the minimum requirements indicated in edinn's terms and conditions, available in this address: http://edinn.com/termsandconditions

LIMITATION OF LIABILITY

TO THE FULLEST EXTENT PERMITTED BY LAW, IN NO EVENT WILL EDINN, ITS AFFILIATES, OFFICERS, EMPLOYEES, AGENTS, SUPPLIERS OR LICENSORS BE LIABLE FOR (A) ANY INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, EXEMPLARY OR CONSEQUENTIAL (INCLUDING LOSS OF USE, DATA, BUSINESS, OR PROFITS) DAMAGES, REGARDLESS OF LEGAL THEORY, WHETHER OR NOT edinn HAS BEEN WARNED OF THE POSSIBILITY OF SUCH DAMAGES, AND EVEN IF A REMEDY FAILS OF ITS ESSENTIAL PURPOSE; (B) AGGREGATE LIABILITY FOR ALL CLAIMS RELATING TO THE PRODUCTS AND SERVICES MORE THAN THE AMOUNTS PAID BY YOU TO EDINN FOR THE PAST THREE MONTHS OF THE PRODUCTS AND SERVICES IN QUESTION. Some countries do not allow the types of limitations in this paragraph, so they may not apply to you.

MODIFICATIONS

We may revise these Terms from time to time and the most current version will always be posted on our service. If a revision, in our sole discretion, is material we will notify you (for example requesting your acceptance on the service). Other changes may be posted to our blog or terms page, so please check those pages regularly. By continuing to access or use the Products and Services after revisions become effective, you agree to be bound by the revised Terms. If you do not agree to the new terms, please stop using the Products and Services.

MISCELLANEOUS LEGAL TERMS

THESE TERMS AND THE USE OF THE SERVICES AND SOFTWARE WILL BE GOVERNED BY THE SPANISH LAW EXCEPT FOR ITS CONFLICTS OF LAWS PRINCIPLES. ALL CLAIMS ARISING OUT OF OR RELATING TO THESE TERMS OR THE SERVICES OR SOFTWARE MUST BE LITIGATED EXCLUSIVELY IN THE COURTS OF VALENCIA, SPAIN, AND BOTH PARTIES CONSENT TO VENUE AND PERSONAL JURISDICTION THERE. These Terms constitute the entire and exclusive agreement between you and edinn with respect to the Products and Services, and supersede and replace any other agreements, terms and conditions applicable to the Products and Services. These Terms create no third-party beneficiary rights. Edinn's failure to enforce a provision is not a waiver of its right to do so later. If a provision is found unenforceable the remaining provisions of the Agreement will remain in full effect and an enforceable term will be substituted reflecting our intent as closely as possible. You may not assign any of your rights in these Terms, and any such attempt is void, but edinn may assign its rights to any of its affiliates or subsidiaries, or to any successor in interest of any business associated with the Products and Services. Edinn and you are not legal partners or agents; instead, our relationship is that of independent contractors.

IF YOU PROCEED TO USE THESE PRODUCTS AND SERVICES, THEN YOU ARE ELECTRONICALLY SIGNING THIS CONTRACT AND THEREFORE YOU FULLY AND EXPLICITLY ACCEPT IT.

Revision notes

WARNING: Changes marked in yellow indicate **IMPORTANT actions that must be carried out by you or the administrator of the edinn system** during the upgrade process to avoid issues.

2024-07:

Build	Subsystem	D
673	User's Guide	New version 2024-07 is generated as readiness for integration with AI tools User's Guide.
673	Terminal	Improvement in the efficiency of starting work orders that have an associat
673	Terminal	Corrected issue: now, when a work order advances its start to collect orpha the order, it is removed.
674	Server	The supervisor now periodically checks the API and may restart edinnM2Repo
674	Terminal	Issue fixed: when a resource was inside a process belonging to an area, it w
674	API	Improved API call to obtain current status of processes: now only processes

675	API	Fixed a bug in the Hourly Dashboard that caused production data to be miss
676	Server	An issue with the path has been fixed. When the option to view planned tim
677	Reports	An error in the autocontrol configuration report has been fixed where tasks
677	Server	Improved the optimizer so that it can run without crashing, despite the am
677	Reports	Improved status evolution report to allow adding to favorites and caching.
677	Console	Fixed an issue in the database backup process: if no destination folder is sp
677	Terminal	Warehouse class fields are now taken into account when requesting stock.
678	Help	Explains better the behaviour of the Speed0 tags.
678	Terminal	Fixed issue in Schedule Edit form. The I/O records belonging to the selected
678	Help	Help for the following API call has been updated: processes/kpis/followup
678	Integration	In the I/O integration the fields were not being sent to the ERP.
679	Integration	Fixed issue in ERP stock query and subsequent management.
679	Reports	Improvements to the activity report. Now, the activity time always appears status and the statuses are summarized for each resource and at the end o
679	Integration	Fixed issue in ERP prduction performance cancelation message.
680	Reports	Fixes a bug in the dashboards when calculating the activity indicator.
680	Help	Productive Efficiency ratio has been explained better.
680	Terminal	The dashboard now saves selected indicators so that the selection is mainta
681	Server	Now, when there is a new automatic time usage, the previous time usage i
682	Server	The periodic OEE calculation excludes archived areas and areas whose subar
682	Server	The option to save the status evolution report has been added.
682	Server	Now autocalendar schedules statuses only for the next 720 days.
682	Server	Fixes a bug in the Total Time calculation for processes where an operation t
682	Terminal	Allows you to display OEE, EP and Activity indicators on both the dashboard
682	Terminal	Now, the terminal writes into server log as high detail messages

683	Integration	Fixes an issue with fields creation.
683	Reports	Resolves an issue in the results report that did not differentiate between th
683	Terminal	Now, when a work order that has already been started is assigned, only the
683	Terminal	Do not ask for notifying production when active order has target equal to z
683	Terminal	Improves the time response of the check for the presence of the result in t
683	Terminal	Fixes a bug when changing the process status when starting an order.
684	Terminal	Fixes an issue with checking active orders.

2021-01:

Build	Subsystem	D
501	Terminal	New Area Administrator role, which allows to manage processes and their P
501	Terminal	Now, when you copy and paste a dashboard between users, if there are par copy of the rest of widgets.
501	All	Size of description fields of results, statuses, order reference, SPC program
501	Reports	Fixed error that creates an area loop. Now is not possible putting an area i
501	Console	Now, in the Console, there is a new button to manually clear the Logs.
501	Terminal	Now, in Schedule, you can also import a list of orders from the clipboard.
501	Terminal	Now you can change the descriptions of the default statuses, which is usefu
501	Server	Process responsible will be no more notified of auto changes and OnOff char
501	Terminal	Added Image property to edinn Button to put a image instead of an Icon. N
501	Terminal	Now processes can show images.
501	Reports	Fields option added to Schedule FollowUp report. If you choose a record of t the work order statuses will appear.
501	Reports	Now you can filter by order in the Status Status Report.
501	Terminal	New device configuration option, now you can choose a default period for e
502	Terminal	New "Remember me" option in the offices login, which allows fast log in wit

503	Terminal	Now administrators can also manage processes, areas, resources and other mode.
504	Terminal	Now if the process is related to a resource, the default starting status for a configuration.
504	Terminal	Now you can access the work order URL folder from schedule edit.
504	Terminal	Now in the win32 terminal main window you can asign areas and processes
504	Terminal	New PSR Relation button. Now you can assign a status to multiple processes
504	Terminal	Now orders must be assigned to an area and can be assigned to any process
504	Server	Now the server can automatically perform its own database backup. By defa their own backup of the database in c:\Program Files (x86)\edinn\edinnM2\s
505	Server	Fixes a bug that was causing some UTF8 characters to be changed in the la
505	Server	Fixes a bug that was preventing to show up the alerts of delayed orders.
506	Server	Fixed that prefinished orders were taken into account for the users alerts,
506	Terminal	Now the Planned Operation Time can be indicated in the process configurat
506	Terminal	Now in the main window, when showing current status of areas, colours of which affects that specific area; otherwise, are shown as unscheduled as th
507	Terminal	Now you can change the priority (WBS) of multiple tasks at the same time.
508	Terminal	Now you can change the resolution of the win32 terminal by maximizing the
508	Terminal	Fixes a bug that caused the Console to not copy the fields of the PER relatio
510	Server	Now orders are only archived to historic data when they are finished or can
510	Terminal	Now, in the orders, the description field is multiline, and you can see it fro
510	B2MML	Prevents the creation XML messages not well formed and write the error in
510	B2MML	Fixed issue #518872. When an order is deployed to several processes the sys
512	Terminal	Now when you reschedule a work order, or change its status to prefinish or
515	Server	New Work Management module. Servers might need relicensing.
515	Terminal	Now statuses can be configured to require a mandatory comment.

515	Terminal	2 more device themes, Light bright and Dark bright, have been added. Now
516	Terminal	The speed graphic has been corrected so that it is reescaled to +10% of targ
516	Terminal	Corrects a problem which affected to the internal time of the terminal whe
517	Terminal	Prevents the Pop up messages to open multiple windows, to avoid confusing
518	Terminal	New Scheduled configuration option for Areas.
518	Terminal	Now the limit date and time of the work orders is automatically updated wh
519	Terminal	Now a status can require an update of the progress of results or orders.
519	Server	Now the monitor can calculate the time when results should be generated c current working order limit date.
520	Terminal	Now you can change (move) the order of the statuses while you are selectin
520	Reports	Now the Quality/Summary report can be displayed as a graphic.
521	Terminal	New support for recurring orders and auto calculation of their deadline.
523	Terminal	Bug fixes, improvements and better colours.
528	Server	The behaviour of the exceptions has been improved.
529	Terminal	Bug fixes (In&Out window, duplicate entries when creating API sessions, and
530	Reports	New Operation report, useful for processes which are persons.
531	Terminal	Now user cannot login if the version and the build do not match with those case. When it launches the automatic update, it copies to the clipboard the
531	Terminal	Now users can manage autocontrol tasks which are shared by multiple proce
531	Terminal	Big data has been improved so that calculations are done much faster. Serv
532	Terminal	The size of the URL fields has been increased up to 16 kilo bytes to be able
535	Terminal	Bug fixes and improvements.
536	Terminal	Now emails are sent to users when certain scheduling events occurr and also
537	Terminal	Now there is a button + on the main window if you are admin. Allows you to
537	Terminal	Now the gantt shows in grey days where all processes are not working.

538	Terminal/Reports	The path graphic has been corrected to adjust to the "path" graphic especif production, what is higher, at the last horizontal line which contains the ma
539	Terminal	Tooltips have been improved and now can be activated/disactivated from th
540	B2MML	Added a validation for Location tags. Now, if the ERP sends a Equipment El
540	Terminal	Now multiple comments can be associated with records.
540	Terminal	Now the user can select which columns wants to see in the scheduling tables
540	Terminal	Added support for mouse wheel scroll in the the tables of scheduling and in
541	Terminal	Fixes a bug that caused that when the terminal was forced to update from current configure server for updates.
541	Terminal	New contextual menu on the main window allows to move areas to other ar
541	Terminal	Now future statuses scheduled manually by users in the calendar or in the st
541	Terminal	Fixes a bug in the speed graphic.
541	Terminal	Now you can navigate to all the records related to another record.
541	Terminal	Now users can insert future statuses in the planned calendar.
542	Server	Now the system warns administrators when the configuration of statuses ge
542	Terminal	The ? button of all windows has been converted to a menu button as it now
543	Terminal	Fixes a bug that was showing the wrong web page when the automatic upd
543	Terminal	Now administrators and administrators of areas can assign roles using the n
543	Terminal	Fixes a problem when inserting bad results.
543	Server	Fixes a problem that affected reports and other server tasks when analyzing
544	Terminal	Fixed a problem that did not allow to justify certain statuses indicating inco
544	Server	Fixed a problem when opening sessions (API, Terminal and Reports) that sho
544	Terminal	Fixed a problem when assigning devices to areas from the mouse right butt
545	Server	The monitor will not warn to resources associated with processes of next st
545	Terminal	New recurring results which are the only applicable for recurring tasks or wo

545	Terminal	Now the main window, and therefore all the others, can be resized without
546	Server	Fixes a bug by which backups were not automatically generated by the Opti
546	Reports	Fixes a bug in the report Analysis\FTE.
547	Server	Now the Master Port is restarted if it fails.
548	Terminal	Fixes a bug by which some searches did not work in the win32 terminal Loc previous search.
548	Terminal	Now tasks/work orders can only be started or stopped from the view sched started in the same process.
549	Terminal	Fixes a bug when moving areas to areas, from the main window with the ri
549	Terminal	Now if the user only sees a process or an area, it will have to click it from t inhibited the user to see the global ratios of that element.
549	Reports	In the reports, the selection of areas and processes was lost when changing
550	Terminal	The functionality of assigning area roles to resources is now available when
551	Terminal	Fixes a bug when starting an order, the server option 'Ignore FAIs in manual
551	Terminal	Tooltips to guide new users have been improved in the win32 terminal.
551	Reports	Fixes a bug by which some widgets were showing data from other widgets.
551	Terminal	Fixes a bug by which default values of fields were not being loaded into task
552	Server	Some bugs have been corrected and performance of Big Data has been impr
553	Terminal	Fixes a bug that was causing the path of the main window of the terminal t
553	Server	Fixes a bug that was launching autocontrol tasks despite they were marked
554	Reports	Fixed issue #741956. There was an error which caused slight differences of
554	Reports	Fixed issue in report Status\Status and Registry\Not recorded: Order does no
554	Reports	Improved the load of work orders in report Statuses\Statuses that was some
555	Terminal	Fixes a bug where I/O heritage was not working correctly for new results.
555	Reports	Now favorite reports inform of when each report was last updated and the resources.

556	Terminal	Now you can assign the active order to a production record which was creat
556	Terminal	Fixes a bug by which some results did not appear in the PSR Relations confi
557	Server	Fixes 2 bugs in Big Data: was showing errors when there were no changes i
557	Server	Fixes a bug in reports cache that was not caching certain favorite reports.
557	Terminal	Now results are not requested when justifying or inserting a status which is
558	Terminal	Now can require strong passwords when user is manually changing it from w
558	Terminal	Now the automatic scheduler considers calendar as all available when no cal
558	Terminal	Fixed bug: Autocontrol task does not generate production.
558	Server	New function InvokeAddOn to be used for any AddOn of the company.
558	Terminal	Speeds up entrance to the Results window.
558	Terminal	Fixes several little bugs and improves stability of automatic scheduling.
560	Reports	Fixes a bug that could hang certain reports.
560	Terminal	Now URLs can also open folders.
560	Terminal	Fixes a bug that was not putting the current date and time when using the
560	Terminal	Fixes bugs that were not updating the quantity of the work order when inse
561	API	Now there is an API entry for report Analysis/Productivity.
561	Terminal	Fixed Issue: When charging field values for results (or IO), it tried to load fi
562	Integration	Now production is synchronized with ERP according to product type.
562	Terminal/Integration	Avoid sending zero production message to the ERP while starting order.
562	Terminal	Fixed issue in user invitation. It was sending father area id instead of curre
563	Terminal	Fixes a bug that was not allowing to start a work order when there was an
564	Terminal	Now the calendar can be managed by creating and deleting statuses even fo
564	Terminal	Fixes a bug that was not updating the Recurring option when creating a wo
565	API	The load of the tiles from the terminals has been optimized because now in

		(by default is empty) and set if you want get items recursively (by default i
566	Terminal	Automatic scheduler has been improved so that now takes into account mor these possibilities.
567	Terminal	Now you can browse unread messages.
567	Terminal	Now PSR relations can be copied an pasted to other processes.
567	Server	Fixes an error (Duplicate Entry) when copying and pasting a process.
567	Server	Envision (prediction of failures using Neural Networks) has been improved. administrator. In addition to many internal improvements, it is also more p
569	Terminal	Now user can import/export the work orders via CSV files.
570	Terminal	Now you can see a URL document stored in a field with the folder button us
570	Terminal	Results window, and starting a work order, have been speeded up as results
570	Terminal	Schedule window has been speeded up as calendar is only loaded when neede
571	Reports	Fixed bug: when user logged out from Reports, the URL used the server nar
573	Reports	Fixed bugs in Reports Cache.
574	Terminal	Fixes a bug that was not allowing the user to change result when it needed process.
575	All	Component XML4 has been updated to XML6 in all the system (AlertsUpdate StockSync, some UDLs and Wizard).
575	Terminal	Now, reference changes in inputs and outputs can be done from the results
575	Terminal	Now you can test the URLs configuration before saving the new configurati
576	Terminal	Now the schedule CSV import and export allows to manage the recurrent ar
577	Reports	Fixed issue in Results\Results report. It was grouping by machdate, hidding s Additionally, it did not show all the available information about inputs & out
577	Reports	Type and class have been added to reports related with statuses.
577	Server	There are changes in the structure and content of internal multi database f with http or https [depending on the security configuration of the webserve
578	Server	Fixes a bug that was creating autocontrol tasks related to statuses even wh

579	Terminal	Fixes a bug that was causing to not load processes in the main form after c
579	Terminal	Fixes a bug that was causing an error when forwarding (sharing) a commer
579	Terminal	New Calendar report, showing the calendar of all processes.
579	Terminal	Now user can use MTP "paste" botton with multiple selection.
580	Server	New auto register option for new users. This feature requires a new line to Files\edinn\edinnM2\Server\websrv\Apache\conf\extra\httpd-vhosts.conf, aft
580	Terminal	Fixes a bug by which the alert of new messages blinked but then, when the
580	Terminal	Fixes a bug by which, sometimes, the week number shown in the date and
580	Terminal	The reserved word ALL for lines, and its special behaviour, have been depre
581	Reports	Fixes a problem in the Wizard of company creation that was not inserting A
581	Terminal	Now, when a result is marked in PSR relations to be propagated, the result
581	Server	Now Autocalendar replaces calendar periods instead of temporary deleting t completion of work orders.
581	Terminal	Now the View Schedule window is only refreshed when the user clicks the re
581	Terminal	Now when a work order is assigned to a process it can inherit its PSR config
581	AlertsUpdate	Fixed issue: Alerts now propagate correctly across areas of areas, whether
581	Terminal	Now the area chart shows the alerts of the processes it contains and you ca
581	Reports	Now, the ratios report in area chart mode shows alerts for the processes it
581	Terminal	New property "Archived" to archive areas. Archived areas do not appear in t
582	Server	Now the automatic update of server saves disk by leaving only 2 folders for
582	Terminal	Now the user can assign roles of process resources from the main window w
582	Server	The precision of Big Data has been increased as now the monitor dumps res quantities in results matching those of reports and dashboards, by avoiding
582	Server	Third party components have been upgraded to avoid vulnerabilties and imp (5.7.35).
582	Server	Fixes a bug by which statuses set in the calendar were not being passed to

582	Terminal	Fixes a bug by which area administrators could not assign roles to the proce
583	Terminal	Now the updater and the setup uninstall and install the ODBC driver, as a wa
583	Terminal	Fixes a bug that was making not possible to navigate to URLs which contain
584	Terminal	MS Project (XML) and FIEBDC-3 (BC3) files are not supported anymore to im
584	Server	Fixes a bug that was causing some FAI statuses to appear blank in the statu
585	Reports	Fixes a bug in the filter by status of the Schedule\Schedule report.
586	Server	Now the representant processes have the same OEE of the areas they belon
586	Server	Now the optimizer does not perform backups of the m2logs table to assure
586	Server	Now the VeryHighDetail option of the logs will only last activated a maximur
587	Terminal	Not it shows a message when loading a work order of which result does not
588	Reports	Now backups of the data are available for download (only for super adminis data.
588	Terminal	New event for automatically launching autocontrol tasks: Field change. It ap field changes in results.
588	Server	Now signals of production, scrap and rework pulses, in combination with the edinnManual, edinnBOX_1 and edinnSIM_1.
588	Reports, Terminal	Now you can customize the information shown in the path and the hourly be
588	API	Now you can customize the KPI list in the functio ProcessKpiPanelGet.
588	API	Now you can delete status searching by process, finish date and start date.
588	API	Now you can delete production order or tasks searching by process, and ord
588	Server	Now processes will auto justify with the manual entries of the calendar whe
589	Terminal	Fixes a bug that was sometimes crashing the terminal when starting a work
590	Terminal	Fixes a bug that was causing to check changes in fields when finishing a wo
591	Reports	Now you do not need to be Administrator to test the API from the reports.
591	Terminal	Now user can change working orders priority changing their grid position.

591	Terminal	Now user can specify fields for the autocontrol option "field change".
592	Terminal	Now, when creating an area with right mouse clic from the main window, i
592	API	Now you can use reschedule to change planned start date, planned end time
592	API	Now you can replace a whole parameter of a work order.
593	Terminal	Fixes a bug by which areas of areas were not reflecting correctly the color
594	All	Internal treatment of dates has been improved to avoid problems with certa
594	Terminal	Now, notifications of changes in the status of work orders can be disabled f
594	Terminal	Now, every time a change is done in PER Relations, it can be applied to all p
594	Terminal	Now, when importing work orders from a CSV file, duplicates of work order
595	All	Internal treatment of dates has been improved to avoid problems with certa
595	Terminal	Fixes bugs (was changing the WBS priority field, was not assigning work to
596	Terminal	Now in Schedule Edit form, only areas of processes can be selected.
596	Terminal	Fixed issue in Schedule Edit form: Now edinn will recalculate cycle time afte
596	Reports	Now you can view status summary and evolution grouped by working order.
596	Terminal	Fixes a bug that was importing work orders from the clipboard with an inco
596	Server	Now emails are sent more robustly by the server, also accepting new domai
597	Terminal	Fixes reescaling problems in the window to edit scheduling.
597	Terminal	Now, when marking 3 messages as viewed in the same user session, the ter
597	Reports	Fixed issue: Some database connection were unclosed.
597	Integration	Fixed issue in B2MML function to write XML files.
597	Integration	Removed unnecessary getchunk reading XML from database.
597	API	Added funtion to get processes from an area.
597	API	Added a parameter in processes get results to filter by type.
597	API	Added a parameter in processes get statuses to filter by type.

598	Server	Now the path graphic internally supports to now start from 0 when a result
598	Terminal	Fixes a bug by which the vertical scroll bar of the schedule editing window v
599	API	Fixes a bug of duplicate id when inserting statuses in calendar.
599	Console	Hides unused parameters and checkbox in Console/Interface: Notify on stat
599	Server	Fixes a bug that was causing the central service to not send data to the B2/
599	Reports	Fixed a bug in report analisys->pie. This report did not work for areas.
599	Reports	Now you cannot add the home of the reports to favorites and reports_cach
599	API	Added a parameter in API calendar in statuses get to select from actual stat
599	Terminal	Now users can launch the automatic scheduling for a specific date.
600	Terminal	Now users can import working order relations from CSV.
600	Terminal	Fixed issue: Duration always 0 for areas in view schedule type Gantt.
601	API	Added parameter showfields to results config get request.
601	API	Added parameter type and showfields to processes_statuses_results relation
601	Reports	Increases the URL field length of favorites table to 2048 chars.
602	Terminal	Due to technical limitations, sizes of URLs have been changed to 341 for th
602	Server	The optimizator now does not backup m2reca to be able to perform the ba
602	Server	Fixes a bug in BigData by which it was not updating the record following the
602	Server	Now the Central service does not send statuses to the interface when it is c
602	Terminal	Fixes a bug in the editing schedule window that was omitting changes.
603	Server	Fixes a bug in BigData which was showing an error in the logs.
604	Terminal	Fixes a bug by which it was not automatically adjusting the start time of a
604	Server	Fixes a bug in Big Data which was not updating correctly when only an end t
605	Terminal, Reports, API	Added customized data CycleQTheor & CycleTTheor.
605	API	Added a paremeter prefix to filter PSR relations by the result.

605	API	Added a paremeter history to indicate if all the PSR relations for a result wi
605	Terminal	Now, when you change process-status relacion, the terminal free the cache
605	Terminal	Now, when you acces to view schedule window from main windows with a p
605	Terminal	Now, you can copy and paste roles/functions between resources.
605	Server	Reports caching now is done with a task that is programmed per company, reports of another in the same server.
605	Server	Now reports caching is done starting by the reports with the shortest period
605	Server	Fixes a bug that was justifying old statuses as future statuses.
606	Server	Fixed issue in MachPeriodGetResults causes timeout error in query.
606	Server	Work orders will not be purged because of Real time oldest records. This be
606	Terminal	Fixes a bug that allowed the user to program a future status without end ti manually inserting any status.
606	Server	Now the monitor converts to capital letters any data read in a Result_Id sign contains an id of a status or an id of a result, and they must be capital letter
606	Server	Fixes a bug in ScheUpdQt: the order quantity was not updated if there were
606	Server	Fixed issue in BigDataPrevious when there is no previous record.
606	Terminal	Fixed issue in FieldsGenerate when there is no previous record.
606	Terminal	Now, in the calendar you can delete a status from all the processes it belong
607	Server	Now the wizard encrypts the backups compression password to be coherent
607	API	Added new API call to get informatio about one result.
607	API	Added new API call to get information about the PSR relation indicating one
607	Terminal	Autocontrol: Now you can launch a status when an autocontrol task is mark
607	Server	Now, when creating a company in the cloud, Community emails are not ger
607	API	Added new API call to create relations between results: ResultsConfigRelation
607	Terminal	Fixed a bug that was changing CycleT and CycleQ, to those indicated in the
607	Server	It has been removed the restriction that cycle times had to be greater than

		anymore.
607	Terminal	Fixed a bug tha was adding a new status when finishing order. Now only ad
607	Server	Added extra information in log on error exporting to CSV. Now ReportsCach
607	Terminal	Now the automatic scheduler allows to keep work orders in the same proces
607	Terminal	Fixes a bug by which when assigning personnel, when changing the date and made.
607	Terminal	Load of work orders in scheduling windows has been accelerated.
608	Terminal	When assigining a work order, request to update CycleT and CycleQ of work question.
608	Terminal	Fixes a bug when importing work orders with percentages in the fields whic
608	Server	Fixes a bug in Big Data that was causing sometimes an error in the record p
609	Terminal	Fixes a bug that was causing, sometimes, a crash in the load of the schedu
609	Server	Now the supervisor periodically checks the API and could restart edinnM2Rep
609	Server	Fixes a bug that was not closing statuses with maximum time when they wo
610	Terminal	Fixes an error in the Terminal win32 setup package.
611	Terminal	Improves OEE calculation so that when a process is producing more than the
612	Server	Fixes a bug in the Wizard for the creation of new companies that was dupli
612	Terminal/Reports	Sets pen width to 2 for production line in path.
612	Monitor	Fixed issue when you have EdinnHIP and other protocols, edinn only tries to
612	Terminal/Reports	Now the path graphic draws flat in not productive periods where there are
613	Terminal	Fixes a bug by which when giving access to a user to an area, was removing
613	Monitor	Now more methods are capable of managing all the signals. edinnBOX_1, e
614	Terminal/Reports	Fixes that the path and speed graphic were not showing the scale in the Y a
614	Terminal	Now, if a process is scheduled, it behaves identically when modifying a resu

		to supervisors. Before this change, modifying was possible to supervisors bu
615	Terminal	Now you can add dashboards by company.
615	Reports	Fixed issue: When a favorite report reurns no rows reports_cache crashed.
615	ReportsCache	Now if you schedule a list of reports to reports cache, they will be calculate
615	Terminal	A field marked as to be asked at work order starting, was asked twice by e
615	Reports	Fixes a bug in the Operation report by which processes which started worki
616	Reports	Fixed a bug where description in results was truncated, but only when selec
616	Monitor	Now, when a monitored device is not availble, the monitor will not create r item...". It will create only 1 and another when communication with device
616	Terminal	Fixed a bug in Locate window, when showing reults of many processes, the they did not appear.
616	Terminal	Now terminals will connect with the API through a new field de in companie
616	Server	Now server applications will connect with the API through the field APIHost.
616	Server	Tries to address a problem where the server component edinnM2_CentralPo
617	Terminal	Now, when re-assigning work orders, also the area can be changed.
617	Reports	Fixes a bug in the Registry\Not recorded report.
617	Monitor	Now the monitor generates SPC alerts when measures are out of limits.
617	Reports	Fixes a bug in reports cache that was removing URL of a favorite when the
617	Reports	Now reports cache reopens the conection for each report to avoid timeout
617	API	Now in processes/setatuses/results relation get call, API will show result equ
617	API	Now in bigdata get call you can leave process in blank and API will show you
617	Reports/Terminal	Fixed issue in graphic production path. Now production path is horizontal w
618	API	Fixed a vulnerability in API login since build 597.
618	Terminal	Now it can be configured, in resources, what must happen to the status of

618	Reports	Corrected an issue which was making the URL of some favorites to be delet
619	Console	When Creating the first company, now also the M2Reports user is given acc
619	Terminal	Solves an error that was appearing in the main window, when loading the t Host).
619	Server	Changed parameter in Apache configuration ThreadsPerChild to 500 to allow
619	Terminal	Fixes a problem that was accepting empty fields that were mandatory when
619	Server	Fixes a bug that was preventing the summary of alerts to be sent by email.
620	Reports	Fixes a bug in the contains filter and in the Class column of the Statuses\Sta
620	Reports	Now you can get the ratios and ratios-evolution report of areas of areas an
620	All	Represent property has been moved from process to area-process relation.
620	Server	Now messages between the Monitor and the Central are not deleted, but m analyze problems related with these messages.
620	Server	Fixes a bug (in the database triggers) that was preventing Big Data to be u
621	Server	Fixes a bug causing the Central to not consume messages.
622	Reports	Fixes a bug that was causing the operation time not to be considered per t
622	Reports	Fixes a bug in Reports\Ratios Evolution that was showing the shift name in
623	Terminal	Fixes a bug by which the estimated date and time to finish a work order wa form.
623	Server	When updating, the task Reports_Cache is no longer re-scheduled.
623	Server	M2B2MML no longer sets automatically a setup time.
623	Terminal	Fixes a bug by which PSR CycleT and Q were not being populated to a work
623	Terminal	Fixed error in the query for manual input of resources.
623	Terminal	Now the labels of hourly dashboard will adjust their sizes to the size of the
623	Reports & Terminal	Now the path will show a new theoretical line as the projection of the curre
624	Server	Now the windows process id is included in the system logs which is useful to

624	Reports	New Night Mode in the web reports.
625	Terminal	Fixed an issue in the PSR configuration window when tring to modify a resu
626	Terminal	Schedule editing filter, when selecting All, properly includes pre-scheduled w
626	Reports	Performance of the web reports has been improved as now the reports cach
627	Terminal / Reports	Now the path graphic can be configured to have the axises outside of the g check the Edinn Server Installation and Maintenance Guide.
628	Server	Minor stability improvements, like for example that now Big Data updates c although it could unlikely cause certain big data calculations to be repeated.
629	Terminal / Reports	New visual theme "Softer" and now themes are available in the reports.
629	Terminal	Now, when a state is a consequence of a propagation you can navigate to t
629	Terminal	Reactivates autocontrol tasks launching from terminal.
629	Server	Corrected an error where conmunication with ERP failed, in ProdSync.
629	Terminal	Manual operations with work orders have been speeded up when the server
630	Terminal	Fixed issue in assign personnel window, making IN to an area.
630	Reports	Fixed issue in dashboards. Now font size in widgets will adjust to the availa
630	Terminal	Fixed issue in path control of dashboards. Vertical alignment of labels did no
631	Reports	Fixed issue while dropping column in RepRatios.
631	Server	Fixed exception in BigData module where a very short period without previo
631	Terminal	Fixed issue in autocontrol configuration. If you had only one process, you w
631	Terminal	Modified scrollbars of grids.
631	Reports	Fixed issue in report schedule/followup target column showed wrong data.
632	Terminal	Allows cancelling multiple selected work orders from the View Schedule windo
632	Wizard	Fixed issue creating process for a new resource.
632	Terminal	Now, in Configuration\Process Status Result Relation, the history of all the r
632	Reports	Fixed issue in autocontrol report. Comments were copied to the next record

632	Terminal	Fixed an issue that was showing pre-scheduled work orders when selecting t
632	Bigdata	Fixed an issue thas was getting production reports previous to the start tim
632	Server	Fixed a bug that was archiving work orders even if they were not finished.
632	API	Fixes a bug in ResultsCreate (POST) which could be causing problems.
632	Terminal	Fixes a bug that was not showing the colours of the processes according to
632	Server	The MTBF calculation now does not consider as a different failure a consecu
632	Terminal	Now the look up list of the on screen keyboard shows up over the keys, and
632	Reports	Fixes a bug that was showing incorrectly OEE in the Recommendations\Diag
632	Console	Fixes a bug by which it was incorrectly indicating of no Envision license whe
633	Server	Update of work orders (ScheUpdQt) has been improved to reduce work loac
633	Server	Propagation of statuses has been improved to take exceptions into account.
634	Server	A bug has been corrected and now representant processes are propagated w
634	Terminal, Reports	Added new parameter to path: now you can show the comment of the activ lines.
634	Terminal	Now you can cancel work orders that are not assigned to any process.
635	Terminal	Standard API Dashboard has been improved to consider longer periods.
636	Terminal	Fixed a bug that affected position and scaling of the labels in the path.
636	Reports	Fixed a bug in report Calendar: substring out of range.
636	Reports	Fixed a bug: the path was not being cached.
636	Reports	Fixed a bug: when reports cache user had no permission to show any proces
636	Reports	Fixed a bug where Hourly Board would not adapt to selected theme.
636	Reports	Removed detail check in activity report as is useless.
636	Terminal	Fixed issue using areas with 0 as id.
636	Wizard	Fixed issue registering in a company without processes.
636	Reports	Fixed issue where ReadProduction could collapse under specific circumstance

637	Reports	Path fixed to adapt properly to selected theme.
637	Console	Now it does not check the reports level in order to delete the backup files.
637	Terminal	Now you can configure autocontrol tasks to automatically open previously as
637	Terminal	There is a new KPI called Id_Prod to show the current result.
638	Terminal	Fixed a bug that did not allow autocontrol tasks to be deleted.
638	Reports	Fiexd issue: In report ratios->evolution the data of the last area appeared c
638	Reports	Fiexd issue: exportCSV did not write title for each period.
639	Terminal	Fixed issue trying to remove autocontrol task when it is the unique row in t
640	API	Fixed issue in creating work orders from the API. CycleT and CycleS were ca with a duration given.
640	API	Fixed issue when creating results from API. Now you can create a result wit
640	Console	Fixed issue when renaming a process from the Console. The update query w
640	Terminal	Fixed issue when launching autocontrol tasks by status (time usage) change
640	Terminal	Fixed issue in in&out operation by areas.
640	Terminal	Fixed issue showing fields in a new production record in timeproduct.
640	Reports	Fixed issue. Export to CSV was creating empty files or files with an extra en
640	Terminal	Fixed issue. Roles were calculated before using them.
641	API	API: Renamed functions ProcessDashboardGet, ProcessRecourseRelationCrea
641	Terminal	Fixed issue in resources In&Out management, when a resource has to be ou
642	B2MML	Fixed issue in SegmentResponse Insert method. MachDate value was miscalc
642	Reports	Fixed issue in Ratios Evolution by areas, Area did not appear in the first col
642	All	Function DebugToFile: pBolCloseFile param added for compatibility with Writ
642	API	Added option to Insert function of Prod.cls to replace record.
642	API	Fixed issue in InsertUpdateProductRelation function: process defaults for Cy
642	API	Added option to Insert function of Prod.cls to replace record / Fixed issue in

642	API	Enhanced function explodeString. Now it will return an empty array [] for n
642	API	Enhanced response treatment. Now function returns data when Error is zer
642	API	Added replace parameter to ResultsConfigCreate function. Now you can rep
642	Terminal	Fixed issue in process assignment of working orders, process was wrong ide "Calculate" instead of "Update".
642	All	Fixed issue in triggers. Now if you modify the type of a time usage trigger
642	Reports	Fixed issue in report production when trying to show inputs and outputs.
642	Reports,Central	Fixed issue in ratios report. MTBF indicator was misscalculated.
642	All	Now, if you have pending autocontrol tasks of type Quality, a Quality alert
642	Reports	Added a result filter to autocontrol report.
643	BigData	Fixed issue in bigdata calculation (subscript out of range).
643	Terminal	Now when starting a paused work order, autocontrol tasks for starting orde
643	Terminal	Fixed issue: Now you can start a finished work order only if you have sched
644	Reports	Fixed column ordering on reports.
644	Terminal	Fixed issue: If you click on quality alert button on the main window and you
645	Terminal	Fixed issue: In autocontrol window, fields text box returned an error when
645	Terminal	Fixed issue: Disabled F5 key in view "schedule periods" window.
645	Terminal	Fixed issue: Forced schedule calculation when modifying a period even if the
645	Reports	Improvement: Now you can show the reference field in schedule reports.
645	Reports	Fixed issue: In report Status Summary time class array was not initialized pr
645	Terminal	Improvement: Now you can stop a work order that requires indicating the g
645	Terminal	Fixed Issue in autocontrol window for tasks configured to show a document
645	Reports	Fixed an error which added an extra semicolon at the end of the exported
645	Terminal	Fixed an issue where you could justify the same type of status only if you ha

646	Terminal	Fixed an issue where you could justify the same type of status only if you h
646	Reports	Now you can hide the column titles in the widget by setting text size for tit
646	Reports	Adds warning if there are no columns to drop in report ratios by areas.
646	Reports	Excludes from sorting columns of reports: panel, profitability and home.
647	Optimizer	Now the optimizer deletes old logs in batches of 7 days.
647	Reports	Fixed issues in report schedule followup: Total time column was defined but selected working order.
647	Terminal	Load production records was enhaced. Now load prodution records does not descripton is cached and database access was reduced.
647	Terminal	Fixed issue loading filtered products. Id field was ambiguous.
647	Reports	Fixed an issue where report schedule followup was not checking if the data
647	Console	Now the console stores the port that is being used in order to read it in PHP
647	Reports	Fixed issue in analysis->followup report. Now only speeds above 100% will be
647	API	Fixed issue in the creation of results through the API. Creating relationships
647	Terminal	Improvement implemented in the edition of work orders. Now, when start instead of every time the focus of the control is changed.
647	Terminal	Improvement implemented in the change of status of work orders. Now wh to 0 and the estimated finish time will be cleared.
647	Reports	Fixed issue in the planning follow-up report. Now the quantities and the obj of measure field, the unit of measure of the process is written.
647	Terminal	Fixed issue in stock window. Now you can ask for stock and send consumpti
647	Terminal	Now planner uses WBS field to sort work orders.
647	Terminal	Now you can reassign priority to prescheduled orders.
647	Server	Now the optimizer optimizes data in weekly periods, to avoid becoming hur
647	Reports	Fixed issue in ratios report: MTTR param is displayed in hours
647	Terminal	Now you can finish a working order in manual processes and notify a good c
647	Integration	Fixed issue in unit conversion.

647	Terminal	Added identifier of material input to inventory window to enable deletion of
647	Terminal	Fixed date and time comparison issue when modifying work orders.
647	Terminal	Added a checkbox in the table view of view schedule window, that shows th
648	Integration	Now, if time is not sent in the order but quantity is sent, the order will tak
648	Terminal	Added a checkbox in the table view of view schedule window, that shows th
648	Terminal	Fixed an issue when selecting a value in a field in autocontrol.
649	Reports	The power of the edinn web system has been improved by activating variou
650	Server Update	Fixes a bug in the distribution of the server update.
651	Supervisor	Now the supervisor also checks the API Tree, including a time out, in order
651	Monitor	The following monitoring drivers have been separated in 2 threads for coun
651	Integration	Fixed issue in integration selecting a working order periods with the time li
651	Integration	Fixed issue in integration. Now activity percentage in PSR relations inherits
651	Server	Now the central will close a status for which it has information of its closing
651	Server	New option in the Console, Behaviour/Optimizations allows to generate in t
651	Autocalendar	Now you can 1) add setups of work orders as future statuses, and 2) to calc
651	API	Added data types to the API call ProcessesStateRead.
651	Monitor	Now the UDLs and OPC Bridge support writing of items into PLCs even for c
652	Terminal	Fixes a bug which could crash de application when a measure was marked a
653	Monitor	Now the monitor is robust to counters generating results 500% of the nomi
654	Terminal	Improves the load of the work order in the Results windows, which in additi
654	Terminal	Fixed issue starting working orders. Some field values were duplicated.
654	Server	Fixes a bug when exporting favorite reports to files and sending them by e
655	Terminal	Now, the date and time of a comment is set with the work order end date scheduled time.

655	Terminal	In the log window, if the time of the found log is not between the start and earlier than the current time, the current time is used as the end time.
655	Terminal	When you select a message with the request to plan an order, it sends you window.
655	Terminal	When edinn sends a production or consumption modification to the ERP, it
655	Reports	Fixed issue in date time format in productivity and followup reports.
655	Server	New parameter in the Console, in the Monitor and Results tabs, named "Inc sources. More info in the user's guide.
655	Terminal	Fixed issue in automatic planning. Now takes into account the dates of ord
656	Optimizer	Sometimes automatic backups were not being compressed and sent to the been made more robust.
656	Console	Fixes an "Object variable not set" that was being showed in the logs when r
656	Terminal	Added a menu in the bad item of the hourly board that enables to choose b
656	Terminal	Now, when you change the date and time of a period of a work order you
657	Terminal	Now statuses of which children statuses are included in the PSR relation of title.
657	Terminal	Fixed issue in datetime format adding inputs and outputs.
657	Reports	Fixes a bug which was not showing correctly the colors of records in the rep
657	API	Added options to Processes State function: showarea (displays the parent ar fields and field values), showplanning (displays the next planned status with
657	API	Fixed issue in Activity calculation by quantity column.
658	Server	Fixed issue which was not sending emails.
658	Terminal	Fixed issue when trying to show comments from a not started working orde
658	Terminal	The launch of autocontrol tasks due to order change will not be carried out
658	Server	When the status of a self-monitoring task is changed, it adds an indication
658	Reports	Now, when you export a report to CSV, the file will include a title row abov
658	Reports	Fixed issue in time calculation of schedule follow-up report.

658	Scheduler	Fixed issue in planner. Now, in fordward planning, working orders are sent
658	Help	Added more detail in autocontrol configuration section: only tasks programe
658	Terminal	You cannot assign negative quantities and you cannot assign zero units unles
659	Terminal	Now it does not allow to create areas without filling the fields of processes
659	Terminal/Console	Added option to calculate estimation date and time by NPC or OEE.
659	Server	Now, autochanges calculates estimated date time avoiding overlaps and gap
660	Terminal	Enhances Process-Result configuration. Now you can filter by a part of the r
660	Terminal	Fixed issue in results modificacion. Now, when modifying a result edinn will
660	Server	Added an option to calculate estimated finish time of working orders only o
660	Reports	Fixed issue in setParamValue adding new parameter in the URL was replacin
660	Reports	Fixed issue in report Conumption:Summary.
660	Central	Fixed issue in estimated time calculation for working orders. Now preserves
660	Monitor	Fixed issue. Now monitor only updates signal values if the request to write
660	Monitor	Fixes an error when receiving data of type 19 by the OPC/UA driver.
660	Terminal	Fixed issue in inheritance of field values in IO reference changes.
660	Reports	Fixed issue in report ratios->evolution. The report was always displayed in g
660	Central	Now BigData has a dedicated thread for current shift, to avoid that change
660	Integration	Fixed issue in work order creation. Now if PSR relation has different cycle t
660	Server	Now autocontrol tasks configured by time and with a minimun production t
660	Reports	Improved visual adjustment of legend in some widgets of reports.
661	Server	Now autocontrol tasks which are generated by result change are also genera
661	Terminal	Work order will be shown in autontrol manage window.
662	Console	Now, you can activate or deactivate automatic estimation date and time fo
663	Terminal	Fixed issue when trying to save data in a work order without estimated dat

663	Reports	Fixed issue in the "schedule/follow up" report. Now, quantity shown is limite
663	Terminal	Fixes a bug that was preventing new autocontrol tasks to be generated whe
663	Terminal	Now, in the autocontrol configuration window, shows the unit produced (re
663	Terminal	Fixed issue in SPC Control graphic.
663	Terminal	Fixes that the configuration of small buttons in the main window was not k
663	Server	Now the estimation of finish date of the started work orders are calculated
663	Integration	Fixed issue in workorder import. It was creating duplicated PSR relations.
664	Server	Fixed issue. Avoid to close a state when central is inserting another state th
664	Big Data	Remove unnecesary transaction when processing queries in bigdata.
665	Terminal	Fixes issue in schedule edit window help.
665	Server	Fixed issue when dividing a FAI status by shift.
666	Console	Now the use of modules is calculated for the last 365 days, instead of 30 da maintenance yearly.
666	Terminal	Fixes a problem when making IN/OUT on areas.
667	Terminal	Fixed issue: Overflow in schedule edition form.
667	Terminal	Fixed issue in bigdata. Now active order changes only when a change in pro
667	Reports	Added time usages to activity report.
667	Reports	Options "Only productive" and "Not scheduled" have been removed from repo
667	Reports	Option Top in report Statuses\Summary has been improved so that it is appl
667	Reports	New option "Only required" which shows only periods with required (demand
667	Terminal	Fixed issue in recourse assign. If a recourse has not multiprocess role, it ca than zero will appear as asigned areas.
668	Server	Improved AutoCalendar and now deletes future statuses which belong to pro
668	Console	Disables the option in the console to create setup time usages in the calend
668	Server	Added a function to avoid overflow error in data type conversion from strin

668	Server	Fixed issue in estimated seconds calculation.
668	Server	Now states with 0 seconds are removed by the server (CENTRAL) except if
669	Terminal	Added parameter in resources configuracion: Maximum Time (Max. Time): After this time, the system will automatically OUT it. If left zero, it will not
670	Terminal	Avoids a delay that was occuring in installations where there are a lot of rec
670	Terminal	Fixed issue in in&out management, trying to make in or out with the optio resources in the area.
670	Reports	Now in schedule, followup when speed is over 100%, the ratio value is mark
670	Reports	Now the shifts are ordered by Id instead of the start time, so you can choo
670	Terminal	Fixed issue in shifts configuracion. When you tried to modify a shift, the fo
670	Reports	Results filter only cares about results of production type.
670	Terminal	Now prevents configuring a selection or multiple selection type field without
670	Reports	Fixed issue in Activity Report. Now you can cache activity report without se
670	Reports	Fixed issue in the analisys panel report. Now you can show in the speed gre
670	Reports	Fixed issue. When operators made out from a process or an area, they mad
670	Reports	Removes asking when making out if a resource has the multiprocess role.
670	Central	Improves the function which is updating the table of status of processes.
671	Server	Fixed issue by adding a semaphore to the integration process to avoid sendi
671	Server	Fixed issue in hourly dashboard report.
671	Terminal	Fixed issue in fields configuration form. It showed an error message if you multiple selection).
672	Server & Terminal	Now a change in a work order is not considered when going from a valid wo
672	Server	Now OEE and OCE are not refreshed in the Terminal for areas which have b
672	Reports & Terminal	Now you can show activity target in production path and in the panel report
672	Reports	Added quantiles selector to the histogram report.

672	Terminal	Now processes marked as not productive are not shown in the windows to v

2020-07:

- New selector for type of Ratio in the Panel report.
- To ensure that edinn servers are secure, its PHP engine must be upgraded to version 7.3.11.
- Now edinn does not ask for change status when stopping order on automatic processes.
- Added to Consumptions/Summary: Filter by working order, totalization by working order and possibility to see the detail by clicking.
- Security of the edinn M2 web services has been increased.
- All the names of the tags related to working orders, have been changed from OT to WO.
- Now the Windows registry is modified in every update of the terminal in order to guarantee the correct performance of the dashboards and the writing in the database.
- New automatic management of backups.
- Improves Automatic and Manual update of terminals.
- Now the Autocontrol tasks which have zero in the 'Every' fields, and which will not be automatically changed to status Not Done, are automatically changed to status Expired when a new task is created with the same id. This is to avoid the accumulation of old tasks.
- Now a status can be automatically generated every time an Autocontrol tasks is performed, this is, when it passes to the ok state.
- The Big Data and the Artificial Intelligence (known as Envision in edinn) modules have been separated, which requires relicensing.
- Fixes a bug in the Console, Tools, that was not copying a process.
- Added a new option on Analysis/Evolution to add a column with the resources (persons) Ids which did IN in the period.
- Now the server updater also stops the web service when updating as some files could not be replaced if the service was running.
- Now, users with the ADMIN role have an account (COMPANYID_IDOFUSER) for Remote Apps (server console and terminal) in case this feature is configured.
- Now the reports service cannot be operated from the console in companies running in edinn cloud (ecloud) as it could affect other companies.
- Added area and process to the autocontrol report (with hourly option).

- Now you can set a css class for a singular cell.
- Now, when you click on the terminal dashboard, the default web browser is opened instead of IE11.
- Fixed quality percentage issue in Report Analysis-Pie-Detail.
- Now, if remote applications are activated, for those users which have the Administrator role, their user ids must be less than 10 characters in long and their password must be strong enough. This is due to compatibility with other systems.
- The Artificial Intelligence (Envision) module has been improved (dynamic memory, self-training function and multipurpose) and tested.
- Improves the monitor service which was adding too much time when the Add Cycle Time and Add Threshold options were selected.
- The calculations of the Big Data task have been improved especially when using IN&OUT of users and teams. This will also improve precision in Reports.
- Fixes a bug in the reports Consumptions\Consumptions, that was causing an error in the Rate column with consumptions associated to results
- Fixes a bug in the caching of reports as it was using the decimal point and the language of the generic user to cache reports instead of the user who created the favorite.
- Now the View Logs list from the Console clearly indicates if a record is Warning, Error, Normal, etc.
- Now the Console\Activity tab has a button to delete the queue of pending messages from the Monitor to the Central.
- File FileReplace.exe, which is used in the process to update the terminal, has been renamed to edinn_FileReplace.exe, to let the user know that it is a safe file.
- Now the automatic update of the server also re-creates all the database users, to increase security and to keep these permissions updated.
- Fixes an error that did not permit to change the Period dropdown list in the win32 terminal under certain circumstances.
- Fixes an error by which string signals did not arrive to UDLs.
- Fixes an error in the interpretation by Big Data of Teams IN/OUT.
- Improves response from server to edinn OPC Bridge for it to consume less resources from the computer.
- Improves stability of the Big Data when in big load environments.
- Improves the efficiency of the Big Data when massive processing is needed.
- Fixes an error in Autocotrol Report. The report did not show the production order in its

column when there was not comments. The column name and tool tip was wrong.

- Fixed a problem that was restarting the OPC Bridge.
- A new help document to optimize the edinn database has been created in the server documentation.
- Fixes a bug that was requiring the license when showing the API dashboard in the terminal.
- Fixes a bug that overlaps some graphics with the tiles in the main window of the terminal.
- Changes the estimated date and time to finish an order when does not have target.
- Now the OPC UA server monitor shows the obtained endpoints if in very high detail or in case the indicated endpoint is not valid.
- Fixes an error that was not showing the path graphic in the main form when opening the process from the tree of processes.
- Now an Autocontrol task can be associated with multiple results.
- Now if the Only productive check is marked, resources only appear in ratios evol when there is a quantity produced by them.
- The current active working order and result are now shown in the report Autocontrol\Autocontrol.
- Now the Autocontrol\Autocontrol report can show the fields in horizontal.
- Now in the report Analysis\Evolution, if the Only productive check is marked, resources only appear in ratios evol when there is a quantity produced by them.
- The current active working order and result are now shown in the report Autocontrol\Autocontrol.
- Now the Autocontrol\Autocontrol report can show fields in horizontal.
- Now the server updater also performs an update of the DB client user.
- Now the console also updates the server updater exe before starting the update of the server.
- Fixes a bug in the BigData process which could affect some reports.
- Fixes a bug that was not showing colors and was causing an error in the Locate window of the win32 terminal when showing In and Out records.
- Fixes a bug that was not showing the API dashboard in the main window of the win32 terminal when coming from a tree view.
- Fixes an error when starting an order which generates autocontrol.

- Fixes an error that was showing certain orders in black after using the Automatic Planner & Scheduler.
- Now, for those processes which are configured to be forced to follow the schedule, in the Results window, when the order id is typed, the correspondent Result code is loaded into the field in the same window.
- Fixes an error that happened in certain computers when creating an order.
- Fixes a bug that was showing the API dashboard overlapped with the processes in tiles.
- Fixes a bug that was inserting scrap results as production when using Goods and Totals counters.
- Fixed a bug that was showing a white screen on presentation mode, when only one tab was configured.
- Fixes a problem that was not showing the calendar correctly. In addition, now the update progress of the calendar is better shown.
- Added an option in the report results, summary to show the inputs and outputs.
- The help document of the procedure to automatically or manually update an edinn server has been improved.
- Added API GET call to Big Data.
- Fixed error in Results -> Summary. Now I/O records quantity is calculated by Qt Outputs - Qt Inputs for each result
- Fixes an error that was taking the previous working order for results even when the process was marked as Scheduled and there was no working order.
- Fixes a strange behaviour in the win32 terminal where, after certain combination of Keys, you could not select any row from the Results grid.
- In the Reports, periods of previous 15 and 30 days have been changed to 14 and 28 days to be multiples of weeks. Also future periods for 7, 14 and 28 days have been added to all reports.
- BigData calculations have been optimized to mark as processed the last calculation done, to avoid repeting it.
- Fixes a bug that, when copying & pasting a process, was not updating its order.
- Fixes a bug that, when associating processes to area from the pop-up menu on the Main form, was not updating the area formula.
- Fixes a bug that, after using any of the buttons of the View Schedule form, sometimes, it would show the buttons as active although no order was selected.
- Fixes a bug that was not showing the last totals in the Results\Summary report.

- Now, when scheduling orders automatically, the user can type the maximum number of possibilities to consider per order.
- Fixes a bug that was making the Manual Update of the terminal to not complete all its tasks.
- A new Prescheduler role allows to create orders in preschedule status and modify and delete orders created by the user and in preschedule status.
- Fixes a problem when reordering columns when editing the schedule.
- Now controls that forbidden chars are not entered into configured fields.
- Now date fields are internally stored inverted. Conversion queries can be found in folder **sql\DB_Conversor** and must be launch manually, depending if the server manages dates in DD/MM/YYYY or MM/DD/YYYY format.
- Improves Autocontrol report by adding the number correponding to the status of the task.
- Now you can easily add style to a single cell by wrinting {style_name} before the cell value.
- Now Results button blinks in yellow more adjusted to the limit (when the remaining time to generate results equals the time limit to generate results after the compromised date).
- Fixed Speed0h calculation, now checks if the process is working to calculate the actual speed.
- Now, when the user does not have the reports role, the win32 terminal dashboard will be shown using the reports_cahe user and the user will not be able to click on it.
- Fixes a problem when generating Community companies.
- Fixes a bug when viewing internal edinn messages and searching for the next pending one.

2020-01:

- Update to 2020-01 could take several hours on servers with a lot of records in the registry. Therefore, we recommend to upgrade to this version in periods when there is no production (weekends or similar).
- Replaces Processing tag for Productive tag for consistency with status classes; Replaces Processing tag for Productive tag for consistency with status classes
- Now server licenses update automatically. From now on, all servers need a permanent internet connection and need to be relicensed for this version.
- Now the help button of the Win32 Terminal has been improved to provide access to the on-line help, on-line support and external help. The last option can be configured in the

new Recommendations tab of the server console.

- Now a comment can be sent to a device, in addition to a person.
- Now it is registered and shown the person who viewed an event, and when.
- Now you can mark an event as available for the users and therefore they can use it to send messages. This is useful to organize messages upon categories: suggestions, proposals, improvements, etc.
- Now you can share a message or event inside edinn or even with other external networks (social networks like twitter and linkedin will be added soon).
- Now you can respond to a message or event from the Registry of Events.
- Calculates the trend by % instead of by difference, in report Analysis, Evolution.
- In the Win32 terminal, shows recommendations (with a light bulb), congratulations (with a smily face) or need to improvement (with a sad face) on the majority of the statuses and results.
- Recommendations of the Win32 terminal include: duration of last status or result, comparison with current and its trend for a maximum of the last 10 ocurrences.
- Now the cached reports are stored in the database.
- New Community version when the license field is empty. It will provide 1 process with the Production Control and Big Data Modules. Some limitations apply.
- The help manual to install and maintain edinn M2 Servers has been improved.
- Fixed a bug that was not showing the touch keyboard on certain fields.
- Fixed issue in the date of creation of an scheduled order.
- New Terms and Conditions.
- In the Console, the 'Logs oldest record' database optimization field has been moved from the General to the Behavior/Optimizations tab, as this functionality is performed by the Optimizator task.
- Now you can test integration between your ERP and edinn using the integration report.
- Now you can assign steps of the routes to areas.
- Now the edinnM2 Terminal (win32 version) will not remain hidden in the task bar when closed. This is to improve its behaviour when running on tablets and other devices.
- New Calendar window which shows the calendar generated by the configuration of statuses. It is accessible from the schedule and statuses configuration windows.
- Sets target % to 0 for processes with no maximun quantity.

- Export to CSV takes the description of the favorite instead of the Id.
- From version 2020-01 build 108 or higher edinnM2Fields component must be renamed edinnM2Fields_CompanyId.
- Adds the option for sending information to the ERP in XML or JSON format.
- Allows new users to register themselves with an API call.
- Now server monitor UDLs must consider that numeric data will arrive always with the . as decimal symbol.
- Now, when creating or editing a favorite report, it is also configurable the size of the font of the tables inside widgets.
- For win32 terminals to connect to server, you need to reset the Client user per company and per server, from the console.
- Fixes a bug that caused the path to not show labels with accents.
- Now edinn sends statuses and labour times every time a status is modified.
- Fixes a bug that was showing statuses which started at the finish time of the Statuses\Summary report.
- Fixes a bug that was not showing the results generated exactly at the finish time of Results\Results and Results\Summary.
- Unification of column names in Reports: Theor.U. (theoretical units) reflects maximum theoretical units and Poss.U. reflects possible units considering the production time.
- Now you can export to CSV favorite reports with custom period.
- Now you can access the Terminal and the Console remotely from the reports. Needs additional configuration and could imply additional licenses costs.
- Fixes a bug on the speed graphic which was showing, in some rare cases, a extremely high data point when the previous record was zero.
- Fixes a bug that was not showing as marked the Add threshold option in the Monitor tab of the Console.
- Now autocontrol tasks can be disactivated to not be launched automatically by the system.
- Now all autocontrol tasks can be disactivated with just one clic from the processes configuration.
- Now, if a process has more than 1 working order started, the terminal will pause or finish all.
- From the main window, in the Tree visualization, administrators could not create an area which was not hanging from another area. Now this can be done by unselecting the

selected node and proceeding with the right click of the mouse.

- When you created a new area from the main window in the Tree visualization, the area did not appear immediately and forced you to exit and re-enter the application.
- Fixes a bug in the Cache of Reports task which was launching unnecessary internet explorers and causing out of memory and ActiveX problems.
- Now, if a process is marked as not productive, its signals and counters are ignored in the monitor service.
- Fixes a bug that was showing the header of the reports in certain processes in the terminal dashboards.
- All UDL for fields (edinnM2Fields.dll) need to be rewritten according to the new template. This has been necessary to reduce problems in the future as the public interface does not change in between versions.
- Added "Stock" property in fields creation. Fixes issue #914662.
- Fixed error updating password with the forgotten password option in the reports login section.
- Now, when creating a PSR relation, it detects if you have defined rework or scrap for that result, and only then, proposes to create also those relations.
- Fixes a bug which was forcing all working orders to have the 0-Production status as default.
- Fixes a bug which was allowing more than one active working order in a process when starting a new one.
- Fixes a bug that, in processes configuration, was showing incorrectly if configuration of results and statuses was allowed by operators.
- Fixes a bug that was not showing areas composed of more areas in the tiles visualizacion mode.
- Now the monitor will not dump production because of a stop when it has already dumped production after the date and time of the stop.
- Removed some limitations which caused to crash reports with hundreds of thousands of records. User should export to CSV these big reports instead of obtaining them through the web browser.
- Fixed issue in MTBF and MTBFP table in main window. Now selects by result and sorts by next stop.
- Fixed issue: the quantity of the working orders was not updating on shift change when there was no production in the last working order period.
- Fixed issue in Reports Cache when web port is not standard 80.

- Added new tag QxQxActivity[] option to Customization of data.

2019-11:

- Improved visuals of Win32 terminal.
- Edinn OPC Bridge has been improved to manage simulations.
- The monitoring step has been improved in the web pre-configuration wizard.
- Now the server console shows the type of license and the date for expiration.
- Now the user cannot configure cycle times which are less than the frequency of the server to check counters. This is to prevent problems when detecting failures and microstops.
- Some minor problems and bugs have been corrected.
- Now, when the terminal is forced to update, it will also update the language files.
- Now an operator cannot click New Result and, on that record which he is author of, change the quantity. This was allowed previously being a possibility for operators to introduce production.
- Now the web reports are more responsive and better prepared for its use from tablets and mobile devices.
- The setup, updater and restart of the win32 terminal have been improved to assure that they work robustly.
- An improvement in the Monitor service could avoid some very little microstops (those equal or less than the Check Counters Every monitor parameter).
- An improvement in the Monitor service will consider counters which are less than their previous, and inform with a Warning in the logs.
- Replaces Processing tag for Productive tag for consistency with status classes; Replaces Processing tag for Productive tag for consistency with status classes.
- Fixed a problem that caused, in Windows Server 2019, the Win32 Terminal to show Error 481: Invalid Picture.
- Fixed a problem that caused, in Windows Server 2019, the Console to show Error 481: Invalid Picture.
- The times to execute the Win32 terminal and console have been reduced by reducing the size of the exe files by approx. 50%.
- Press CTRL+F5 in all internet explorers: Fixed a bug that was delaying the web report dashboards. This was especially noticeable when there were many dashboards.
- Improves how the Central service processes auto changes of statuses and therefore it is

more capable of processing monitor messages.

- New option "Add threshold" in the Monitor, Results tab, allows to add the effect of the threshold to the Cycle Time when considering production. This should reduce microstops when the data did not arrive on time or when the monitor service was down for a while.
- Now the terminal loads I/O only of the current shift.
- Fixed a bug in the updater of the win32 terminal which was aborting the update when the font could not be updated.
- Fixed a bug that was preventing the screensaver to automatically refresh the dashboard when there was only 1 dashboard.
- Now actual starting and ending time of orders cannot be in the future.
- Fixes a bug that was duplicating the monitoring items when copying & pasting a process from the Behaviour tab.
- Fixed a problem that was changing the decimal point in some computers when editing a working order.
- Fixed a problem that was scheduling working orders in the same period and process.
- Fixed a bug that was not showing the touch keyboard on certain fields.
- Fixed issue in the date of creation of an scheduled order.

2019-10:

- Now a Measurement Program can be created without selecting any result. This is useful for IIoT data capturing.
- A new type of signal (Measurement) allows to capture data without relating it to a result. This is useful for IIoT data capturing.
- Added a new Recommendations report in Favorites section to provide data diagnosis.
- RPM Relations. Now the program can be deactivated so as not to request your measurements.
- RPM Relations. Now the measures can ignore different alerts.
- Now the Control graphic can be also added to favorites.
- Now all Quality SPC Alerts are marked as Expired when the result changes.
- Technology has been improved: pages are shown normally a 10% faster and consuming 15% less memory.
- Base line combo has been removed from the scheduling form as it was not being used by any user.

- Now I/O are not loaded filtering by the Date & Time, but only filtering by the parent Id.
- Enables to set timeout to 0 (no time out error returned) for requests to the API
- Changes in Client GetOEES and Reports OEERead funcitons to use from API. GetOEES returns precalculated ratios only for one of the up to 5 predefined periods and OEERead returns ratios given a start and end date and time.
- New option check in the Console\Reports allows to show or hide the web configuration wizard. By default, the wizard will be hidden for current installations. It will only be activated for new companies.
- Fixes a bug occuring when 2 measures were shown in the control graphic.
- Improved performance of the system by adding indexes to m2reca (id_refa) and m2vico_h (completed)
- Fixed issue in GetPersonsInMachines. Prevent 0 seconds periods to be returned.
- Fixed a bug that was causing the dashboard Copy&Paste and creation to fail.
- Now the screensaver can be activated without having to select the Web Dashboard as the default visualization in the terminal.
- Now the Histogram report can be added to Favorites.
- Functionality added in Control report: Now is possible to select more than one measure.
- New status class for Breaks.
- Now, when configuring a new resource, the terminal indicates if a field is needed because of any specific function. This is to avoid some confusion that was ocurring to users.
- Now, when configuring PSR relations, in the results bottom part, if a bad result is selected, all the fields which are not relevant are greyed and blocked.
- Fixes a bug that was not refreshing the statuses when a new one was created by the operator with the Allow Configuration feature.
- New Advanced Planning and Schedulling functionalities.
- New Theme option in Devices Configuration. This allows the dark mode.
- Fixes a bug that was not showing the type of task in the Autocontrol Efficiency report.
- Fixes a bug that was not considering maximum time of statuses when more than 1 of those statuses were contiguous and the maximum time was configured in the status and not in the process-status relation.
- Now when a counter decreases without a reset value, the system will consider its new value instead of zero as it was doing.

- Fixes a bug in the Status Summary and Evolution reports which caused that when not selecting any top, was showing only the first status.
- Fixes a bug when copying the link of a widget from a dashboard by using the copy icon.
- B2MML ISA-95 related documents have been added to the API Guide.
- Now the New results records are manually inserted at the latest date and time of the system.
- Fixed a bug that could cause the server to get in an inifinite loop when generating auto changes of statuses.
- Now, the monitor service, for signals of type Measure and Measure_Res, if the size of the subgroup is X, will wait until having X measures before inserting them, and in that case, all those measures will belong to the same subgroup.
- Now the date and time when entering a new SPC Data is always the current date and time of the terminal and it can only be changed by personnel with the Multishift role.
- Now if the request to automatically update the terminal fails, the terminal will open the web document that explains how to update manually.
- New edinnSIM_1 monitoring method which allows to monitor simulated factories.
- "Remeber me" check in the login form.

2019-01:

- Now the widgets do not belong to resources, but to dashboards instead.
- Now warns when the 'Restart them and monitor' of the Supervisor tab is not correctly configured, as it was preventing the Supervisor from restarting the OPC Server exe process.
- Big Data calculations have been improved by eliminating the period where data can be shown as zero and by increasing stability.
- Corrected an issue which caused the Follow-up report to show 0 for rework and scrap when grouped by result.
- Now, it is more precise when determining if an autocontrol task is not done, as it considers the CycleT and CycleQ from the PER relation and not from the process.
- The report of Autocontrol\Schedule has been speeded up.
- Fixes a bug that was causing that a report cached in a language could be shown incorrectly in other languages. Now the caching is language dependent.
- In the terminal, when you approved an Autocontrol task, it also changed the status. Now both things have been separated.

- Fixes a bug in the manual introduction of consumption.
- Fixes a bug that was showing less than the shift time in the hourly production API dashboard.
- To increase the system performance, it is recommended to set the MySQL innodb_thread_concurrency parameter to 0, which is its default. For more information, please read: https://dev.mysql.com/doc/refman/5.7/en/innodb-performance-thread_concurrency.html
- There is a new type of autocontrol Task: Quality
- Fixes a bug that was not showing a configured Team when it had no member assigned.
- Now you can configure a device so that the terminal will pre-select the team of the user when entering into the Autocontrol window.
- Now the terminal only launches its update when it detects that its build is lower than the one of the system (indicated in the database).
- Now automatically deletes the consumptions associated with a Process-Status relation when this relation is deleted.
- Now there is an audit table to store changes in autocontrol configuration that can be consulted from autocontrol configuration window
- Fixes a bug that allowed users to delete certain statuses because it was considering it as a modification.
- Now an autocontrol task can be restricted so that only the members of its team can change its status.
- Now the value of a signal (Measure type of signal) can be automatically and natively inserted into a Program an Measure.
- Now you can obtain consumption and cost by unit produced in consumption summary report
- Fixes a bug that was showing, in the report Panel, an error in the decimals.
- Now it does not generate status auto changes messages in the registry when they are referring to the future statuses.
- There is a new type of event in the registry which reflects auto changes of statuses which are due to maximum time reached.
- Function CalcOEE: Works with Speed and Quality ratios higher than 100% in Area calculation.
- Function Normalization: Works with Speed and Quality ratios higher than 100% in Area calculation.

- Fixes a bug that was causing some URLs to not be launched.
- Path. Scrap points were misplaced.
- SPC Control. Y scale was set to zero for periods before the first mesarure
- Session Lifetime should be changed to 86400 (for example) instead of the 0, as it has been detected that 0 is not working in PHP anymore.
- Fixes a bug in the SPC Locate window that was clearing the SPC Control and Histogram graphis when clicking on the graphic.
- Now the CTRL key can be used in the Results window to select a range of records.
- Fixed issue in cons limits calculation.
- Added vbWriteToFile compilation parameter.
- Added accumulated line to status summary report.
- Fixed issue in report Events.
- Removed unused parametres.
- Fixed issue in shift selector. Added gotoFollowUp function. Fixed issue in Product/Result selector.
- Recommendations report.
- Added shift selector in followup report.
- Added operator window to recomendations report.
- Make function LineGetChildrenMultiLevel public.
- Fixed issue in stat.
- Improvements in Recomendations and FollowUp reports.
- Fixed issue when trying to add or delete results from time product form.
- Enables filtering followup report by operator.
- Added tooltips for integration report.
- Fixed issue with decimal symbol.
- Fixed issue in events report.
- Added a function toogleShiftGen to toogle the shift combo un followup report.
- Moves Diagnosys report to Favorites menu. Added a function toogleShiftGen to toogle the shift combo un followup report.

- Fixed issue in Recommendations report styles.
- Enables to set timeout to 0 (no time out error returned) for requests to the API
- Changes in Client GetOEES and Reports OEERead funcitons to use from API. GetOEES returns precalculated ratios only for one of the up to 5 predefined periods and OEERead returns ratios given a start and end date and time.
- New option check in the Console\Reports allows to show or hide the web configuration wizard. By default, the wizard will be hidden for current installations. It will only be activated for new companies.
- Fixes a bug occuring when 2 measures were shown in the control graphic.
- Improved performance of the system by adding indexes to m2reca (id_refa) and m2vico_h (completed).
- Fixed issue in GetPersonsInMachines. Prevent 0 seconds periods to be returned.
- Fixed a bug that was causing the dashboard Copy&Paste and creation to fail.
- Now the screensaver can be activated without having to select the Web Dashboard as the default visualization in the terminal.
- Now the Histogram report can be added to Favorites.
- Functionality added in Control report: Now is possible to select more than one measure.
- New status class for Breaks.
- Now, when configuring a new resource, the terminal indicates if a field is needed because of any specific function. This is to avoid some confusion that was ocurring to users.
- Now, when configuring PSR relations, in the results bottom part, if a bad result is selected, all the fields which are not relevant are greyed and blocked.
- Fixes a bug that was not refreshing the statuses when a new one was created by the operator with the Allow Configuration feature.
- New Advanced Planning and Schedulling functionalities.
- New Theme option in Devices Configuration. This allows the dark mode.
- Fixes a bug that was not showing the type of task in the Autocontrol Efficiency report.
- Fixes a bug that was not considering maximum time of statuses when more than 1 of those statuses were contiguous and the maximum time was configured in the status and not in the process-status relation.

2018-52:

- Allows to show the web dashboards also in the terminals.

- Now certain statuses can be excluded from the prediction of failures in the Calculations\Envision section in the Console.
- Improvements in the integration report selector.
- Fixes an issue when cliking in the SPC control graphic of the main form to see specifications.
- In the Fields configuration, the I/O field has been converted to Apply to, changed it order and its default value.
- Now, when an obsolete device connects to the server, it warns in the log indicating also the device name.
- Now Recovery from slave to master server is automatic.
- Now the fully functional terminal supports more device resolutions.
- In Productivity report you can: select up to two fields to classify and filter the rows, you can group rows by order or preserve operations and you can remove totalization.
- Now is possible to export favorite reports to CSV automatically.
- Fixed a bug that was causing problems in the population of the database table of historic changes to status of working orders. If your control.DB_Upd is minor than 20190228, then you require to execute a database conversion tool named **edinnM2_DB201851to52Conversor.exe**, which is available together with the do.sql file.
- Now it allows to request a quality measurement every time a process is stopped for more than a certain time.
- Server can find inconsistences between scheduled orders and production records: records without schduled order and records assigned to a working order different than expected.
- Terminal can fix inconsistences between scheduled orders and production records.
- Fixes a bug that was showing alerts of needed justifications, in the main window, in the tree, to users which were not supervisors and could not justify them.
- Fixes a bug that was showing, in the statuses window, the button to justify previous stops, to users which were not supervisors and could not justify them.
- Terminal was warning of maximum operators in a process when the maximum numbers of operators was configured zero in the process.
- In the Activity Report, a bug was causing certain statuses to not appear.
- The system has become more robust by avoiding undesired effects of manually inserted results with future dates.
- Fixed a bug that was causing inconsistences in personnel in/out table, when an in/out

record was modified from two devices at the same time.

- Some of the requested parameters and responses of the edinn API have been changed for normalization, as follows:
  - GET /statuses/config Id_Tu -> id
  - GET /statuses/config Tu_Level -> level
  - GET /statuses/config wdays -> week_days
  - POST /statuses/config TuTYpe -> type
  - POST /statuses/config id_tunext -> next_status
  - POST /statuses/config tuclass -> class
  - PATCH /statuses/{id} Id_tu -> status
  - GET /statuses/summary StatusType -> type
  - POST /results/config prodtype -> type
  - GET /results/config/{result}/relatedresults prodtype -> type
  - GET /results/summary prodtype -> type
  - POST /processes id_calc -> calculation
  - POST /processes id_unit -> unit
  - POST /processes machnext -> next_processes
  - POST /processes id_sttu -> microstop
  - POST /processes processtype -> type
  - POST /processes/{process}/statuses nextstatus -> next_status
  - POST /processes/{process}/statuses weekdays -> week_days
- New and useful blocking options in statuses configuration.
- Failure prediction with neural networks (envision) has been improved by allowing administrators to adjust parameters for more successful predictions.
- Classes of statuses have been improved and explained better. Class Indirect has been removed as it was a combination of other classes. All statuses classified as Indirect have been reclassified to Predictive Maintenance.
- Now, when you change the Id of a process, it also changes your favourite reports so that they keep working.
- Now, when you change the Id of an area, it also changes your favourite reports so that

they keep working.

- New useful screen saver that will show the web dashboards of the processes associated with every device.
- Now the calendar does not need 2 clicks.
- Alerts of Quality were blinking about the previous product.
- A screensaver with your favourite dashboards can be shown in all operation terminals. This multiplies the possibilities of the visual plant.
- The (R) symbol has been removed from the title of the apps because it was not readable in asian languages.
- A new licensing system forces all edinn M2 servers to be relicensed.
- Now the server can be updated automatically in the Console, Behavior tab.
- Fixes an error in the terminal that caused the combo of machines in the main form to show the Id of the previous process.
- Fixes an error in the terminal that happened when a user had access to areas which are not root areas.
- Fixes an error in the terminal which opened the Statuses or Results windows with a time frame longer than a shift.
- Now the system can optionally accept speed higher than 100%.
- Fixes an error which prevented multiple pasting of fields in results configuration.
- Fixes an error that was duplicating fields when in Fields Configuration.
- Now the customizable data can be shown in the reports, in Analysis, Panel, Data, and then in the terminal through the reports dashboards.
- Now the customizable data can be shown in the reports, in Analysis, Panel, Data, and then in the terminal through the reports dashboards. Therefore, the Data visualization has been removed from the terminal, as it can be shown through the reports' dashboards in the terminal.
- Now there is a new functionality to check if shifts are correctly configured.
- Pop up messages behaviour has been improved.
- Refresh of main form has been improved.
- Fixed a bug that was showing incorrect information when selecting a team in the Panel and other reports.
- Detection of result change has been improved for the case when users change result without generating a zero quantity record.

- Now the system will use Big Data when it is activated. Therefore, please be sure to prepare Big Data so that it covers all your period of data.
- Reports now will use the Big Data & AI Module, when it is activated in the Calculations tab of the server console, to show the information much faster, in some cases up to a 10% or less of the original time.
- Now, users with the Common Favorite role, can see and delete favorite reports of other users to release work load of the server. A message will be sent automatically to the user.
- Now the terminal warns, from the window to edit working orders, when the selected route has no steps.
- Now old API sessions are purged based on the real time days.
- Now the filter of the statuses configuration has been renamed to Select, as it was not filtering but selecting the status. Its functionality has been also validated.
- Dashboard (API) of hourly production has been optimized in the path graphic area.
- A bug has been fixed in exceptions of statuses: it prevented from creating, for the same status, 2 or more exceptions for different statuses.
- Now warns when the 'Restart them and monitor' of the Supervisor tab is not correctly configured, as it was preventing the Supervisor from restarting the OPC Server exe process.
- Big Data calculations have been improved by eliminating the period where data can be shown as zero and by increasing stability.
- Corrected an issue which caused the Follow-up report to show 0 for rework and scrap when grouped by result.
- The report of Autocontrol\Schedule has been speeded up.

2018-51:

- The wizard for the creation of companies has been improved.
- Calculations of ratios related to areas of areas has been corrected and improved both in the terminal and in the reports.
- Client User needs to be resetted through the Console, General, DB Configuration, Update Client button, or terminals (clients) will not connect to the server.
- Now the SrvPort of the comp table has prevalence over the BasePort on the Console, General Tab. In fact, the first will overwrite the second.
- Now the terminal synchronizes with the Server when changing result in order to avoid a problem that happened very unfrequently which caused the previous result to become the actual.

- Recovery process and procedure has been deeply tested and improved.
- Now, when a new order is started, the server does not automatically pause previous working orders. Only the terminal pauses automatically the previous working order.
- Now fields can be required when starting a working order.
- Now the server can send reports in PDF format to users. This needs to be configured in the Console and then the user should mark it in his favorites.
- New report Registry\Diagnosis shows if the company is using correctly and frequently the edinn platform.
- Now results which are not assigned (PN field) to a working order will not be added to the produced quantity of that order.
- In the scheduling tables, Estimated, Setup and Teardown times are shown in format HH:MM:SS.
- Tag Speed1 has been deprecated from data customization in the main form of the terminal.
- The path shows now the code and description of the statuses, which of these (in the case of failures) have been predicted and the code and description of results.
- Fixes a bug that was causing that the system did not use the latest productive status.
- Fixes a bug that permitted a user who was not supervisor, to view statuses and results of a process which was outside of those filtered by device.
- Now prediction of failures (Envision) considers time in minutes to analyze failures, instead of a fixed number of failures. Please update the "Consider" field in the Calculations\Envision section, or predictions will not work correctly.
- Fixes a bug when copying programs and measures in the RPM (SPC) configuration.
- Improvement in reports which reduces up to a 58% the total time to show the majority of the reports.
- Fixes a bug that was preventing report Result\Evolution to stop
- Report Result\Evolution has been speeded up by removing the option 'High Precision' as was not providing difference and was delaying the report.
- Reports have been speeded up, in some cases dramatically (up to 10% of the previously needed time).
- Fixes a bug that was causing incorrect sorting in the statuses list of PSR relations.

2018-50:

- The option of the Console, Monitor: Need 2 results, has been deprecated (not in use

anymore).

- Some bugs on the Big Data generation have been corrected.
- Now results can be generated every hour (o' clock).
- Now you can create a new company in a new (and empty) server.
- Now you can configure activity percentaje for Process-Status relationship.
- Now there is a new 'Expired' status in autocontrol. Tasks which are in status Not Done, will automatically pass to status Expired, when a new task of the same Id is created.
- Now the terminal can show video on streaming coming from cameras, to help to supervise machines and areas of machines.
- New background colours have been applied for a more comfortable user experience.
- A new layout, skin and colours have been applied to the web reports tool for a more comfortable user experience.
- Now the report Analysis-Follow-up can be stopped correctly.
- Now you can choose between path and speed in results->path report
- Now the alerts in the login screen are only related to the users which do not have multishift role.
- In the main window, now if the processes of an area have alerts, then the function buttons will also blink, and if you click them, they will show you the first process corresponding to that alert.
- Now the resources can be configured with their associated decimal symbol.
- The Envision (predictions of failure using neural network) has been improved and tested.
- Now the server can keep running any executables which are specified in the Supervisor section.
- Fixed an error when manually sorting statuses in configuration.
- Fixes a bug which generated a new FAI after manually justifying a status.
- Allows to send labor time information aggregated or by operator.

7.0.6:

- Fixes a bug that prevented to work the filter in the status and results selector screen.
- Fixes a bug that was creating empty (qt = 0) results. This bug was related to total counters exceeding their reset value. Now reset value is recommended to be 0 or large in the case of total counters.

- Now statuses generated from the monitor respect blocks.
- Fixes a bug that occured when the monitor restarted and the justify_all status was lost.
- Now a manual insertion of a new status. when propagating the status, ignores blocking statuses in the next processes.
- Recovery functionality has been improved.
- Now an Operator Time can be configured in the process and PSR relations. This is the time in seconds from the Cycle Time which are only man work.
- Now the area for the information of the machine on the main window has been increased a 20%.
- Now customized and powerful dashboards can be created and shown in the main window.
- Fixes an error which created a status a few seconds after of a shift change.
- Now SPC measure its possible to set on result change, order change or both
- Now in Quality-> Summary appears 2 new columns with control limits
- We remind our users that Windows XP and Windows Server 2008 are no longer supported by edinn.
- Now when an status is automatically changed because of its max. time, the system is also considering previous statuses of the same type.
- The "Mandatory" option for the configuration of fields has been changed to "Ask", for a clearer understanding.
- Now autocontrol tasks that were performed out of the tolerance period are not considered as not done.
- In the autocontrol window, now autocontrol tasks appear in orange color in all the row and not only the cell with the date and time.
- Now the system warns, by email and in the windows events viewer, when it is running out of disk space. This is a very important warning as the system cannot longer work when it has run out of disk space.
- There is a new report, Quality/Alerts.
- Now in Statuses and PSR Relations, the Next Status has been changed to default FAI-Failure, as this status was the default when this field was blank. This is, blank and FAI-Failure meant the same. By this, users are not confused whether if they have to select blank or the FAI-Failure status. Leaving FAI-Failure has no effect when maximum time has not been indicated.
- Now you can copy and paste all the configuration from a process into another process,

rename a process or delete a process and all its data. Use these features carefully: it is recommended to backup all data before using them.

- Added funcionallity to indicate when SPC alters belongs to the current or a previous shift.
- Now autocontrol tasks can be managed in the form, which allows for step-by-step and faster management.
- Now autocontrol tasks can be associated to new results.
- A bug about the reescalation of the images being shown in the main form has been solved.
- Now autocontrol tasks can include an image.
- Fixed a bug that was flashing the button to justify from previous shifts to operators who had not permission to justify from previous shifts.
- Fixed a bug that was happening when modifying an area when there was only 1 area.
- Fixed a bug that generated future statuses when there was a status with maximum time and start time and end time where equal and the seconds where 0.
- Now, a representant process will not generate emails of "Not justified" statuses.
- Fixed issue in autocontrol reports.
- Fixed issue when loading tasks in autocontrol form.
- Fixed issue in process quantity calculation.
- Automatic ticketing user creation.
- For better understanding and faster loading, basic buttons like insert, modify, delete, accept, apply, close, cancel, etc. have been changed to icons.
- The quantities produced have 2 decimals and the percentages of good production, rework and scrap refer to the actual total production.
- Fixed a bug that was not showing big scroll bars in the Statuses window.
- Added a new Working Order filter in Autocontrol. Now it is possible to filter by the working order both in the terminal and the reports.
- New autocontrol configuration functionality. Now, in case of the "daily" tasks, their termination will be taken into account, instead of their due date, unless they are marked to be scheduled for any day(s) of the week.
- Now a customizable dashboard can be shown in the terminal. A default dashboard shows hourly production above of the path when the Big Data & ML module is licensed.

7.0.5:

- Italian is now supported.
- Ponderation of production after when a PLC reconnects after being disconnected or down for a long time, has been made optional in the Console, Monitor, Results tabs.
- Added scrap and rework in the Analysis reports.
- Now you can copy and paste fields between results and if they are repeated, only the new will be copied.
- Now the deadband can be assigned independently for each signal of the monitor.
- Now counters are also written into the monitor log for tracing purposes.
- Performance of server (monitor, central, reports, etc.) has been increased by improving the function which obtains the actual result.
- Constants of SPC Graphics and calculations have been revised.
- Now bad production depending on Goods & Totals can be also dumped because of a product or order change, and not because of the dischargement of the process.
- Fixed a bug when goods counters passed their reset value.
- Fixed a bug when creating new personnel (relating with the AlertsType field).
- Now when a user is deleted, his favorite private reports are also deleted.
- Now a manual justification becomes a dependency for connected (Next, or processes which have this process as Next) processes when the status has been configured as propagating in the PSR relations.
- The OPC Drivers have been improved in accordance with the latest version of OPC UA.
- Now autocontrol tasks, when they have been not launched for a long period (because the production minutes which are mandatory had not happened for example after a weekend), they are launched very close to the tolerance (-) and not up to 240 min. before, as it was happening until now.
- Now you can assign devices, resources and areas to the process in configuration>processes
- Fixes a bug which was not deleting statuses FAIs, DEP and UNS with 0 secs at the beginning of the shift.
- New field called Delay in the Monitor configuration, applicable to signals, which allows to check the status of the signal not now, but previously considering "Delay" seconds.

7.0.4:

- Support for multi-language in Chinese and Slovakian for justifications and other descriptions.
- The platform can now notify users when the OEE or any of its components are below and above the targets.
- The default statuses like TU_PRODUCTION, TU_FAILURE, etc. must always be in the database. This speeds up all the system.
- Comments in autocontrol have been truncated to 341. Please manage update not to lose any data. Description of task has been increased to 341 characters.
- The resources windows (personnel) allows to obtain the link to auto login on the reports.
- The reports show color grey instead of green, when all the period is assigned to unscheduled.
- The dashboard (home from the reports) of a user can be copied and pasted into another user.
- Notification by email when any ratio (OEE, Availability, Speed or Quality) is below the yellow target in any of the defined periods.
- 100% of OEE due to unavailability of the process is shown as N/A and in grey color (as unscheduled).
- The Not Recorded report also detects when the start of a status does not match with the end plus the duration of the previous.
- Now in the Pie report, you can access to the detail of the OEE represented graphically with donuts.

7.0.3:

- In the OPC Generic monitoring driver, now signals are processed in separated threads. This minimizes or eliminates the possibility to lose signals and speeds up their processing.
- In the Console, Monitor tab, 2 new parameters allow better capturing of signals: deadband, which allows to filter and not capture signals which change less than a certain %; and Times, to capture all signals every certain time.
- Signals can be sent to the Registry in the form of events, natively from the monitoring driver, which increases performance, robustness and releases the UDL to other tasks.
- The value of the signals that were active at a certain time are now easily seen with the new "Last" check of the Registry window, which allows to locate the last values of the selected signals at a certain moment.
- The Locate window has been improved to allow locating production with certain values in

their fields, which is useful for e-Tracking functionality.

- If you associate a new process to an area, all persons who have access to the area, have access to that process.
- You can delete an area without having to delete the relations between that area and persons and devices.
- The monitor divides production quantities and production statuses between the shifts when it could not monitor production during a period which covers multiple shifts.
- The standard UDL in VB.NET has been updated and now it includes a Terminate public method that must be added to all UDLs. This method allows the UDL to close all its resources (database connections, files, etc.) safely.
- There are new 'Button' fields which allow customers to develop any specific need to be executed from the terminal.
- All manual deletions of records are registered both in the logs of the server (Console) and in the Registry of the Terminal.
- The ReportsCache accepts 2 new optional parameters: the first indicates the list of reports to be cached and the second the list of reports to be excluded. This allows to configure the system so that 1 report can not delay the caching of other important ones.
- New improved management for the Teams Auto Change functionality.
- The terminal updates languages when it detects that a new build is being executed.
- The console indicates in the title of the log window the content of the field Contains. This allows to manage multiple log windows and to be able to easier switch between them.
- The server will warn when a terminal has a different version or lower build.
- Added new parameter (IgnCheck) in measure definition table (m2meas) in order to ignore some SPC checks. By setting IgnCheck to 1, ignores 7 measures above or bellow the nominal.
- It requests if same characteristics of previous result even when the result Id is different.
- When you change to a new product, you can inherit the fields per Variable coincidence.
- Allows to launch autocontrol tasks for product and/or order changes.
- New automatic signal conversion into events in the monitor.
- Fields can inherit values from previous fields of production records, or to not inherit, and this is configurable in the Configure Fields form.
- The monitor service start has been speeded up dramatically.

- Several bugs fixes and optimizations.

7.0.0, 7.0.1 & 7.0.2:

- Several bug fixes and optimizations.
- In the edinn® M2 Server, now the most updated data can be kept in memory, instead of in the hard disk, making the system faster and needing less writing to disk. Please contact your system Administrator to activate this feature.
- Reporting web tool:
    - There are 2 new reports in Analysis, Panel: Complete and Evolution. These new reports are panels to analyze the complete status of Areas and Processes.
    - Now reports stop better when the ESCape key is typed.
    - The ReportsCache task has been optimized to avoid consuming too much resources from servers.
    - In the Reporting web tool, there is a new Presentation mode in the Favorites, Home section, which shows the Home dashboard maximized and auto refreshes. This feature has been specifically designed to show the Home (dashboard) of the Reports on big screens.
    - Now reports show the selector of Areas and Processes in the form of a tree on the left side of the reporting tool. This tree can be hide for a better visualization and for printing.
- Terminal Application:
    - There is a new layout of the main form, more intuitive, easier to use and with at least 40% more area for visualization.
    - Now alerts are shown with icons, which make them clearer and language independent.
    - The graphic which shows processes in the main window has been improved: keeps aspect ratio, has a better visualization of ratios and the name of the process and a better management of the image (upload and deletion).
    - Now the Locate window allows to search for a specifi order.
    - Now operators can be warned when their manual insertion of results exceeds or is below of a certain % with respect to the target for that moment.
    - Now fields can be automatically calculated by using a development module. This gives full power to fields.
    - The main form also shows areas and processes in a powerful Tree format.

  - The 'Path' graphic has been speeded up and consumes less resources from the system.
  - The terminal supports Single Sign-On functionality, which allows the user to automatically log-in.
  - The same Status can be scheduled for different days and times, configured in the PSR configuration.
- monitoring:
  - New signals OEE.R, OEE.G y OEE.Y allow to light on beacons to indicate that the OEE of a process is below (OEE.R), above (OEE.G) or around (OEE.Y) the OEE objective.
  - New signals (Justify and Justify_A) allow to justify the active stop of a process.
- Scheduling module:
  - The terminal now supports Gantt visualization and editing, with colors indicating the status of the orders.
  - Now new orders can be automatically generated according to the route of the order.
  - Routes of orders have been improved.
  - Scheduling capabilities have been dramatically improved.
  - The system is capable of importing MS Project 2010 files in XML format and creates all the necessary structure (areas, processes, resources, etc.) to manage the project.
  - Now operations (suborders) can be managed, including Setup, CycleT, CycleQ and Teardown times.
  - Now tasks (orders) can be linked (predecessors).
  - Now CycleT and CycleQ are first taken from the order, and then from the rest of places (Fields and then PER Relations).

6.1.0:

- SPC graphics have been migrated to HTML5 technology.
- In order to improve performance and security, the following components of the edinn® M2 system have been upgraded (a procedure document must be followed):
  - MySQL from version 5.5.8 to 5.5.44

  - PHP from version 5.3.X to 5.6.8
  - Apache from version 2.2.X to 2.4.12

- New ResChange signal in monitor. This signal allows to automatically change the result of a process.
- New behaviour of signals AvailProd and Saturation. Now, when they cease, the dependence status is also ceased and turned into a FAIlure.
- There is a new parameter 'Request time for Bad Results' in the Console, in Behaviour/Optimization, which allows to skip the requirement of the time when inserting bad results manually.
- Fixed various bugs in the Ratios Report.
- Now the FTE report considers only the AT ("Attendance Time"), instead of the Total Time. By this, if a worker has been working in 2 processes at the same time, the AT will only consider the time of 1 process, as the time of the worker is considered to be share at a 50% in this case. This time matches the last column of the Activity Report, so the Activity Report is now the detail of the FTE report.
- Now orders are updated in real time, which is much faster.
- The total update of the quantities of orders can be performed manually from a new button in View Schedule.
- A new parameter on the server called "Request good quantity on task" allows to the user to enter the good quantity that he has produced and therefore an automatic insertion of bad production will adjust the total good quantity produced for the task (order).
- Now the Activity report provides information about the monetary incentives to be given to the workers. This feature is only available with the Consumption and Costs modules.
- Performance has been dramatically improved by caching data in memory. Please check the parameter max_heap_table_size in the MySQL.INI to setup the maximum available memory.
- The internal code of Time Usage for Dependence has changed from 'LIN' to 'DEP'. The database update script performs and update of LIN to become DEP, and therefore it is necessary to check if the client has already used DEP as a Time Usage code. In this case, the DEP code must be replaced to a different one before upgrading to 6.1.0.
- The BarCode module has been retired as a separated module and its functionality has been incorporated inside the system core. The server needs to be relicensed.
- The new modules Personnel and Stock are now available. The server needs to be relicensed.

- The Autocontrol algorithm has been improved to prevent that a task, scheduled upon production quantity or minutes, is launched repeatedly when it has been scheduled for the future and performed in the past.
- Now there is the possibility to request to the worker the good quantity produced at the end of the shift, and the bad quantity will be deduced and inserted automatically.
- Now the "Ref" button allows changing I/O stock smoothly.
- The Order field has been included as a filter and as a valid information in consumption.
- The In&Out form has been improved allowing In and Out of complete areas.
- There are new roles (for example View Schedule): please assign the new roles to the users.
- UDL might need the new Process call declaration.
- There might be the need to copy the dll MSVCRT110.DLL into Windows\System32.
- In the Server Console, in the Reports tab, in the Session Lifetime field, now 0 is not a valid number to indicate that the session must never be auto closed. Please indicate the number of seconds that you wish to allow the user before closing his session.
- New report Analysis Panel which shows in the same page the main data and graphics of a process.
- New possibility to expand the graphics area in the terminal application.
- New possibility to view the graphic of the OEE and its components in the Ratios option in the main form of the terminal application.
- Now the variables or processes in all the formulas must be enclosed with quotation marks. Please see the configuration or Areas and Calculation Types.
- Now Autocontrol can be managed massively by clicking the multiselect option and managing multiple tasks at the same time.
- Now Autocontrol fields are updated without the need to click in the button Mod(ify).

6.0.3:

- Now emails sent are filtered by process and person, so that it does not send information to a user who has not access to a certain process.
- The monitor now can auto justify failures according to the "Justify" signals.
- Now the id of status can have 4 characters.

- Now the terminal speeds up the loading of the working forms, including the login.
- Now SPC data capture shows highlighted with colors the measure.
- New Analysis report call Productivity.
- New checks "Hours" in reports with times.
- New checks "Kg" in reports with result quantities.

6.0.2:

- A new email now can be scheduled on the server to be received periodically, indicating the summary of the status of all the alerts in processes.
- Now processes can be shown ordered by a different criteria than their id.
- Now, when the server is in very high detail, it keeps track of users changing time usages.
- The performance of the reports have been improved.
- The confirmation question when accepting the SPC data capture has been removed as it was unnecessary. Now, instead, the Accept button blinks in yellow to indicate that the user is about to introduce the data and that there will be no confirmation.
- The SPC data capture form has been improved visually.
- Now there is multiple selection in the Locate SPC data form, which allows to easily perform multiple deletion of records.
- Now the user can take several SPC data without the need of exiting the SPC form, by using the Number of Measures field
- Now the user can copy and past all SPC programs and measures from one result to another in the Programs and Measures configuration
- SPC Control and Histogram reports have been splitted in 2 different reports.
- New SPC Program type 'By All Measures'.
- New SPC report call Summary.

6.0.1:

- Now there can be multiple shifts and they can be associated to individual processes using exceptions.

- Now the Next status of a process is also subject to exceptions.
- Now available a new monitoring method: edinn OPC Bridge.

6.0.0:

- The terminal has been speeded up for better operation on the cloud.
- New graphical icons in the reports application.
- If a process is limited to work only scheduled, working orders will not accept status changes unless all pending FAIlures have been justified.
- Improvements in the blinking of buttons to require time usages justifications.
- New button -> in the window to justify status, allowing to go to the next pending status.
- Now the access to processes and areas can be restricted to users, and even to administrators. This is useful to manage different plants with the same system, by configuring the different plants as different areas, and then assigning different administrators and users to each plant.
- A new role SUPERADMIN has been created in order to be able to access all the processes and areas, without restriction.
- From this version on, every time a new process or area is created, it must be manually assigned to users and devices.
- Improvements in the Autocons task and in the Consumption Efficiency Module.
- New URL monitoring method.
- New web Wizard for incredibly easy configuration.
- New features for "do the installation yourself".
- The order of the combo to select Manual or Automatic monitoring in Process configuration has changed. It needs revision process by process.
- In the terminal, in the main window, the alerts in processes are also shown in areas by aggregation.
- The Schedule button has been moved from configuration to the View Schedule screen.
- Some bugs regarding the update of start and end times in View Schedule have been fixed.

- Now the task bar also notifies about alerts (scheduling, results, statuses, etc.).
- Server has been improved in performance as now AutoChanges occur in a separate thread.
- Fixes a bug that was occuring when associating (Auto conf) a time usage to a process, that was not assigning MaxTime and Next Time Usage.
- Fixes a bug that the Central was not considering MaxTime when inserting a new Time Usage.
- Fixes a bug that the terminal was not respecting the Next time usage, when deleting a time usage and the previous one having MaxTime.

5.2.5:

- New 4 reports:
    - FTE, Efficiency, Notifications and Schedule.
- New improvements in reports:
    - New RGraph library version
    - Changes path canvas creation order
    - Graphic reports support HTML5 and Flash format
    - Schedule, shows production, and I/O
    - Shows SyncProductionSchedule messages in integration report
    - SPC Report - measure filters by process
    - Activity report show RT (real time or natural time) and TT (total time or enforceable) instead of only RT
    - FTE report show values based on TT
    - Shows production path in IE8
    - Adds a filter in report for hide NON PRODUCTIVE processes
    - Capable of printint IE HTML5 Graphs
    - Adds an option in Status for showing only lost time
- New integration possibilities:
    - Persons

    - Processes
    - Combos
    - Products (with fields)
    - Production orders (modify and delete)
- General improvements:
    - Adds MaxLength property for fields
    - Adds Equivalence FClass
    - Sets user for update to anonymous user
- Integration ISA-95
    - Adds the capability for Person property to store values and/or quantities
    - B2MML Client avoid to send ScheStatusChange in close period day
    - Adds Unit of Measure parameter in GetMaterialInformation message
    - Adds capability for sending fields into production performance messages
    - Adds capability for insert fields in a determinate position
    - Adds the capability for MaterialProperty to store values and/or quantities
    - Adds the capability of storing fields into the products. Enables insert and delete products and change products
    - Changes the time storage for production request to: first - segment time, second - personnel global time, third - personnel clock time x personnel quantity
    - Improves PendingDelete method adding more detailed error messages
    - Fixes a bug in ST function use in ScheUpdQt
    - Adds a function called DontNotify useful for opening day
- General improvements:
    - Improves OEE calculation for Lines/Areas
    - Enables recovering fields to send
- Bugs and internal:
    - Fixes a bug when Removing virtual stock records

  - Fixes a bug in In and Out management. Prevents unclosed in/out periods
  - Fixes a bug in filer by stock, when the field has no values, uses default value
  - Fixes a bug. when operator navigates into the fields collection, UofM field duplicate values
  - Fixes a bug in RefreshGrid method of Autocontrol form
  - In MTP2 creation check if date is provided
- New *Particular Data* tags.

5.2.0:

- The login to the terminal application has been speeded up.
- The path (© edinn) graphich has been corrected in a bug that occurred under certain specific conditions.
- Internal performance has been improved.
- Now SPC Control graphic allows viewing 2 variables together.
- A new variable **CycleTEquiv** in *particular data* shows the real cycle time.
- In results reports inputs and outputs are not considered for the total sum.
- Fixes a bug of the OPC Generic driver running out of handles with certain OPC Servers.

5.0.5:

- It needs UDL to be recompiled and reregistered with the object name: edinnM2UDL_COMPANYIDENTIFIER.dll
- FaconSvr driver has been finished and released.
- Now in the grids of the working windows (status, results, autocontrol, view schedule) the arrow keys can be used to navigate through the records and data fields.
- The user login process has been optimized and speeded up.
- Now the server needs an access to the Multi main database. This IP address and port must be specified at the General tab in the server console.
- Server needs to be relicensed.

5.0.4:

- Internal server performance has been improved arround 30% in most of the database

functions.

- Fixes various minor bugs.
- Fixes a problem that occured when the terminal time was different to the server time.
- Server needs relicensing.
- The "Quantity" graphic has been removed from the terminal.
- Fixes a problem in the blue dots of the "Speed" graphic.
- Fixes minor bugs on the default data grid on the terminal.
- The graphics in the main window have been converted to multithreading for better user response.
- Reports application webserver must be adjusted at the php.ini file:
    - max_execution_time = 0
    - max_input_time = -1
    - memory_limit = 256M
- Now the server accepts configuration of the SMTP port to send emails.
- Now allows that users' id start with a number.
- New bigger scroll bars make easier touch operation of grids and lists.
- Now the ON button to start the process is not restricted when the expected time of results has expired.
- For a more convenient alert management, the blinking in yellow of the Results button has been reduced to a 6th.
- Now, when rescheduling, the system shows also the paused orders in processes.
- In the Losses pie report, now it only says "Speed loss and microstops", in case microstops are not being automatically detected. Otherwise it says "Speed loss".
- Now certain monitoring errors are only warned to the log when they occur for more than the common monitoring timeout. This is useful to filter and to take into more consideration the error messages in the log and emails.
- Now when the web reports session expires, the session is visually closed, instead of showing an access error.
- Access to multiorganization web reports has been speeded up.
- Task bar has been improved, in performance and visually.

5.0.3:

- It allows to indicate expiration date to persons. After the expiration date, the person will not be shown in the operators list at the login window and will not be able to enter the reports application.
- It allows to automatically set the status of a process to the previous status to the actual. This is useful to, for example, justify a process to a certain status, the process enters in week-end mode, and on monday, it automatically returns to the previous status.
- A bug has been corrected that made incorrect the time field in the detailed results report.
- The exit application button has been set up in another position. The close button has been set up in its position, as it is more standard in the application.
- Fixed a bug on the proportional scroll bars in the reports application.
- Includes a new type of email, to notify persons, when failures in communication messages have occured between the edinn and other systems (ERP, etc.) and that they require manual intervention.

5.0.1:

- Resolves minor bugs derived from intensive use.
- Changes in PHP.ini: in ;extension=php_xsl.dll, remove the ; (uncomment the line).

5.0.0:

- Upgrading from version 4.4.3, 4.4.4 to 5.0.0 requires manual steps to be performed by a technician. Further upgrades can be performed by the edinn® M2 Server updater.
- New automatic update applies to terminals and now also to servers.
- New hardware has been created. This hardware is located by central edinn® M2 servers and servers gather all the data from them.
- New consumption reports. A new Panel report shows consumption in real time, allowing to turn on and off devices and see their consumption and cost.
- New visual aspect.
- Internal communication and timers control have been updated to new technology in order to improve speed and robustness.
- Database driver and engine has been updated to latest stable version.
- Multiple companies and edinn® M2 servers can run on the same computer.
- Reports and terminal allow access to multiples companies.

- Terminal has been optimized to be used on the internet.
- New ecloud services have been created. The application edinn® M2 can run on a varied environments on the cloud ecosystem.
- New synchronization functionality with ecloud.