

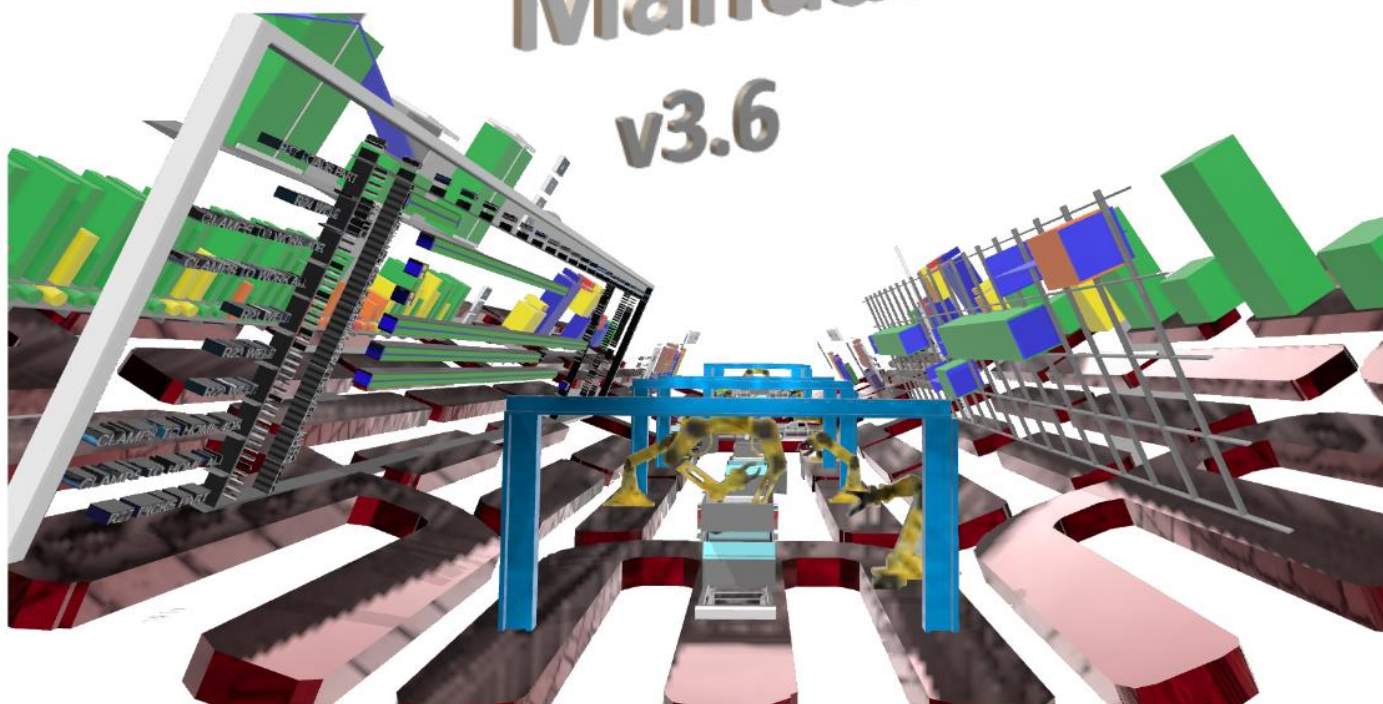
# BEET

# evision

## Administration

## Manual

v3.6





# Table of Contents

What's New ▾ .....	9
More Information ▾ .....	10
Introduction ▾ .....	12
Admin Module ▾ .....	13
Factory Tree .....	14
Factory window ▾ .....	15
Object Naming ▾ .....	16
Factory (PLANT).....	16
Area .....	16
Line .....	16
Station .....	16
Asset.....	16
Group .....	16
OPS .....	16
Object Overview ▾ .....	17
Factory .....	17
Area .....	17
Line .....	17
Station .....	17
Asset.....	17
Group .....	17
Tooling Group.....	17
Robot Motion Group:.....	18
OPS .....	18
Tooling Group OP .....	18
Robot Motion OP .....	18
WTM A / Bodyshop / 6X_S010_050 / S020/ S020FX1 / ADV PIN 1 / PIN 1 ADVD.....	18
Objects and Relationships ▾ .....	19
Building a Factory Tree ▾ .....	20
Definition of Asset, Group and OP objects.....	22
Assets .....	22
Overall Station Asset.....	22
Operator Asset .....	22
Robot Asset .....	22
Summary Asset .....	22
Asset States.....	23
Additional notes.....	23
Special Case Cycles.....	23
Groups.....	24



- START signal ..... 24
- STOP signal ..... 24
- OP signals ..... 24
- Examples: ..... 25
- Adding a Group and OPs ..... 26
  - START signal ..... 27
  - STOP signal ..... 27
  - OP signals ..... 28
- Importing ▾ ..... 29
  - Importing .enx file ..... 29
  - Importing SeqOfOp .json file ..... 30
- Deploy ..... 32
- Exporting ▾ ..... 34
- Configurations ▾ ..... 35
- Multiple Classification Update ..... 36
- Positions ..... 38
- DeviceWise Templates ..... 39
- Add Default Note ..... 40
- I/O Expressions ▾ ..... 41
- Code Explanation ..... 41
  - AssetStart ..... 41
  - AssetStarted ..... 41
  - AssetStartOS ..... 41
  - AssetRunning ..... 41
  - AssetDone ..... 41
  - AssetStop ..... 41
  - AssetStopOS ..... 41
  - AssetMem ..... 42
  - GroupStarted ..... 42
  - GroupStartOS ..... 42
  - GroupRunning ..... 42
  - GroupDone ..... 42
  - GroupOccurCount ..... 42
  - GroupStopOS ..... 42
  - GroupMem ..... 42
- enVision IO Expression Shortcuts ▾ ..... 43
  - Asset Status bits ..... 43
- Group Tag Explanation ▾ ..... 44
  - Add New Record ..... 45
  - Group Tag Definitions ..... 45
    - START ..... 45
    - STOP ..... 45



FAULTED.....	45
BLOCKED .....	46
CYCLETIME .....	46
SPARES (4-15).....	46
OPTRIGGER SPARE (1-16).....	46
Asset Tag Explanation ▾ .....	46
AUTO .....	46
MANUAL.....	46
FAULTED.....	46
BLOCKED .....	46
PRIMARY STARVED.....	46
RUNNING .....	46
IDLE .....	46
SECONDARY STARVED.....	46
INTERRUPTED.....	46
NON-PRODUCTION .....	46
USER 01-04.....	46
SPARE 11-27.....	46
enVision Application Server (EAS) ▾ .....	47
Components of EDC .....	47
Collector Engine ▾ .....	50
Adding a New Record.....	51
PLC ▾ .....	52
PLC Function.....	52
One Channel setup with Multiple PLC's (2 or More).....	53
Two Channels Setup with 1 PLC Each.....	53
PLC Devices ▾ .....	54
Overview .....	56
PLC Channel ▾ .....	56
Array.ANALOG.ArrayCount .....	57
Array.IO.ArrayCount.....	57
Array.OP.ArrayCount.....	58
Array.USER.ArrayCount.....	58
PATH.....	59
Program.Name .....	59
SLOT .....	59
Editing Asset Tags ▾ .....	60
PLC Tags .....	61
PLC Setup and Configuration.....	62
One Channel setup with Multiple PLC's.....	62
Two Channels Setup with 1 PLC .....	62
Models ▾ .....	63
Adding, Editing, and Deleting.....	64
Name.....	64
Part Code .....	64



Type.....	64
Priority .....	65
Baselines and Tolerances ▾ .....	66
Object Classifications .....	66
SmartNotes .....	66
Baseline Values .....	67
Tolerances .....	69
Exporting Baseline and Tolerances .....	72
User Maintenance ▾ .....	74
Create New User .....	74
Changes and Editing Profile .....	75
System ▾ .....	76
System Configuration.....	76
Warnings .....	76
State Display Settings.....	77
Object Classifications .....	78
Note Categories.....	79
Date and Time .....	80
Date Format .....	80
Time Format.....	80
Nomenclature .....	81
EDC Agent Settings.....	82
SMTP Settings .....	83
Others – Subscriptions and Default Dashboard.....	84
AUDIT ▾.....	85
Audit Date .....	85
User .....	85
Method.....	85
Entity ID.....	85
Comments.....	85
Data Changed.....	85
APPLICATION LOGS ▾ .....	87
Log History Selector .....	87
Descriptions .....	87
Subscriptions and Notifications .....	89
Subscriptions.....	89
Filter Bar.....	89
Search Bar .....	89
Notifications.....	90
Edit .....	90

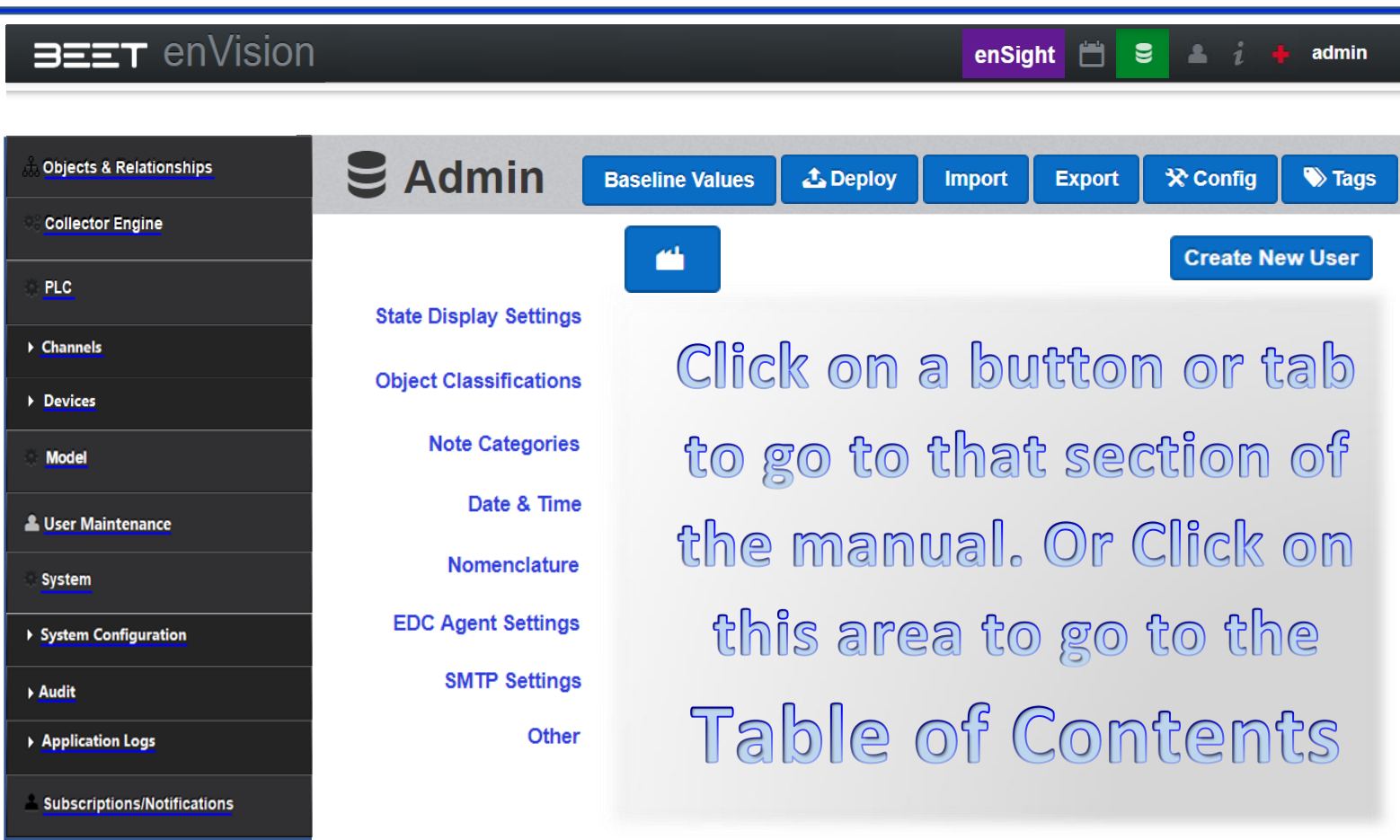


API Query .....	91
Glossary.....	92
Product Support.....	93
Trouble Shooting.....	93
License and Copyright .....	94

## [Sectional Quick Jump](#)

[Introduction](#)  
[Admin Module](#)  
[Object Naming](#)  
[Object Overview](#)  
[Objects and Relationships](#)  
[Building a Factory](#)  
[I/O Expressions](#)  
[enVision Application Server \(EAS\)](#)  
[Collector Engine \(EDC\)](#)  
[PLC](#)  
[Editing Asset Tags](#)  
[Models](#)  
[Baselines and Tolerances](#)  
[User Maintenance](#)  
[System](#)  
[AUDIT](#)  
[APPLICATION LOGS](#)  
[Subscriptions and Notifications](#)  
[API Query](#)

Click on the image to view a section.



The screenshot shows the BEET enVision Admin interface. The top navigation bar includes the BEET enVision logo, an enSight button, a calendar icon, a menu icon, a user profile icon, and the text 'admin'. The left sidebar contains a list of navigation items: Objects & Relationships, Collector Engine, PLC, Channels, Devices, Model, User Maintenance, System, System Configuration, Audit, Application Logs, and Subscriptions/Notifications. The main content area is titled 'Admin' and features a list of settings: State Display Settings, Object Classifications, Note Categories, Date & Time, Nomenclature, EDC Agent Settings, SMTP Settings, and Other. A 'Create New User' button is visible in the top right of the main content area. A large, semi-transparent text box is overlaid on the right side of the interface, containing the instruction: 'Click on a button or tab to go to that section of the manual. Or Click on this area to go to the Table of Contents'.





## What's New



To advance to the next section.

Click on the \***BEET** logo in the upper left side to go to **Table of Contents**.

Click the Page Number or Upper right-hand blue icon   to Go to the [Sectional Quick Jump](#) List.

\*These links are active when saved and opened in PDF or PDF in a web page only. Word documents require Control + Link to use.

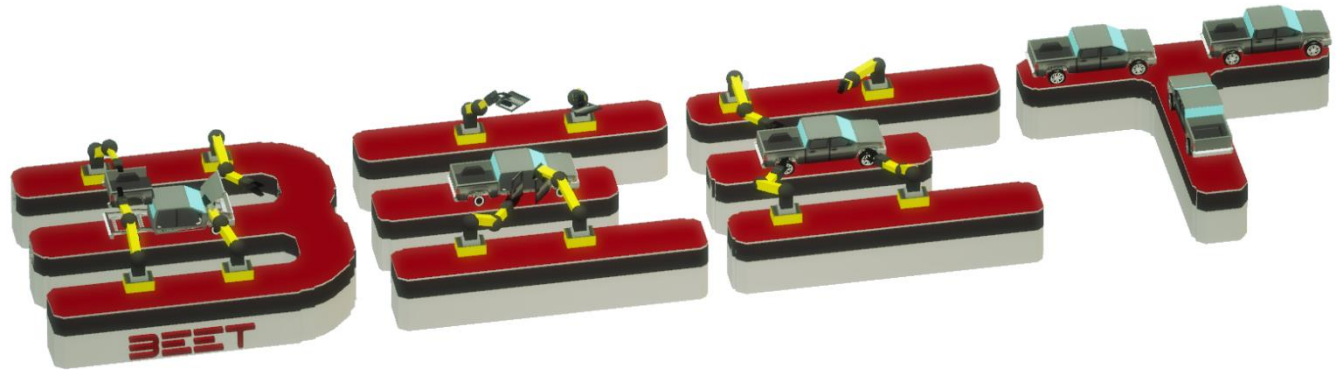
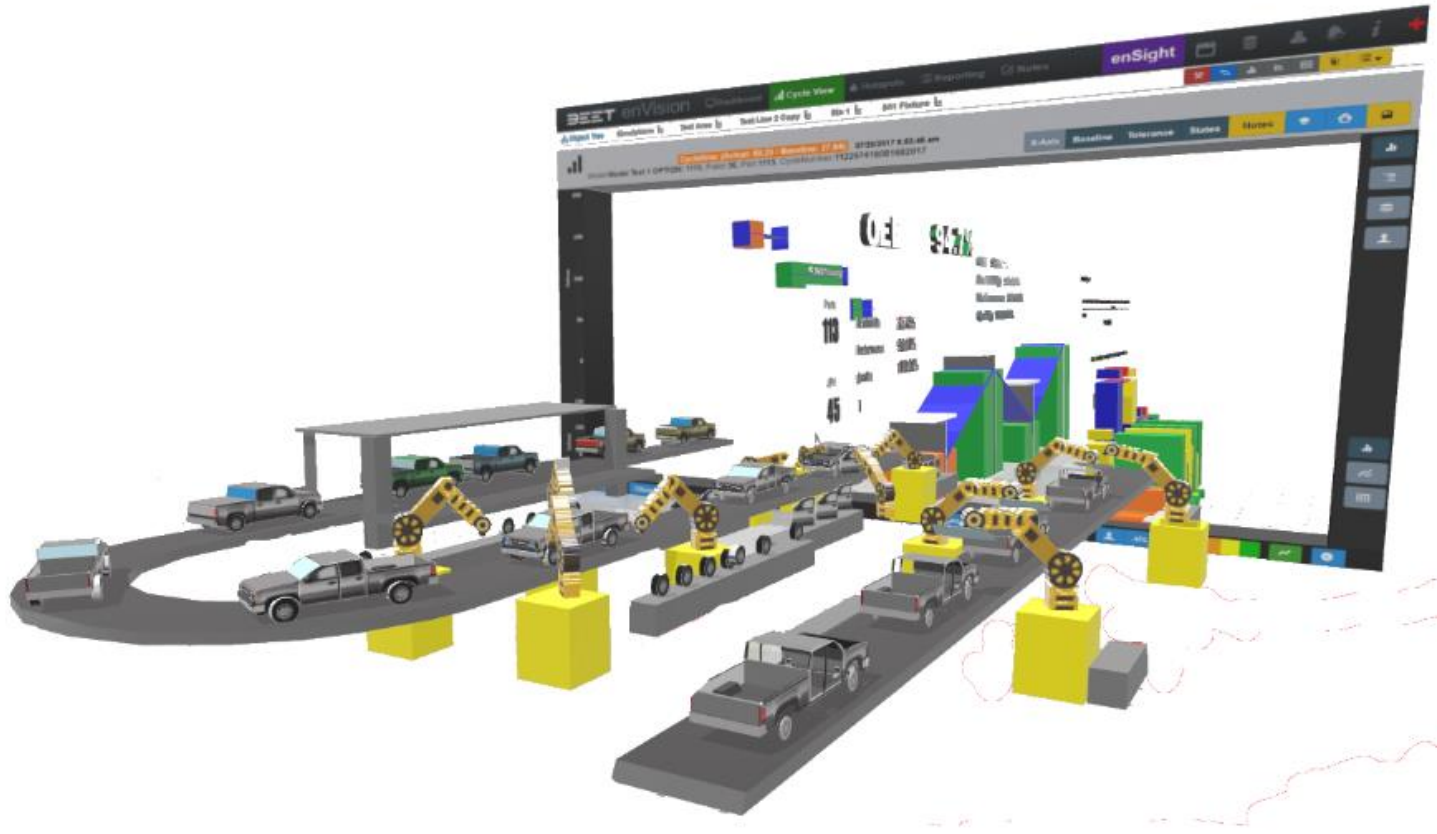
New link to the [Admin Menu Overview video](#).

## More Information

This document, when viewed as a PDF or web page (opened in a tab), has active hyperlinking which allows the user to navigate by clicking on a button in the image. Example, clicking on the PLC tab will take you to the PLC section of the document.

[Click on the image to view a section.](#)

The screenshot displays the BEET enVision Admin interface. At the top, the logo 'BEET enVision' is on the left, and 'enSight' with various utility icons and the user name 'admin' is on the right. Below the header is a navigation bar with buttons for 'Baseline Values', 'Deploy', 'Import', 'Export', 'Config', and 'Tags'. A 'Create New User' button is also present. On the left is a dark sidebar menu with categories like 'Objects & Relationships', 'Collector Engine', 'PLC', 'Channels', 'Devices', 'Model', 'User Maintenance', 'System', 'System Configuration', 'Audit', 'Application Logs', and 'Subscriptions/Notifications'. The main content area is titled 'Admin' and lists several settings sections: 'State Display Settings', 'Object Classifications', 'Note Categories', 'Date & Time', 'Nomenclature', 'EDC Agent Settings', 'SMTP Settings', and 'Other'. A large, semi-transparent text box is overlaid on the right side of the main content area, containing the instruction: 'Click on a button or tab to go to that section of the manual. Or Click on this area to go to the Table of Contents'.



# Introduction

## Overview

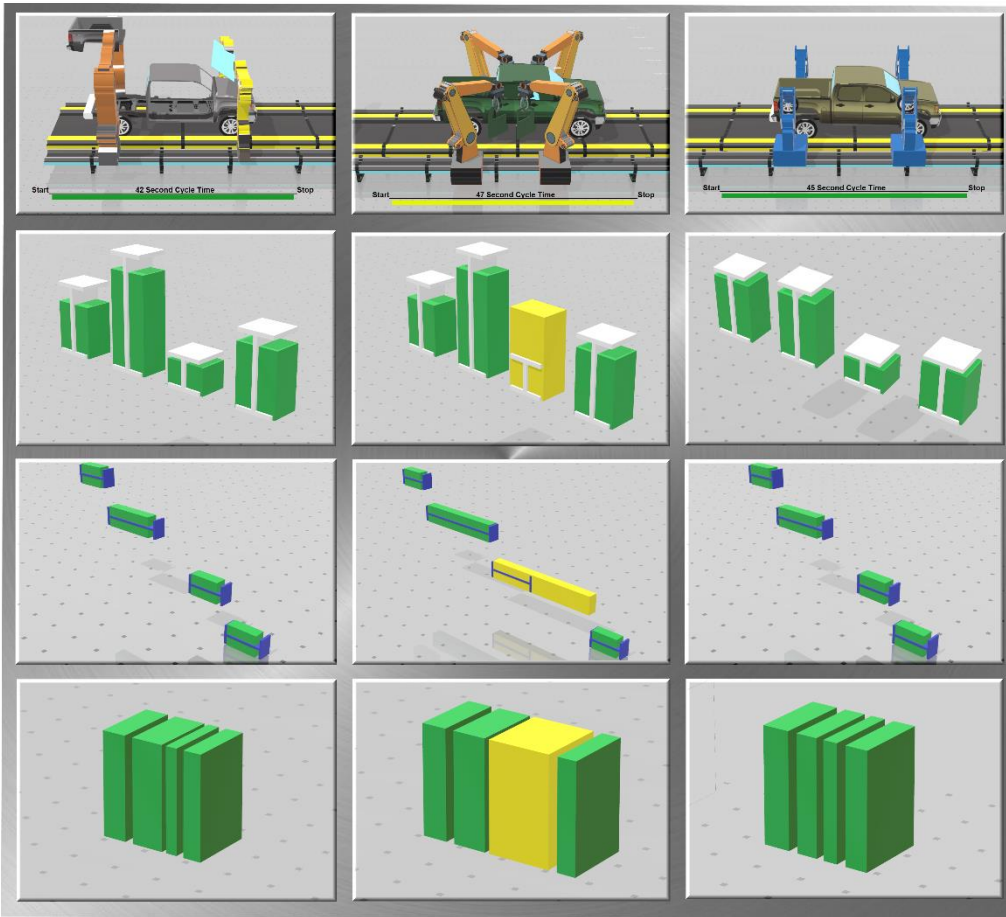
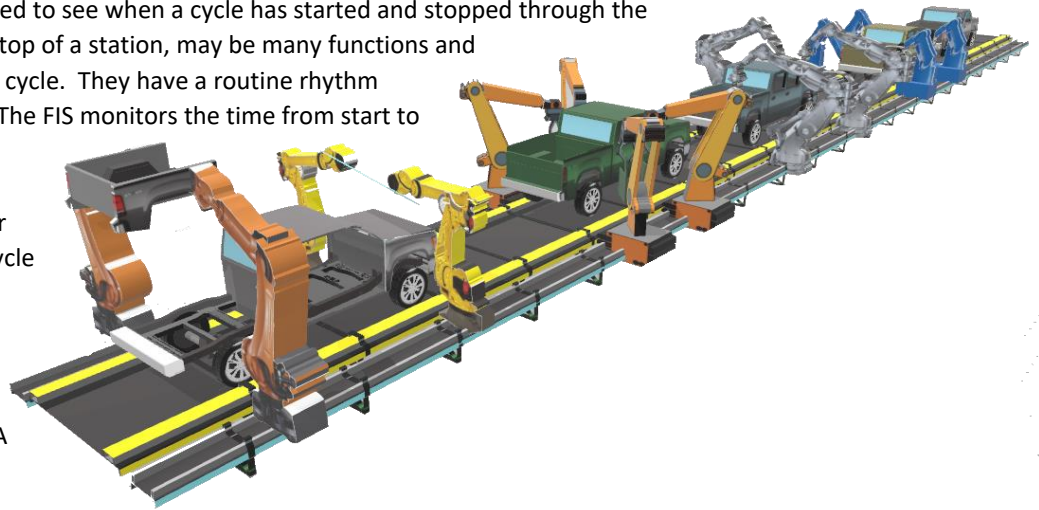
This document is intended to provide basic guidelines when it comes to entering descriptions and expressions for an enVision project. This applies whether the project is initially entered in an Excel spreadsheet that is later used to copy and paste the descriptors/expressions to enVision or typing the information into the enVision application directly.

Under a *typical* FIS, you would be limited to see when a cycle has started and stopped through the cycle time. In between the start and stop of a station, may be many functions and process that are happening during the cycle. They have a routine rhythm that can be tracked, but they are not. The FIS monitors the time from start to stop. It does not monitor the actual processes going on in between.

With **enVision**, it is possible to monitor the actual processes within a cycle. Cycle time is measured along with all the processes and operations in the cycle.

Each process within a station can be closely monitored to see if it is taking longer or shorter to complete. A single process within a cycle has a baseline that it set initially. If the operation takes longer than expected, it is recorded and shown

visually to show you where, when, and how long it took to complete. With the ability to set tolerances on a specific operation, the operation would be labeled Good, Watch, Warning or even Faulted.



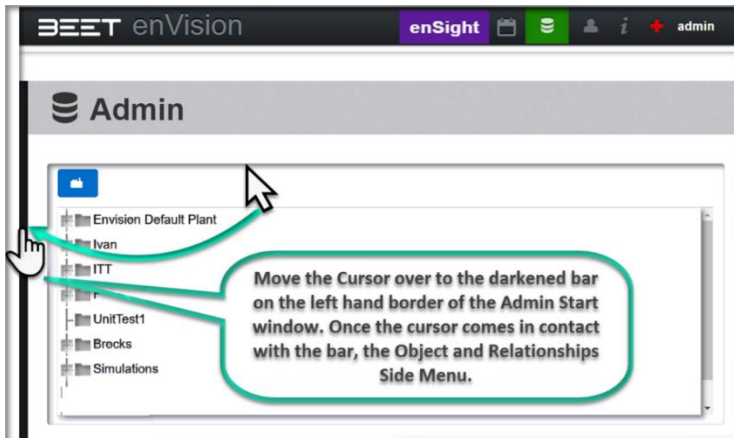
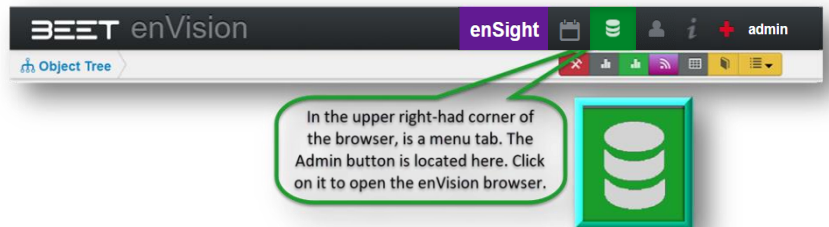
The Operation within the cycle, can be displayed in many forms. A column graph which shows the actual cycle time. When a cycle takes longer than the tolerance allows, it will turn from Green (Good) to Yellow (Watch), Orange (Warning) or possibly Red (Fault). A Bar chart would show the cycle length as well as the time between cycles (Dark area between). It can also be shown in a Sequence view. This would display the operations of the cycle in sequence of when they happened.

# Admin Module

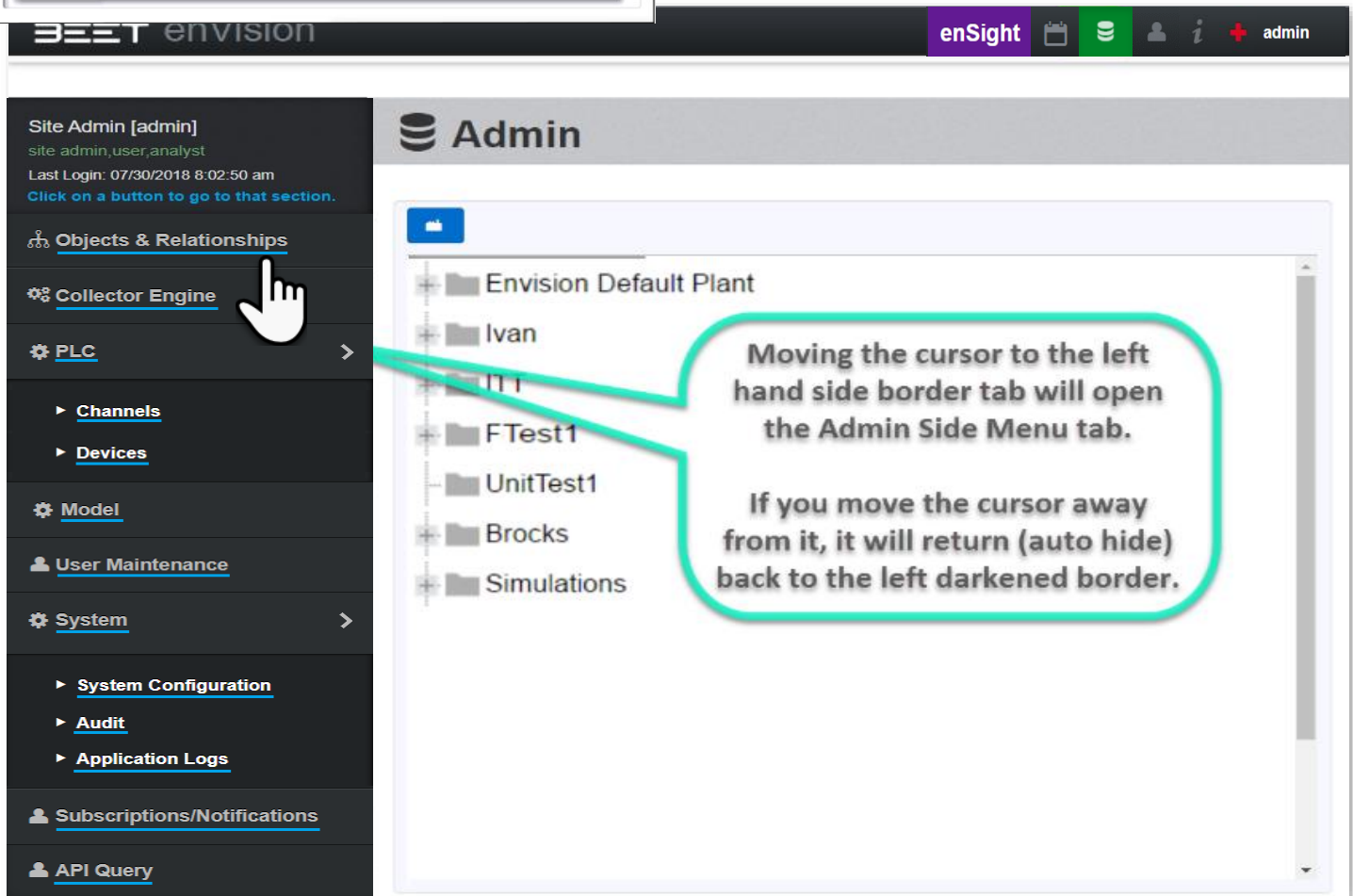
Click on the **Admin button** on the top right-hand side of the menu bar.

Slide the cursor over to Left-hand edge of the window until side menu appears.

This action will open a side display menu.



See [Admin Menu Overview video](#) for a brief description of the uses and functions of the Admin Browsers Administrators Side Menu.

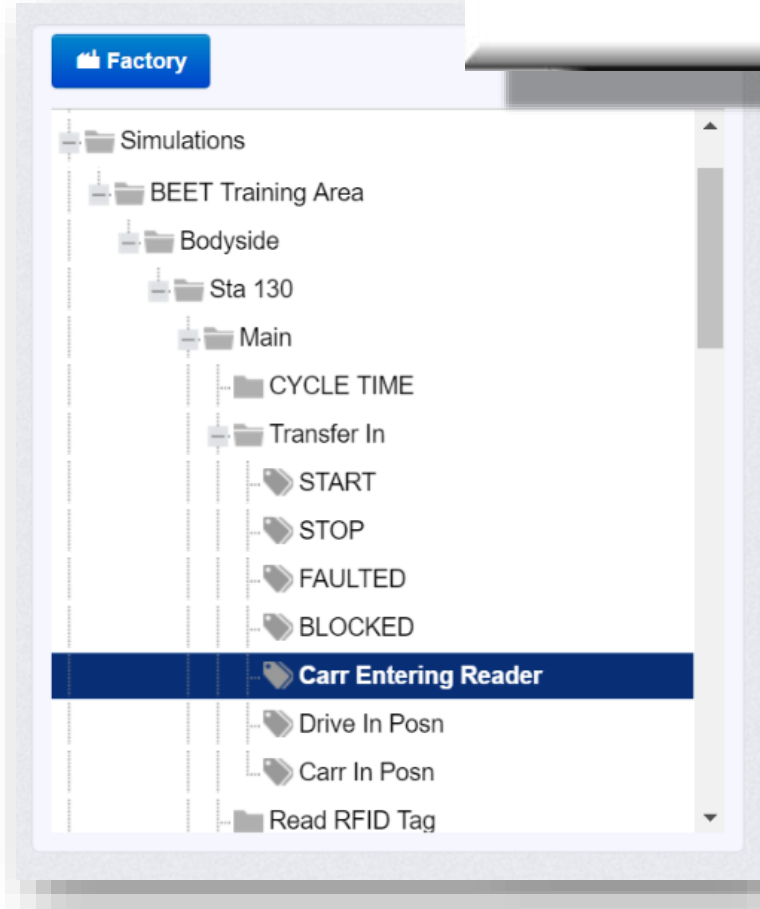
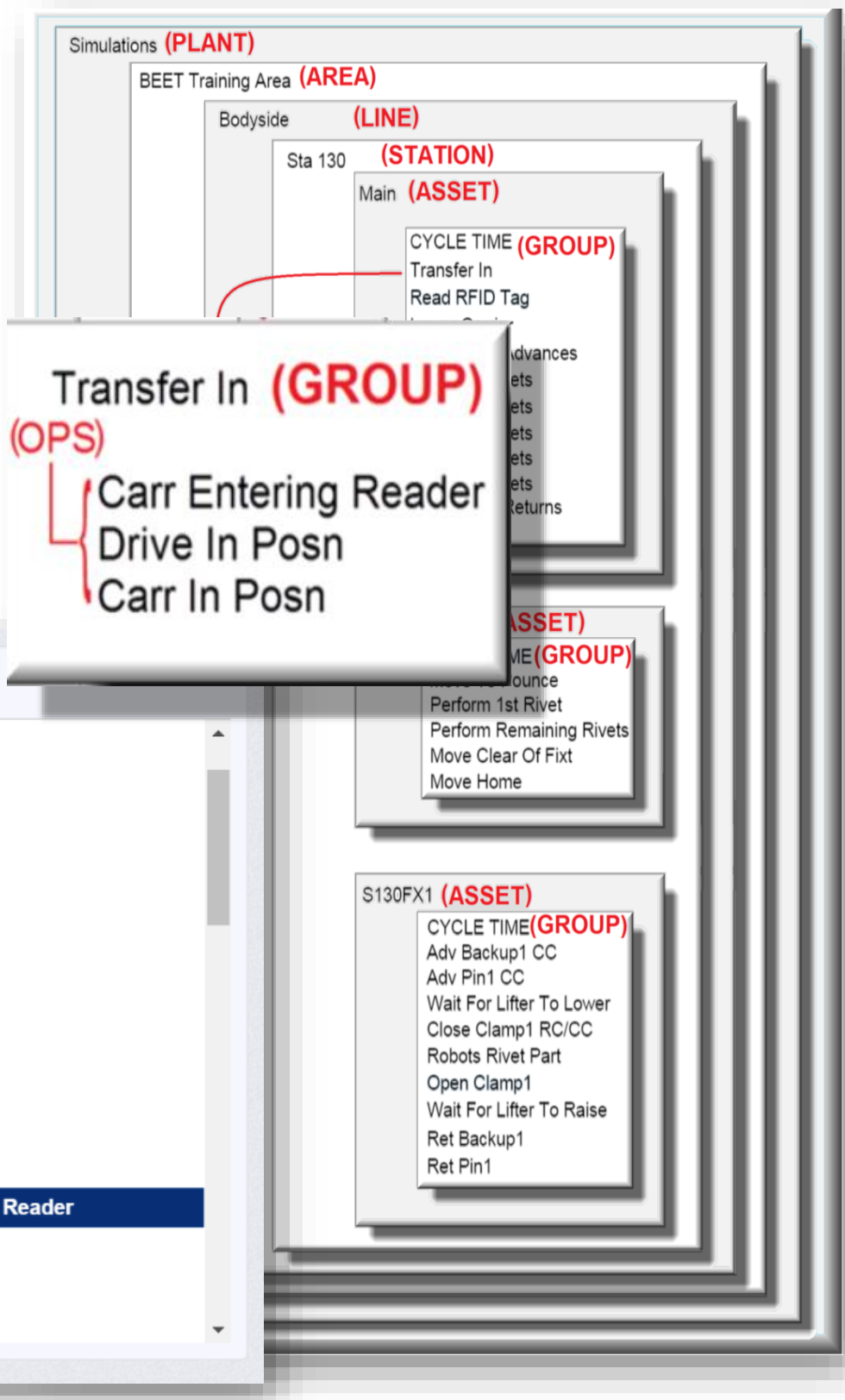


# Factory Tree

## Overview

This document is intended to provide basic guidelines when it comes to entering descriptions and expressions for an EnVision project. This applies whether the project is initially entered in an Excel spreadsheet that is later used to copy and paste the descriptors/expressions to EnVision or typing the information into the EnVision application directly.

- [Factory Window](#)
- [Objects and Relationships](#)
- [Building a Factory Tree](#)
- [Factory Window](#)
- [Object Overview](#)



## Factory window

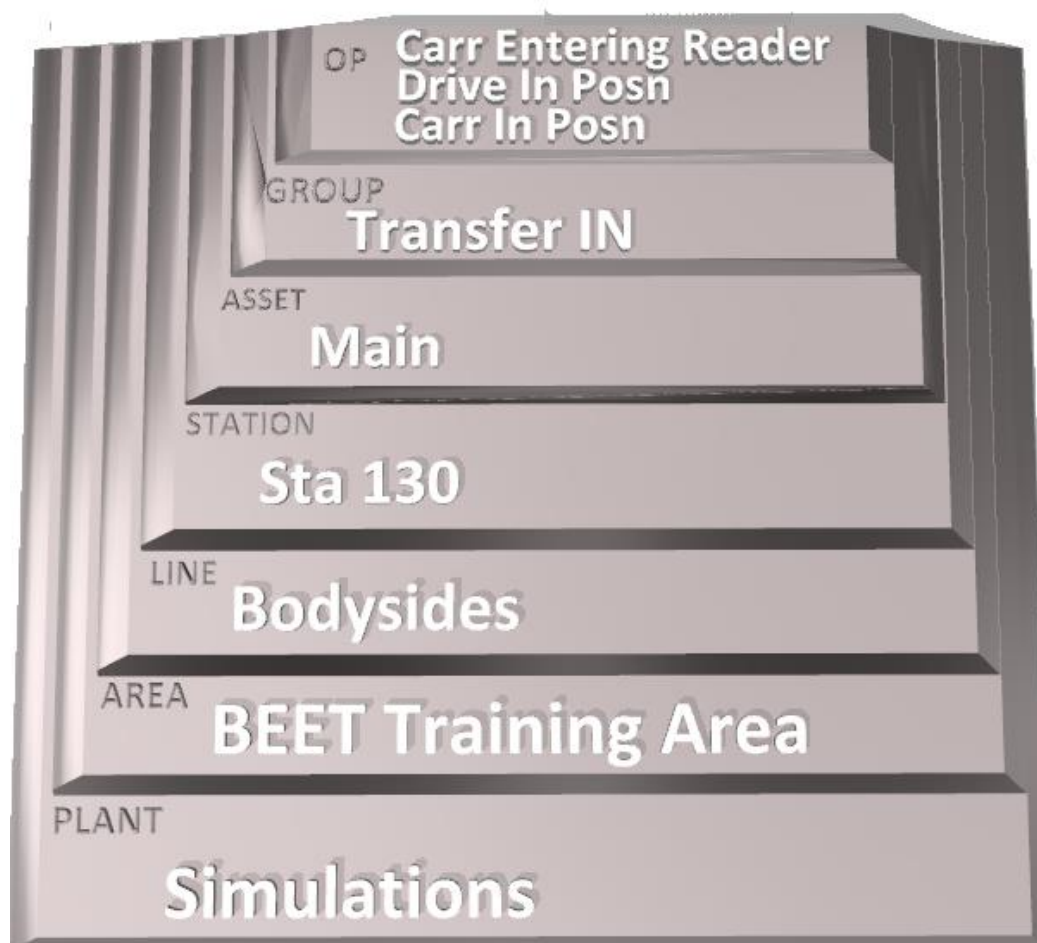
Basic structure of an enVision application and their suggested descriptor lengths.

Object tree -

- 1) Highest level referred to as the "**Plant**" - try to keep between 5 and 15 characters - no more than 20.
- 2) Next level (or 'Child') under Plant is the "**Area**" – it can be more than 1 - same punctuation rules as Plant
- 3) Next level under Area is the "**Line**" - it can be more than 1 - no more than 20 characters.
- 4) Next level under Line is the "**Station**" - it can be more than 1 - try to keep around 10 characters
- 5) Next level under Station is the "**Asset**" - it can be more than 1 - same punctuation rules as plant
- 6) Next level under Asset is the "**Group**" - it can be more than 1 - no more than 20 characters
- 7) Next level, (the last one) under Group is the "**OP**" - at least 1 but no more than 16 for any one Group - no more than 29 characters

Use a mixture of upper and lower case letters. This will help in interpreting abbreviations. Lower case letters also seem to use less screen area than upper case letters. Try to use logical abbreviations as much as possible.

Obviously there will sometimes be exceptions to these guidelines. Also, some customers use different terminologies for their processes such as using "OP ##" instead of "STA ##", etc. These need to be treated on a case by case scenario.



# Object Naming



## Factory (PLANT)

The Factory/Plant Object can be named after the Customer Name (Or Accepted Abbreviation) or Plant Name (Or Accepted Abbreviation). It is the highest-level object.

When naming, keep it between 5 and 15 characters - no more than 20.

## Area

The Area Object is a Customer Designated Area, such as a Bodyside, Door, Transmission, or other large part of a whole product. The Area Object is the "Child" under the Plant object.

When naming the Area Object, there can be more than one Area Object per Plant, follow the same character limits as the Plant. Example Bodyside and Closures.

## Line

The Line Object is a Customer Designated Line Segment and Style of Station range, such as a 6X S010 or 6X S010-S050 (if there are multiple stations in one or more of the Line Objects). The Line Object is the "Child" under Area object.

When naming the Line Object, it can be more than 1 - no more than 20 characters, stating a segment and a range of station (element) in the Line.

## Station

The Station Objects are Customer Designated Station number, usually in reference to the 2<sup>nd</sup> part of the Line Objects name (aka **S020** which is part of the 6X S010-S050 Line Object or S010, **S020**, S030, S040, and S050). The Station Object is the "Child" under Line object.

When naming the Station Object, there can be more than 1, to try to keep it around 10 characters.

## Asset

The Asset Objects are Customer Designated **Asset Name** and a **description of the type**. It consists of Main Stations, Robot Stations, and Fixture Stations. Main Stations, are Customer Designated Station Number. Robots are Numbers, Robot Number, and description. A Fixture is a station number. The Asset Object is the "Child" under Station object.

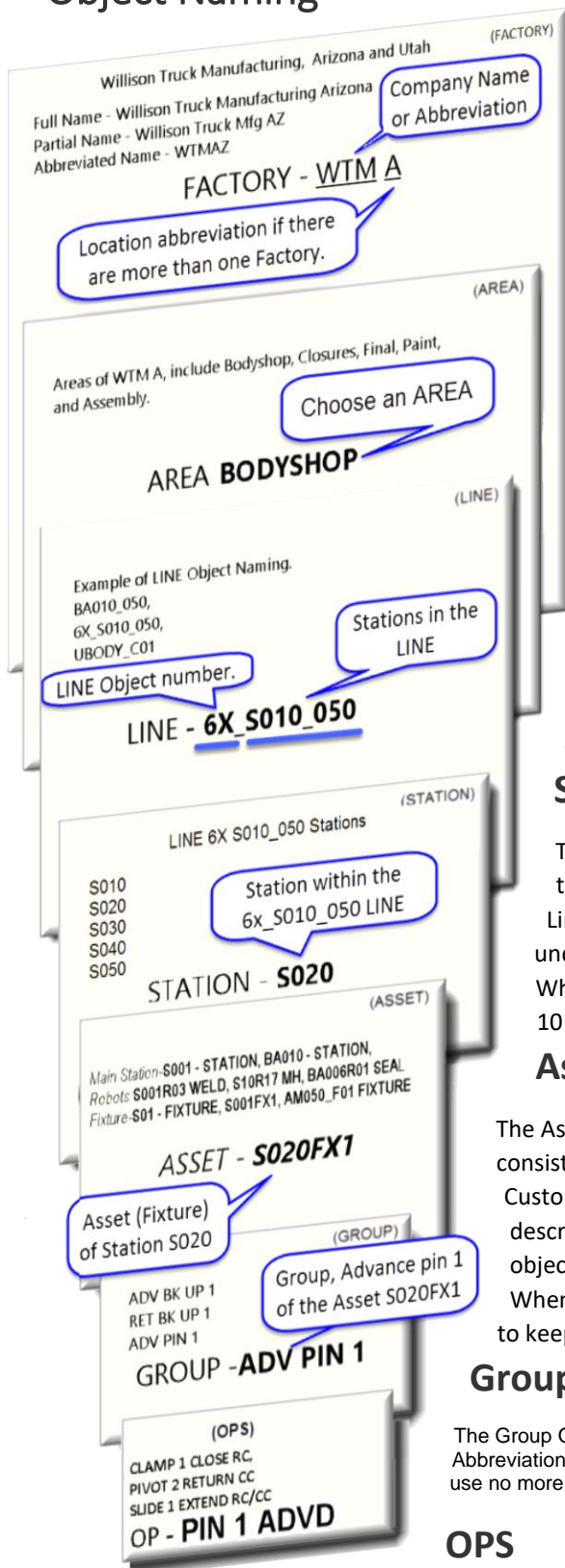
When naming the Asset Objects, there can be more than one Asset Object per Plant, try to keep it under 10 characters.

## Group

The Group Objects are Tooling Group – [Unit Type] [Unit Type Number#] [Unit Type Motion] Model Abbreviation. The Group Object is the "Child" under Asset object. There can be several Group objects, so use no more than 20 characters.

## OPS

The OPS or Operations objects are the "Child" objects of the Group Object. There may be at least 1 but no more than 16 for any one Group – use no more than 29 characters when naming the OPS. The naming of a Tooling Group OP typically consists of a **Unit Type Abbreviation**, a **Unit Number**, and a **Unit Type Position Abbreviation**.





# Object Overview

## Factory

**WTM A** / BODYSHOP/ 6X\_S010\_050 / S020/ S020FX1 / ADV PIN 1 / PIN 1 ADVD

The FACTORY for this example is the fictional **Willison Truck Manufacturing (WT Mfg.)**. They make several versions of trucks. In this example, there are two factories locations, Arizona and Utah.

Selecting a Factory name could use the initials of the company name and the location, if there are multiple sites.

- **WTM A** (Willison Truck Manufacturing) + A (Arizona). WTM A.
- **WTM U** for the Utah Factory.

## Area

WTM A / **BODYSHOP**/ 6X\_S010\_050 / S020/ S020FX1 / ADV PIN 1 / PIN 1 ADVD

In this factory, the manufacturer will have several areas that could be monitored. The Areas would be separated by the types or parts of a product they produce, such as Bodysides, Framing, Painting, Bodyshop, etc. For this, we would specify a particular AREA.

- **Bodyshop**, Bodyside, Framing

## Line

WTM A / BODYSHOP/ **6X\_S010\_050** / S020 / S020FX1 / ADV PIN 1 / PIN 1 ADVD

In the Bodyshop AREA, they may have multiple LINES. Each LINE has an abbreviated designation for them and the STATION or STATIONS in the LINE Name. If a line has two or more STATIONS within the LINE, you would want to use the LINE Number (**6X**) + the STATIONS (S010\_050) included in that LINE. Note the examples below.

- BA010\_050
- **6X\_S010\_050** (this line has multiple stations)
- UBODY\_C01 (this LINE has only one station)

## Station

WTM A / BODYSHOP/ 6X\_S010\_050 / **S020** / S020FX1 / ADV PIN 1 / PIN 1 ADVD

The STATIONS are the Child objects located in each **LINE** Object. Within the **6X S010-S050**, there would typically be 5 (S010, S020, S030, etc.) stations in each LINE Object:

- S010, S020, S030, S040, S050

## Asset

WTM A / BODYSHOP/ 6X\_S010\_050 / S020 / **S020FX1** / ADV PIN 1 / PIN 1 ADVD

The ASSET Object naming is a combination of the **Station Name + Type of station it is (Main, Fixture, or Robot)**. You would name as follows:

- **S020FX1** (Station Name (**S020**)+ Fixture(**FX**) + Number(**1**))

## Group

WTM A / BODYSHOP/ 6X\_S010\_050 / S020 / S020FX1 / **ADV PIN 1** / PIN 1 ADVD

Group Objects naming is typically made up from 2 group types (**Tooling** and **Robot Motions**) and 3 descriptors (**Unit Type Motion, Unit Type**, and **Unit Type Number**).

**Tooling Group** (Example - **ADV PIN 1** [Unit Type Motion] + [Unit Type] + [Unit Type Number#])

Unit Type Motions:			Unit Types:		Unit Type Numbers:
<b>ADVANCE</b>	<b>RETRACT</b>	<b>EXTEND</b>	<b>PIN</b>	<b>BACKUP</b>	<b>1, 2, 3, etc., dependent on how many of the type that are present in the group.</b>
<b>RETURN</b>	<b>ENGAGE</b>	<b>RAISE</b>	<b>CLAMP</b>	<b>VACUUM</b>	
<b>CLOSE</b>	<b>DISENGAGE</b>	<b>RETRACT</b>	<b>DUMP</b>	<b>PIVOT</b>	
<b>OPEN</b>	<b>LOWER</b>		<b>SLIDE</b>		

## Robot Motion Group:

<p><b>“GO TO”</b> Work Type Position:</p> <ul style="list-style-type: none"> <li>• <b>POUNCE</b></li> <li>• <b>PICK</b></li> <li>• <b>DROP</b></li> <li>• <b>1ST RIVET/FDS 2<sup>ND</sup>, LAST</b></li> <li>• <b>PEDESTAL</b></li> </ul>	<p><b>“MOVE”</b> Work Type Position</p> <ul style="list-style-type: none"> <li>• <b>CLEAR OF PICK</b></li> <li>• <b>CLEAR OF DROP</b></li> <li>• <b>CLEAR OF PED</b></li> <li>• <b>CLEAR OF FIXTURE</b></li> <li>• <b>HOME</b></li> </ul>
---	---

## OPS

WTM A / Bodyshop / 6X\_S010\_050 / S020/ S020FX1 / ADV PIN 1 / **PIN 1 ADVD**

## Tooling Group OP

The naming of a Tooling Group OP typically consists of a **Unit Type Abbreviation**, a **Unit Number**, and a **Unit Type Position Abbreviation**. The full spelling can be used, however, if the OP has a long descriptor, it may be advisable, to use an abbreviation.

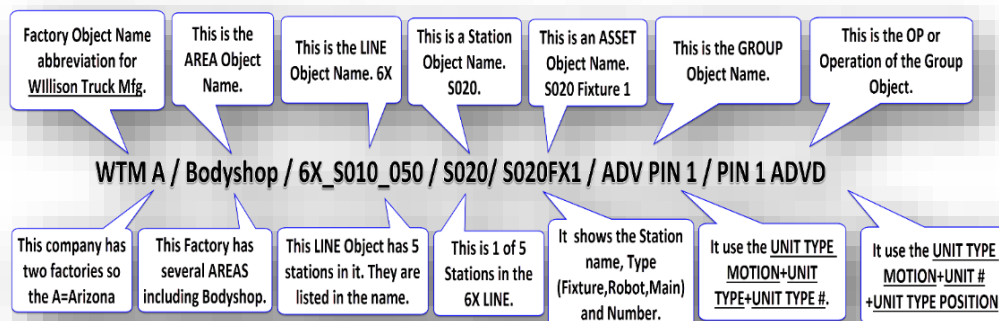
<p>Unit Type Abbreviation:</p> <ul style="list-style-type: none"> <li><b>PIN</b> (Pin)</li> <li><b>CLP</b> (Clamp)</li> <li><b>DMP</b> (Dump)</li> <li><b>SLD</b> (Slide)</li> <li><b>BKUP</b> (Back Up)</li> <li><b>VAC</b> (Vacuum)</li> <li><b>PVT</b> (Pivot)</li> </ul>	<p>Unit Type Number:</p> <p><b>1, 2, 3, etc.</b> Depending on the amount of the same Ops that a present.</p> <p><b>PIN 1 ADVANCED</b></p> <p><b>PIN 1 ADVD</b></p>	<p>Unit Type Position Abbreviation:</p> <ul style="list-style-type: none"> <li>Advanced = <b>ADVD</b></li> <li>Returned = <b>RETD</b></li> <li>Closed = <b>CLSD</b></li> <li>Opened = <b>OPND</b></li> <li>Extended = <b>EXTD</b></li> <li>Retracted = <b>RTCD</b></li> <li>Engaged = <b>ENGD</b></li> <li>Disengage = <b>DSNG</b></li> <li>Lowered = <b>LWRD</b></li> <li>Raised = <b>RSD</b></li> </ul>
--	--	---

## Robot Motion OP

The OP also uses Robot Motions stated as Work Type Position in the naming. These are **“AT”** and **“CLEAR OF”**.

<p><b>AT</b> consists of:</p> <ul style="list-style-type: none"> <li>• <b>POUNCE</b></li> <li>• <b>PICK</b></li> <li>• <b>DROP</b></li> <li>• <b>1ST RIVET/FDS 2<sup>ND</sup> LAST</b></li> <li>• <b>PEDESTAL</b></li> <li>• <b>HOME</b></li> </ul> <p><b>AT POUNCE</b></p>	<p><b>CLEAR OF</b> consists of:</p> <ul style="list-style-type: none"> <li>• <b>PICK</b></li> <li>• <b>DROP</b></li> <li>• <b>PED</b></li> <li>• <b>FIXTURE</b></li> </ul> <p><b>CLEAR OF FXT</b></p>
---	---

WTM A / Bodyshop / 6X\_S010\_050 / S020/ S020FX1 / ADV PIN 1 / PIN 1 ADVD



# Objects and Relationships

[Factory Window](#) | [Building a Factory Tree](#) | [Adding a Group or OPs](#) | [Start/Stop Signals](#) | [OPS Signals](#) | [Importing](#) | [Exporting](#) | [Configuration](#)

## Overview

From the **Object and Relationships** menu, you can view or create a Factory tree, as well as define the Op Tags. It is where you can create the structure that represents the plant of the manufacturing process. From there you can lay out how you want to collect the data as well as where you can setup all the information for collecting data.

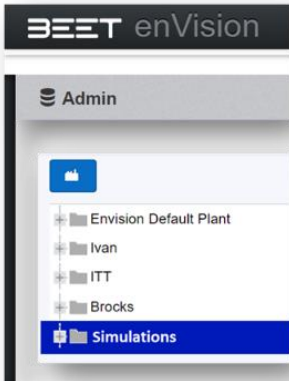
Move the **cursor** to the **left-hand** side of the screen to open the **Admin Menu Tab**. From there, **select** Objects and Relationships to open the Factory window.

Once the Factory window is open, **select** the Plant Object and another window will reveal to the right. From that window you have access to the Edit Select Object and Add New Child Tabs. The Edit Select Tab is for Adding, Delete, or Editing the object selected. The Add New Child tab, adds a new object below the current object selected. If you selected a Plant level object, clicking the Add New Child Object will add an Area level object.

The PLC and System tabs can expand to show other sub tabs to choose from. The PLC tab has **Channels** and **Devices**. The System tab contains the sub tabs for System Configuration, Audit, and Application Logs.

# Building a Factory Tree ▼

[Factory Window](#) | [Objects and Relationships](#) | [Adding a Group or OPs](#) | [Start/Stop Signals](#) | [OPS Signals](#) | [Importing](#) | [Exporting](#) | [Configuration](#)

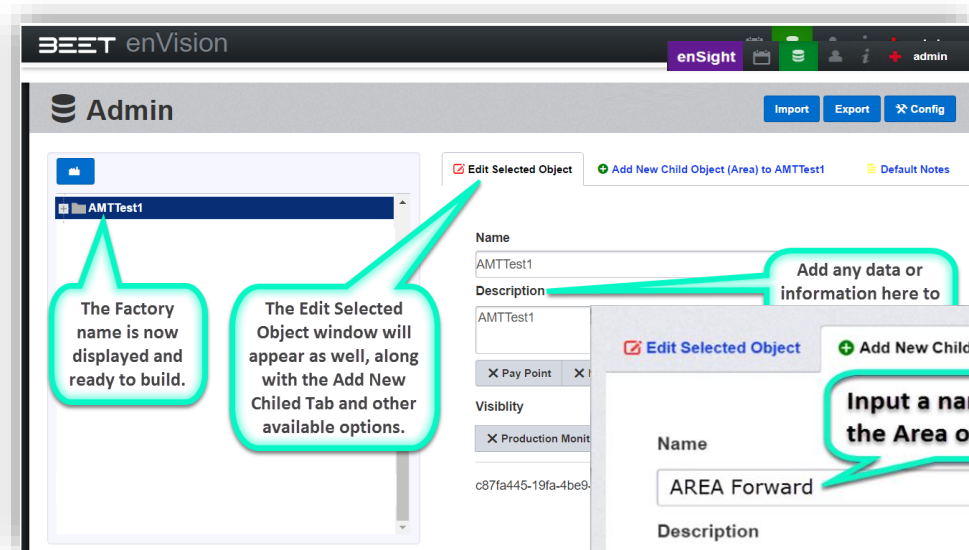
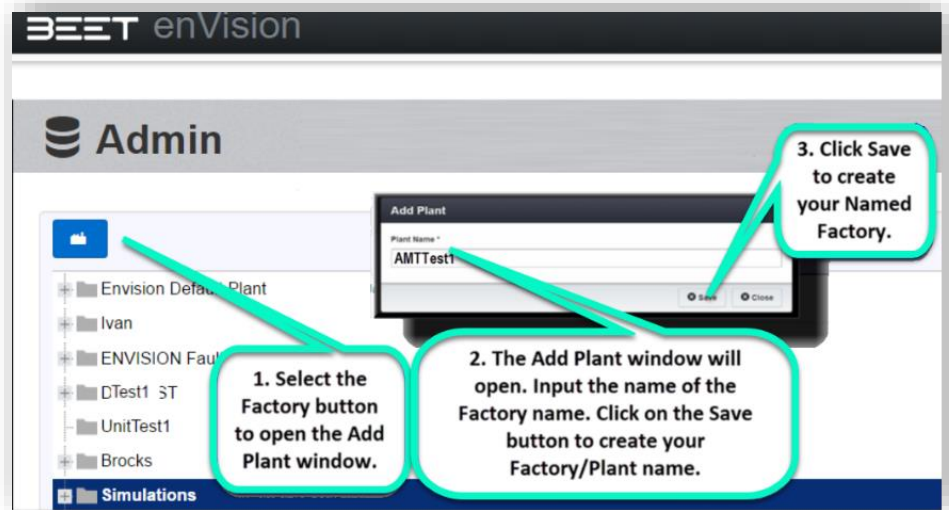


From the enVision Admin Browser, you can select a recent project if they are listed below. To start a new Factory Tree project, select the **Factory** button above the selection window. This will open an **Add Plant** window. From there, input the name that you would like to represent your company or plant. After the name is **Add Plant** click the **Save** button to create your new Factory.

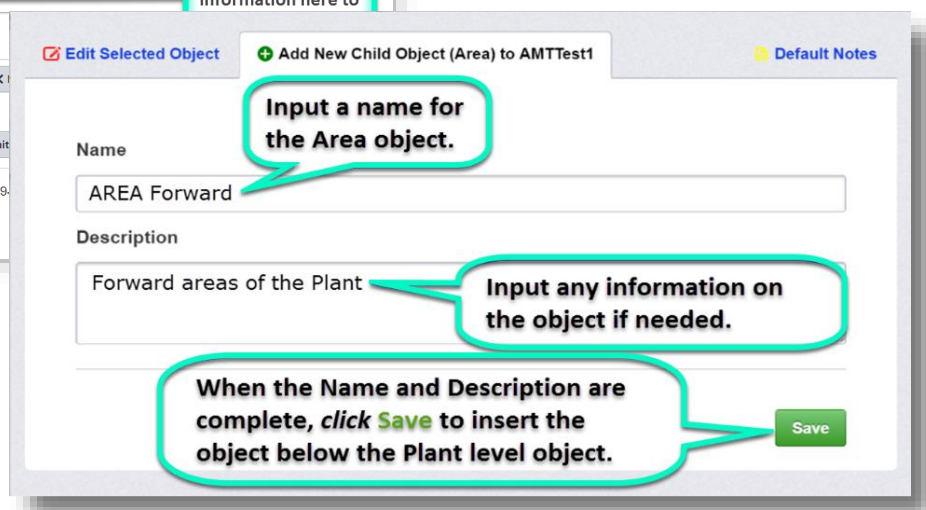
After the Plant Object is saved, this next screen will appear. The Plant object is in the Factory window. To the right of the Factory window is the **Editing** window. Here you can use the Edit Select Object tab to edit the Name, Description, and control where it is displayed (such as

Visibility).

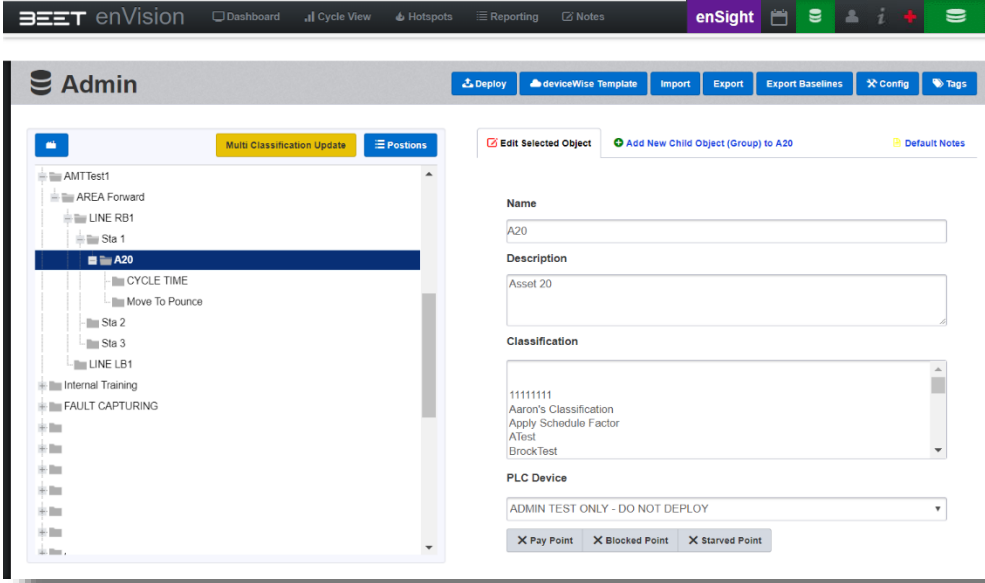
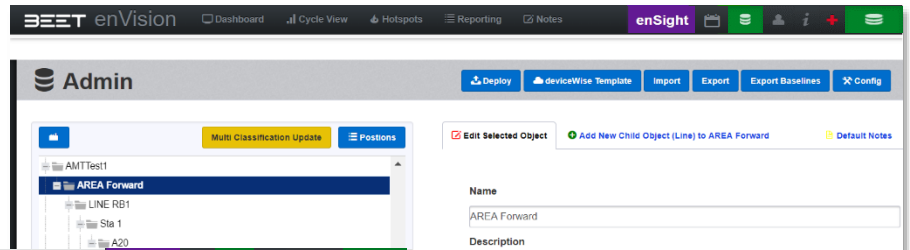
There is also a tab for **+ Add New Child Object**. This will add the levels below the Plant, down to the Group objects. In this tab, you name the **(Area)** Object and its description. You can use the Description area to add pertinent information about the object if needed.



When the **Name** and **Description** are filled out, you now can save by **clicking** the **Save** button. This will add the **(Area)** object under the **(Plant)** Object.

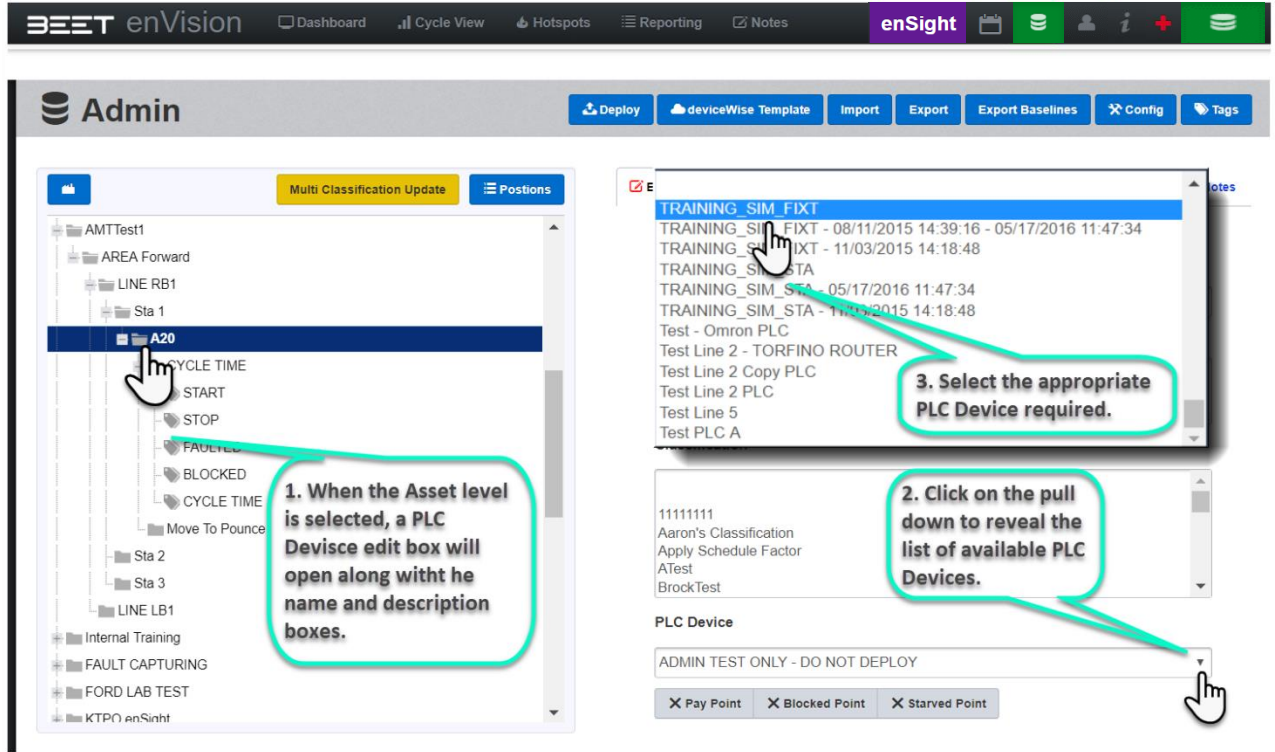


After clicking **Save**, this window will appear showing the new AREA Forward at the next level below the Plant Object (**AMTTest1**). This step can be repeated to all levels down to the Asset and Group Level objects.



In the Asset level, the editing window, with **Edit Selected Object** tab to the right now reveals the PLC

**Device** edit box. At the Asset level, you are required to assign a PLC Device to that Object. **Click** on the pull down to reveal the choices of PLC Devices available to use.



## Definition of Asset, Group and OP objects

### Assets

Assets are used to define a repeatable set of motions/activities (**Groups**) that have defined Start and End signals. Each occurrence is referred to as a "cycle" and is represented by a bar in the Cycle View history window. The number of Assets for a specific Station is completely dependent on the process and how the product transfers in and out of the station. For analysis and report features the following objects should be defined as unique Assets:

#### Overall Station Asset

This shows how the individual sub-Assets contained in the Station performed in relation to each other. This Asset can also have other Groups for individual motions that are independent of the sub-Assets such as the transfer of parts into and out of the station and tooling motions that are part of the overall Station sequence. In addition, this Asset might also have groups that represent an overview of other sub-assets that perform work during this Assets cycle. Examples include an operator group, robot groups, and a summary sub-asset.

#### Operator Asset

Each Station that requires a physical operator to enter a light screen protected area to perform some activity for every cycle is considered a unique Asset. If 2 operators share the same light screen area they will share the Asset. They can have separate groups for their Operator PB (Push Button) if they each have their own PB. If there is no light screen for each cycle an operator must acknowledge that they have performed some prescribed activity by pressing a PB for a cycle to continue, then a separate Asset will be required for that operator ONLY if there is a clearly defined signal that can be used to Start the Asset (example: Part in Pos - Rdy For Oper).

#### Robot Asset

Each robot will be a separate Asset. This is recommended not only because robots have considerable detail themselves and could cause an Overall Station Asset to become cluttered, but Robots may have processes that run outside of the Overall Station Asset's cycle. (e.g. Tip Dress, Cap Change, Magazine Fill)

#### Summary Asset

A summary asset can be used in situations where there is a high density Overall Station Asset or when there is a tool change that can have considerable sequences changes. This asset is generally used to combine large tooling fixtures into smaller summary groups in the overall asset, but maintains the level of detail desired in a separate sub-asset. Some examples could include changeable framing gates, fixtures mounted on tool-trays, or even high unit count fixtures mounted on pallet transfer systems.

## Asset States

Asset States are used to flag important periods in a cycle and can also be used for reporting purposes.

**Auto** – Used to indicate when machine is currently in automatic production mode. This state can be concurrently active with other machine states.

**Faulted** - Used to denote there is an active machine fault, and the cycle will not be used to calculate averages for reporting purposes. This state can be concurrently active with other machine states.

**Blocked** – Used to indicate when an Asset has reached a point of work complete, but cannot progress further due to a downstream blockage. This state can be concurrently active with other machine states.

**Starved** – Used to indicate when an Asset is ready to receive and start the next cycle, but an upstream condition is preventing advancement. This state can be concurrently active with other machine states.

**Non-Production** – Used to indicate Planned Downtime. While this state can be active concurrently with other states, it is recommended to the sole active state for reporting and tracking purposes.

## Additional notes

All part present sensors should be monitored. Quite often it is a part present that causes a cycle to go over. We need to show this situation as clearly as possible. The enVision PLC drivers have the ability to track the pattern of a part present going on and off during a cycle if it is desired to monitor this activity. This may come in very handy when trying to monitor operator part load efficiency.

All operator light screens need to be monitored. The enVision PLC software driver has the ability to capture the on and off status of light screen interruption as long as the appropriate signal from the light screen is wired to an input.

## Special Case Cycles

There are situations when a cycle can have an abnormal process; this can be in the form of an alternate path—when the Asset uses a limited number of different groups or Ops while retaining the original CycleTime—or an outside influence which causes an expected change in performance. For the cases involving an alternate path, it is suggested to distinguish these cycles by using a different Model. If the alternate path is minor, or the use of a distinct MODEL is not preferred, it is also possible to have a GROUP with two Ops to denote the path options; to do this correctly, when one path is taken, the OP designating the unused path must be recorded with a zero-duration timestamp otherwise a purple, missing data, record will be generated. When there is the possibility for an outside influence, the usage of STATE identifiers is suggested to be used flagging the cycles to be ignored, or processed separately, for average cycle time calculations.

### Possible Special Cases:

- Model Changeover with Tool Change—STATE FLAG
- Runout—STATE FLAG
- Processes with possible alternate paths
  - Pull-off & Swap—MODEL
  - MH Robot with an alternate Pick/Drop Locations—MODEL/2 OP w/ zero duration
  - MH Robot with an alternate Work Pedestal—MODEL/2 OP w/ zero duration

## Groups

Assets are a collection of Groups that describe a sequence of events that comprise one complete cycle for the Asset. The first Group for every Asset is called the CYCLE TIME Group and is created automatically when the Asset is created. This Group is used to control the overall Start and Stop of the Asset. All other Groups that are added to describe the sequence for the Asset should have their Start and Stop signals occur between the Start and Stop of the CYCLE TIME Group. In most situations, the CYCLE TIME Group Start signal is the same signal used for Group Start signal that describes the first step of the sequence. A Group is comprised of a START signal, a STOP signal and 1 to 16 OP signals. There are also placeholders for a Fault signal and a Blocked signal for each Group which are used in special cases which we will not discuss here.

## START signal

The **START** signal is used to indicate that the step has been initiated. Solenoid valves, robot initiate, operator clear to enter, etc. are examples of common Start signals. The START signal is a retentive signal and goes true upon the first OFF → ON transition. It is not reset until the Group STOP signal goes true. The enVision PLC driver stores the current Timestamp during the scan that the transition is detected.

## STOP signal

The **STOP** signal is used to reset the Group and clear all retentive signals and memory bits in the PLC driver that are associated with that Group. The only Group that MUST have an expression entered for its STOP signal is the CYCLE TIME Group. This STOP signal has special meaning to the enVision application. It turns on the Asset Stop signal inside the PLC driver that indicates to enVision that an Asset has completed a cycle and that all the data captured by the application since the Start of the Asset can be processed and stored as historical data. All Groups other than the CYCLE TIME Group have the option of leaving the STOP signal blank. The Deploy function that generates the PLC driver logic is designed to automatically use the Asset Stop signal to fill in any Group STOP signal that is left blank. It is strongly recommended to leave Group STOP signals blank IF the Group is expected to occur only once per cycle. This ensures that all Groups get reset at the end of a cycle.

There are a couple of exceptions touched upon earlier where it may be desirable to capture multiple occurrences of a Group during one cycle of an Asset such as monitoring how many times an operator moves in and out of a light screen and the gaps between each occurrence. For this scenario, it is necessary to enter an expression for the STOP signal to reset the Group before the cycle ends.

## OP signals

The **OP** signals are the expected results (inputs) for the Group. They should be edited to describe the specific input they represent. There can be 1 OP signal or up to as many as 16 OP signals for any one Group. The START signal is used to represent the beginning of a sequence step. When all the OP signals for a Group have activated, this indicates that the sequence step is complete. Like the START signal, OP signals are retentive and go true upon the first OFF → ON transition for each OP independently AFTER the START signal has gone true. OP signals are ignored until the Group has been Started. They are not reset until the Group STOP signal goes true. A Group is considered as Running once the START signal goes true up until all the OP signals for the Group have gone true. The Group will not be able to START again until the Group STOP signal has gone true.



## Examples:

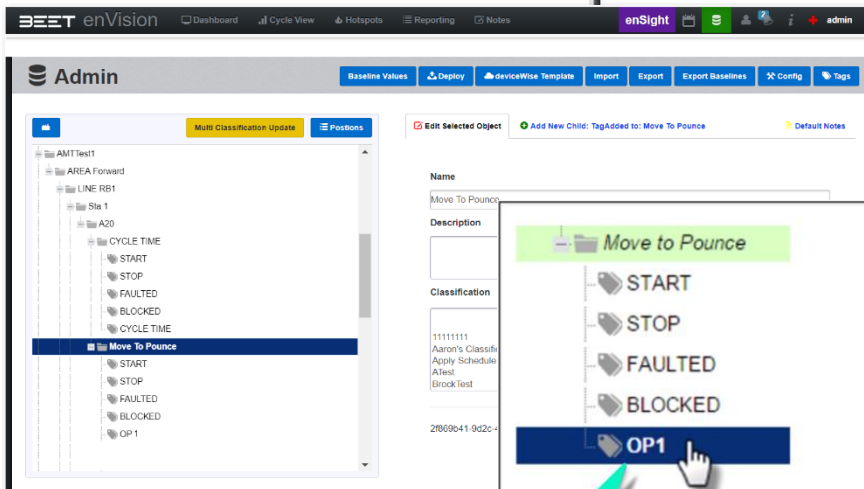
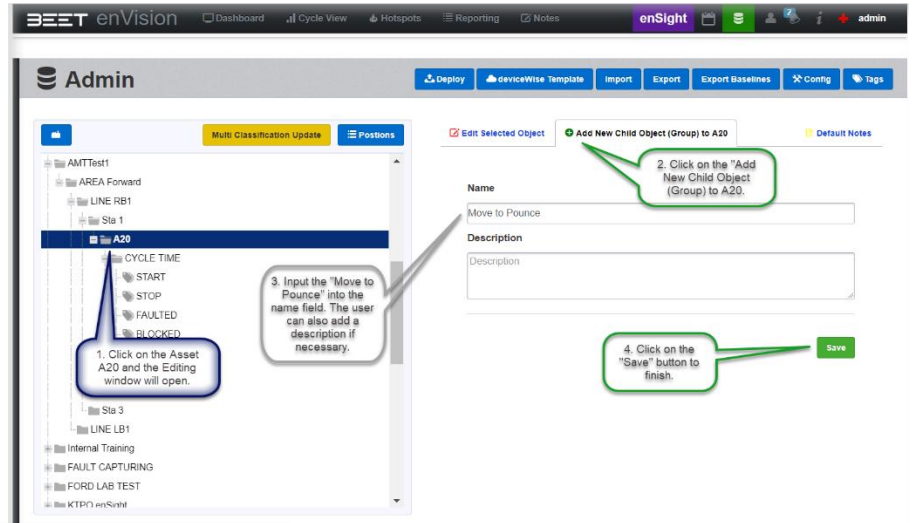
- Overall Station Asset
  - Pallet Transfer Stations
    - CycleTime
    - Transfer In
    - Tooling to Clear to Lower Position (One group per pneumatic valve/Summary sub-asset Group)
    - Read RFID
    - Lifter Lower
    - Part Presents
    - Tooling to Work Position (One group per pneumatic valve/Summary sub-asset Group)
    - Robots Work
    - Tooling to Clear to Raise Position (One group per pneumatic valve/Summary sub-asset Group)
    - Lifter Raise
    - Transfer Out
  - Tooling Fixture Stations
    - CycleTime
    - Tooling Reset to Clear to Load (One group per pneumatic valve/Summary sub-asset Group)
    - Operator/Robot Load Parts
    - Tooling to Work Position (One group per pneumatic valve/Summary sub-asset Group)
    - Robots Work
    - Tooling to Clear to Unload Position (One group per pneumatic valve/Summary sub-asset Group)
    - Operator/Robot Unload Parts
- Operator Asset
  - CycleTime
  - Break LightScreen
  - Load Parts (An OP for each Part Present)
  - Press PB
- Robot Asset
  - Carried Application
    - CycleTime
    - Go to Pounce
    - Individual work completes (e.g. 1<sup>st</sup>, 2<sup>nd</sup>, 3<sup>rd</sup> weld complete)
    - Move Clear of Fixture
    - Move Home
    - Tip Dress/Cap Change/Magazine Fill
  - Material Handling Application
    - CycleTime
    - Go to Pounce
    - Go to Pick
    - Part Presents
    - EOAT (One group per pneumatic valve)
    - Move clear of Pick
    - Individual work completes (e.g. 1<sup>st</sup>, 2<sup>nd</sup>, 3<sup>rd</sup> weld complete)
    - Go to Drop
    - EOAT (One group per pneumatic valve)
    - Move Clear of Drop
    - Move Home

## Adding a Group and OPs

Below the Asset are the Group levels objects. Under the groups are the OPS, also referred to as Operations. To add another Group object:

1. **Click** on the Asset and the Editing window will open.
2. **Click** on the Add New Child Object (Group) A20.
3. Input **"Move to Pounce"** in the name field.
4. **Click** the **Save** button when finished.

The A20 Asset should update to know have the Move to Pounce Group object. The OP **"OP1"** needs to be renamed to reflect the process properly.

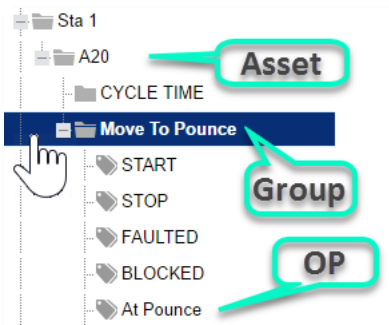


**Select** the OP1 OP, then select the **"Edit Selected Object"** tab located on the right side

of the page. **Delete** OP1 from the name field and replace it with **"At Pounce"**.

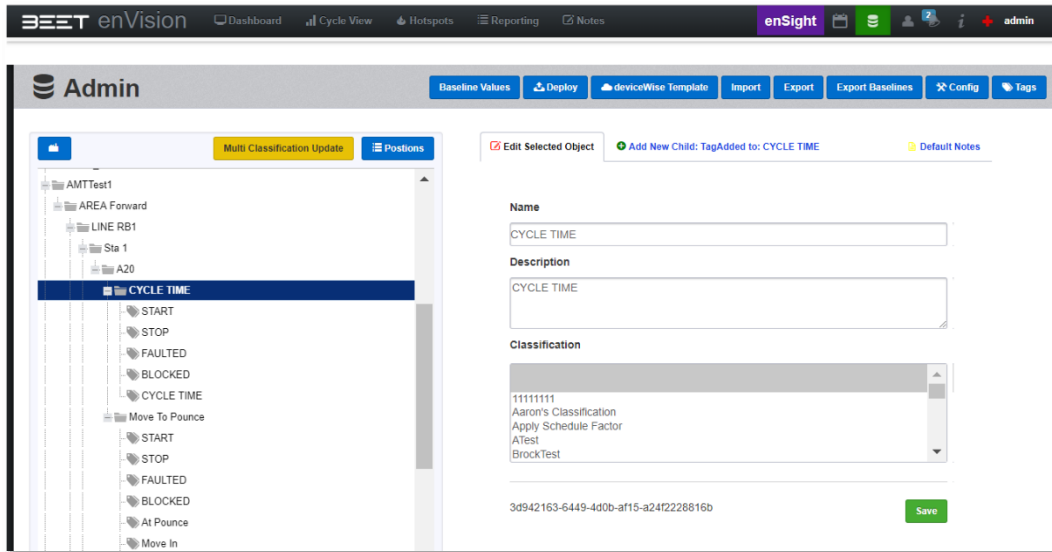
**Click Save**.

**REMEMBER** to Double check to be sure that you now have a properly labeled project from the Asset to the OP.



## START signal

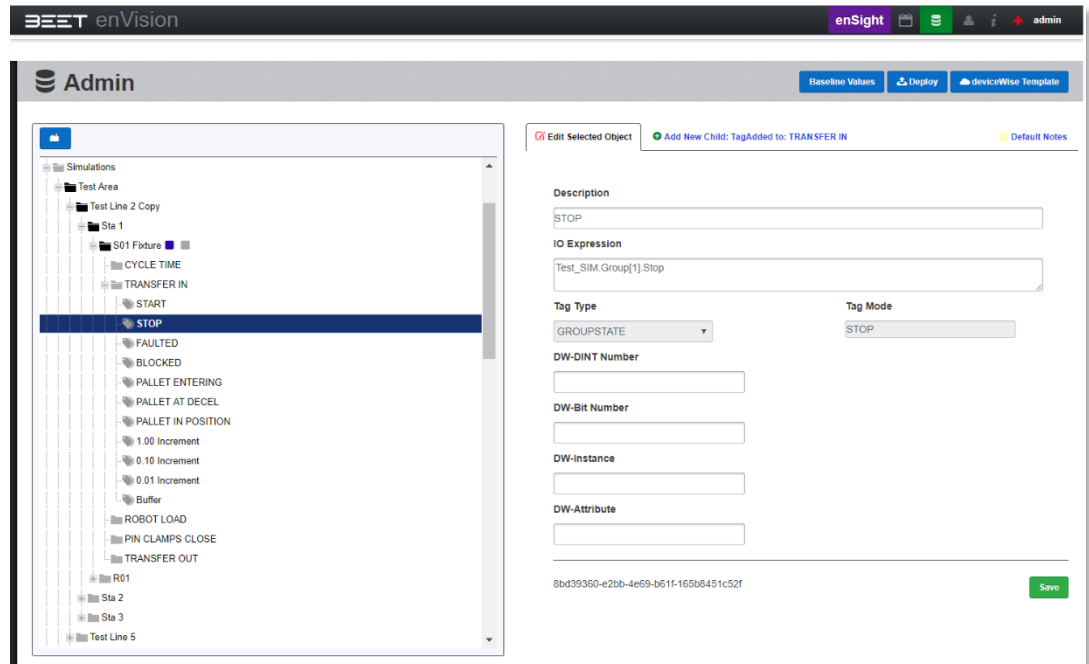
START is used to indicate that the step has been initiated. Solenoid valves, robot initiate, operator clear to enter, etc. are examples of common Start signals. The START signal is a retentive signal and goes true upon the first OFF → ON transition. It is not reset until the Group STOP signal goes true. The enVision PLC driver stores the current Timestamp during the scan that the transition is detected.



## STOP signal

STOP is used to reset the Group and clear all retentive signals and memory bits in the PLC driver that are associated with that Group. The only Group that **MUST** have an expression entered for its STOP signal is the CYCLE TIME Group. This STOP signal has special meaning to the enVision application. It turns on the Asset Stop signal inside the PLC driver that indicates to enVision that an Asset has completed a cycle and that all the data captured by the application since the Start of the Asset can be processed and stored as historical data. All Groups other than the CYCLE TIME Group have the option of leaving the STOP signal blank. The Deploy function that generates the PLC driver logic is designed to automatically use the Asset Stop signal to fill in any Group STOP signal that is left blank. It is strongly recommended to leave Group STOP signals blank IF the Group is expected to occur only once per cycle. This ensures that all Groups get reset at the end of a cycle.

There are a few exceptions discussed earlier where it may be desirable to capture multiple occurrences of a Group during one cycle of an Asset such as monitoring how many times an operator moves in and out of a light screen and the gaps between each occurrence. For this scenario, it is necessary to enter an expression for the STOP signal in order for the Group to reset before the cycle ends.



# OP signals

OP signals are the expected results (inputs) for the Group. They should be edited to describe the specific input they represent. There can be 1 OP signal or up to as many as 16 OP signals for any one Group. The START signal is used to represent the beginning of a sequence step. When all the OP signals for a Group have gone true this indicates that the sequence step is complete. Similar to the START signal, OP signals are retentive and go true upon the first OFF → ON transition for each OP independently AFTER the START signal has gone true. OP signals are ignored until the Group has been Started. They are not reset until the Group STOP signal goes true. A Group is considered as Running once the START signal goes true up until all the OP signals for the Group have gone true. The Group will not be able to START again until the Group STOP signal has gone true.

The screenshot shows the BEET enVision Admin interface. On the left is a tree view of the system hierarchy. The 'At Pounce' signal is selected under the 'Move To Pounce' group. The right pane shows the configuration for this signal:

- Description:** At Pounce
- IO Expression:** (Empty field)
- Tag Type:** OP
- Tag Mode:** (Empty field)
- DW-DINT Number:** (Empty field)
- DW-Bit Number:** (Empty field)
- DW-Instance:** (Empty field)
- DW-Attribute:** (Empty field)

At the bottom right, there is a unique ID: a7675b47-999d-46b7-96c3-6d604fe28912, and buttons for 'Delete' and 'Save'.

# Importing

[Factory Window](#) | [Building a Factory Tree](#) | [Adding a Group or OPs](#) | [Start/Stop Signals](#) | [OPS Signals](#) | [Importing](#) | [Exporting](#) | [Configuration](#)

## Import

The Import button is used to install an **.enx file** or [Import SeqOfOp .json](#) file.

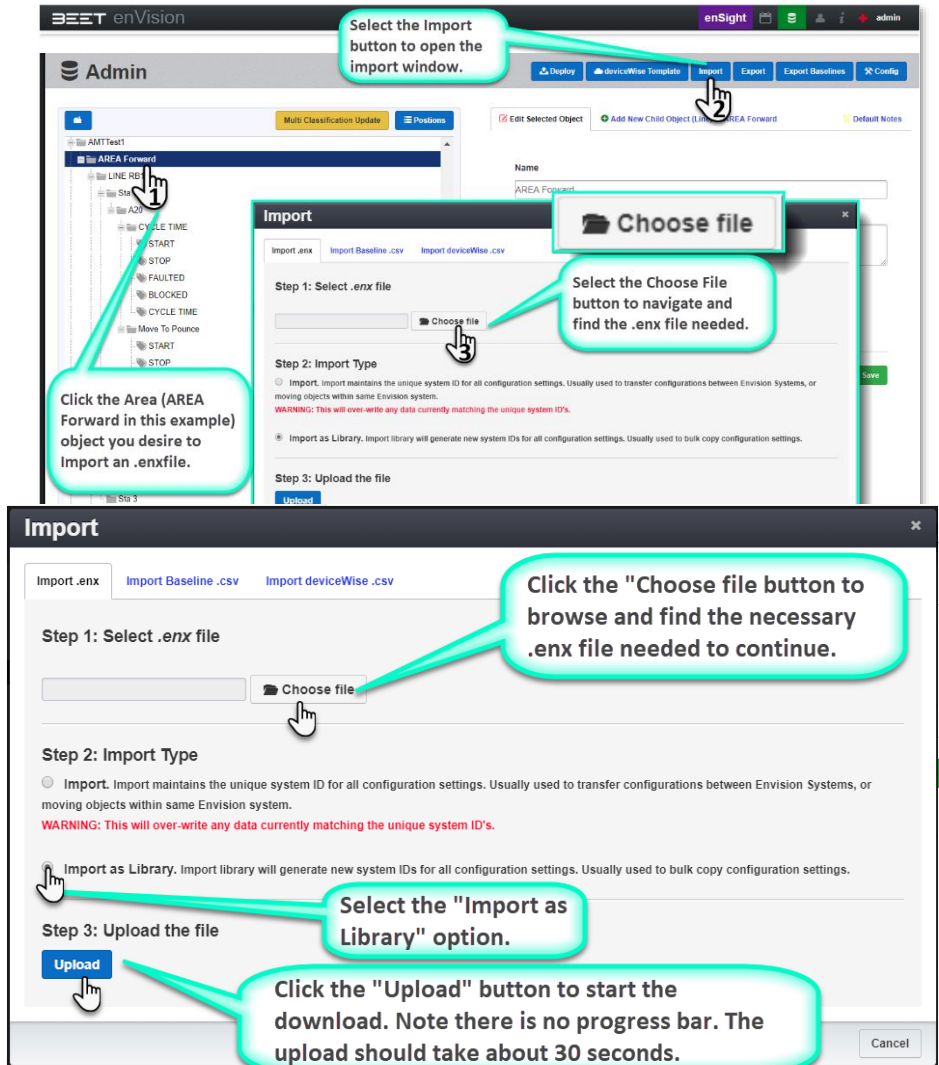
### Importing .enx file

Click on the object you would like to import to. Click the Import button and the Import window will open.

Select the **“Choose file”** button to open and search for the folder that contains the **.enx** file.

Select the **Import as Library** option. Click the **Upload** button to start the download.

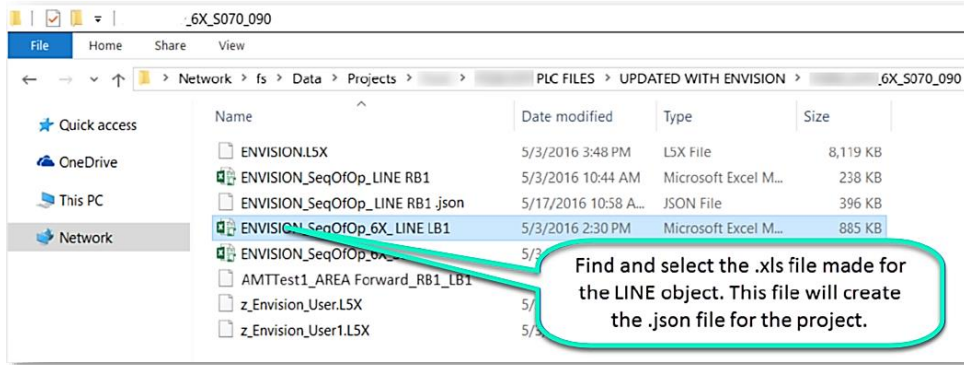
**NOTE:** There is no progress bar. The upload should take about 30 seconds.



# Importing SeqOfOp .json file

[Factory Window](#) | [Building a Factory Tree](#) | [Adding a Group or OPs](#) | [Start/Stop Signals](#) | [OPS Signals](#) | [Importing](#) | [Exporting](#) | [Configuration](#)

This process is done starting at the **LINE** level. At that level, is the only time that will allow you to import a .json file. Before you select the **Import** button, it may be necessary to create a .json file. Navigate to the SeqOfOp.xls file that contains the data needed to build the Factory Tree.

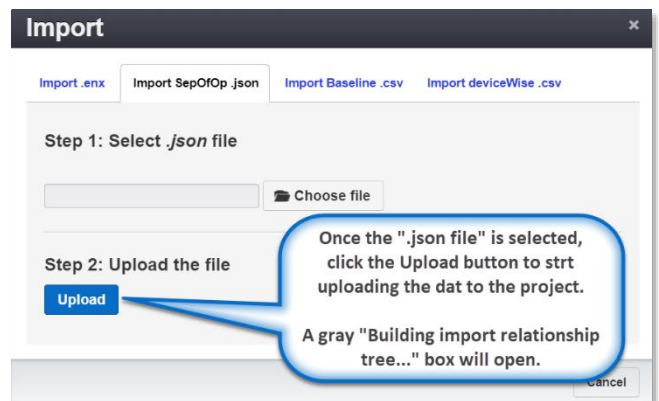
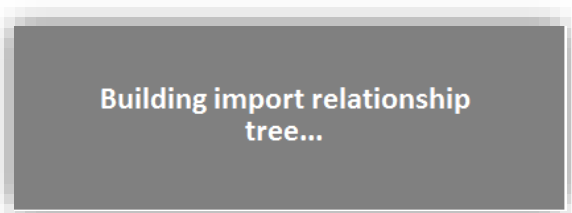
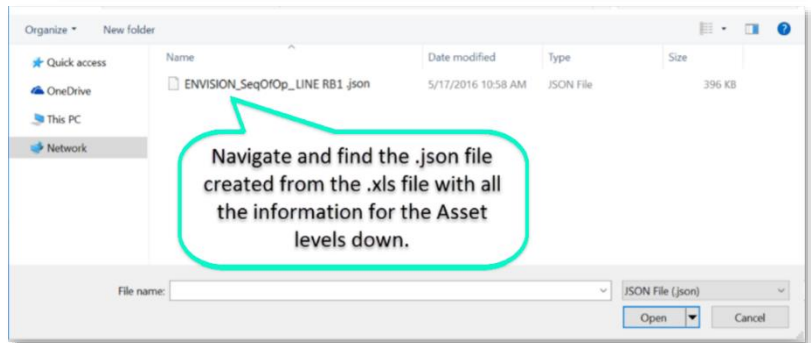
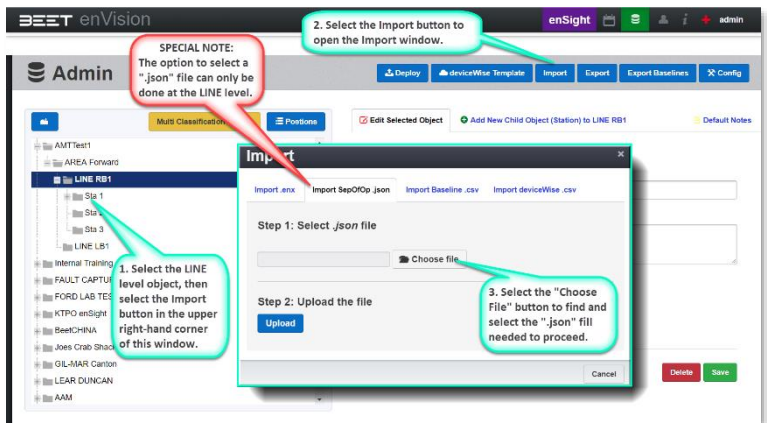


Open this configuration file. To the far right is the Create File button. Select it to start the building of the Factory Tree.

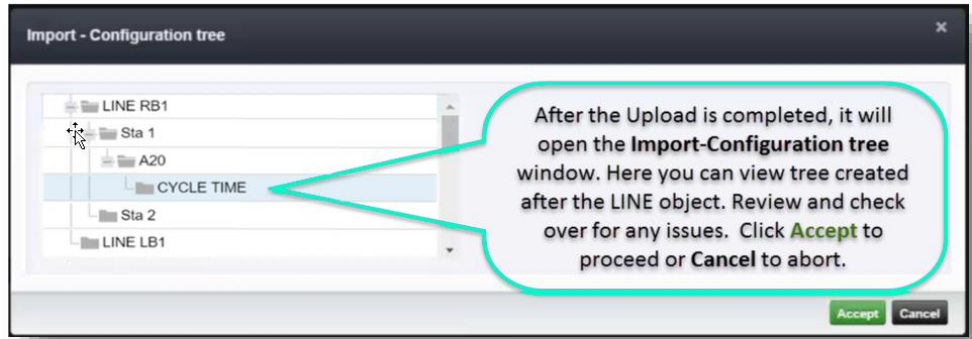
Select the **LINE** level object. **Click** the **Import** button and the Import window will open. Select the Choose File button to find and select the .json file.

Navigate to find the .json file that was created from the .xls file with all the information for ASSET level objects and below. Select **Open**, and this will fill in the Select .json file edit box in the Import window.

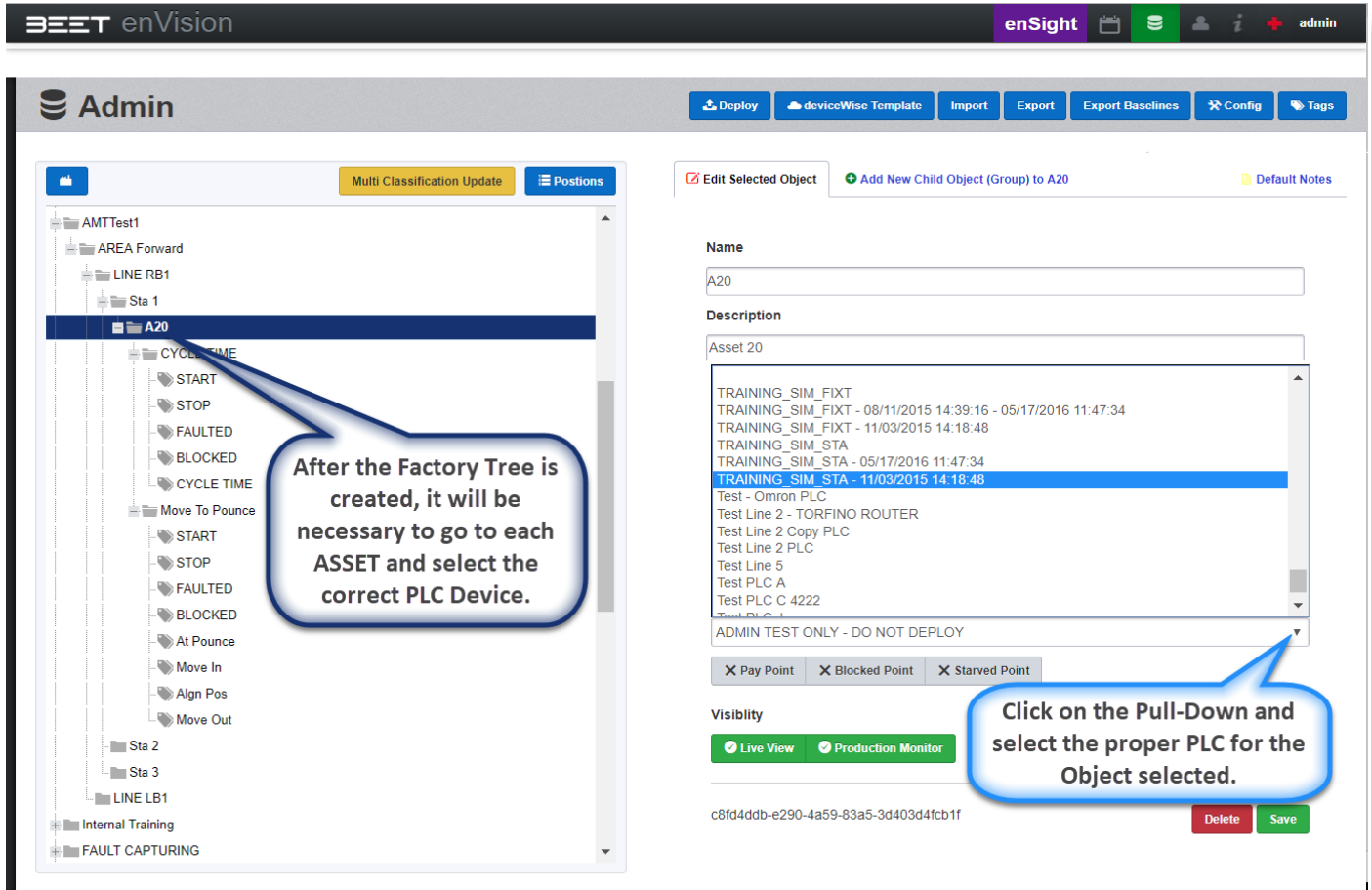
**Click** the **Upload** button to the build Import relationship tree. A "Building import relationship tree..." screen will appear. When it is finished, it will open a preview window.



After the Factory Tree preview is ready, an **Import – Configuration tree** window will open. Review the projected Factory Tree. Check that all the ASSETS and GROUPS are there. **Click** the **Accept** button if you would like to proceed.



After the **Factory** Tree structure is created, it is necessary to select the PLC for the Objects selected. Use the pull-down menu to select the correct PLC Device.

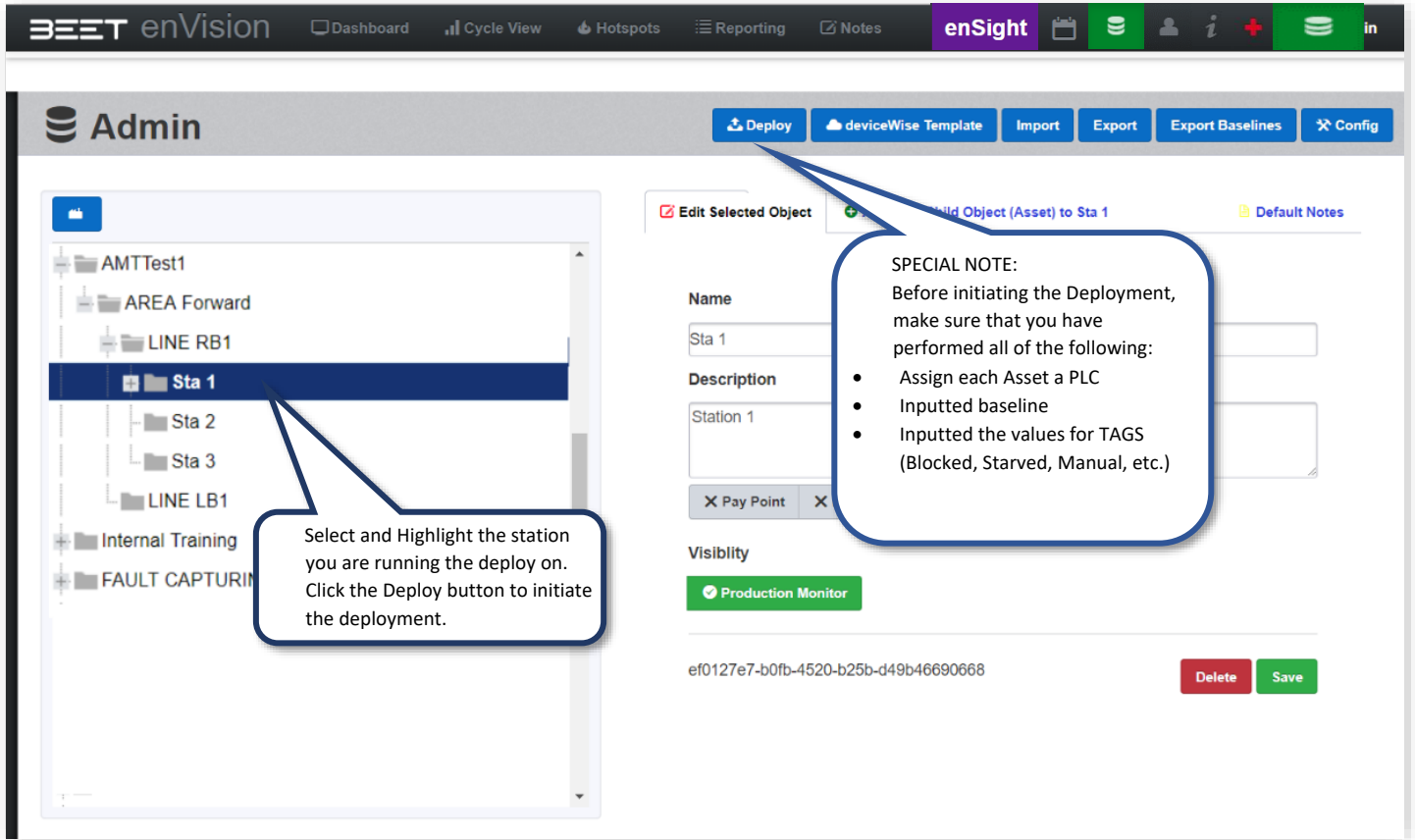


# Deploy

The **Deploy** button is used when everything is setup completely or a change has been made, and is ready to be installed into enVision.

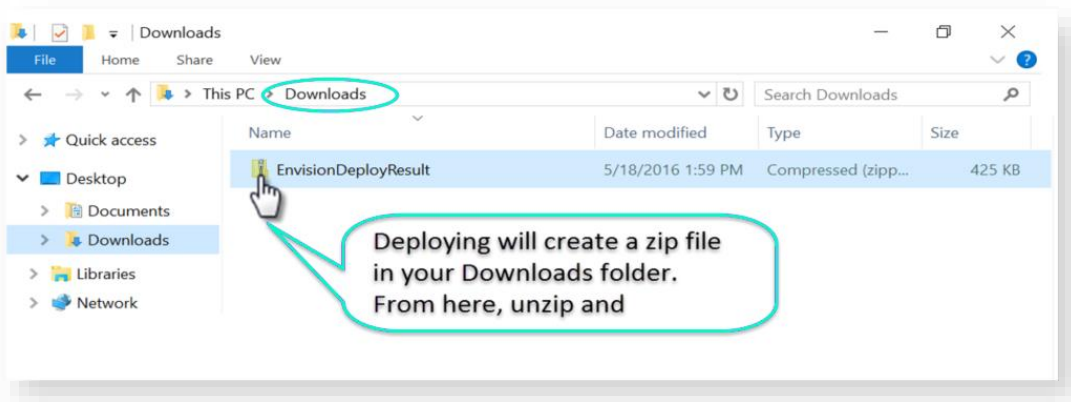
**NOTE: BEFORE** you Deploy make sure you have done all of the following:

1. Assigned each Asset a PLC
2. Inputted baseline values
3. Inputted values for TAGS (Blocked, Starved, Manual, etc.)



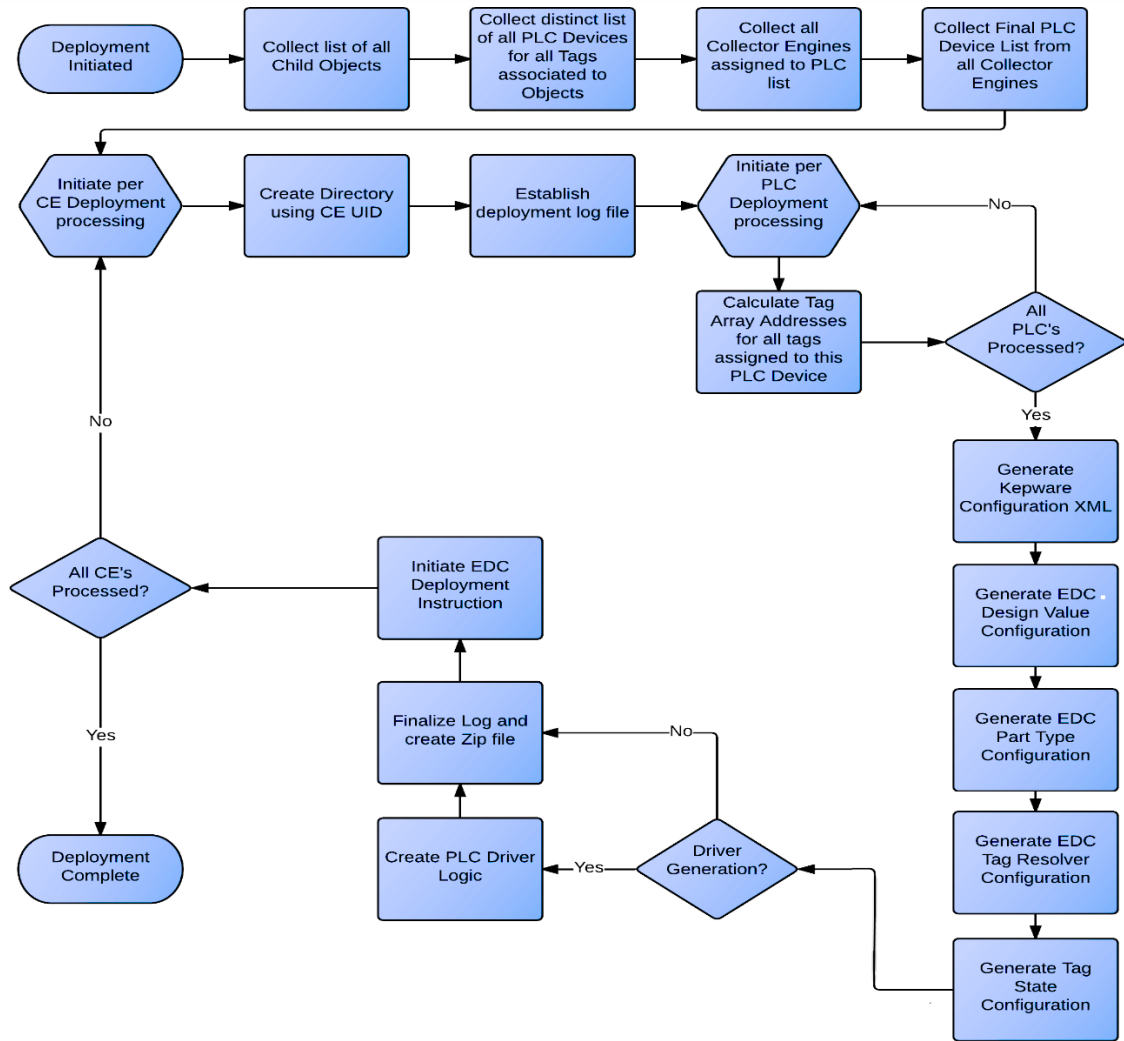
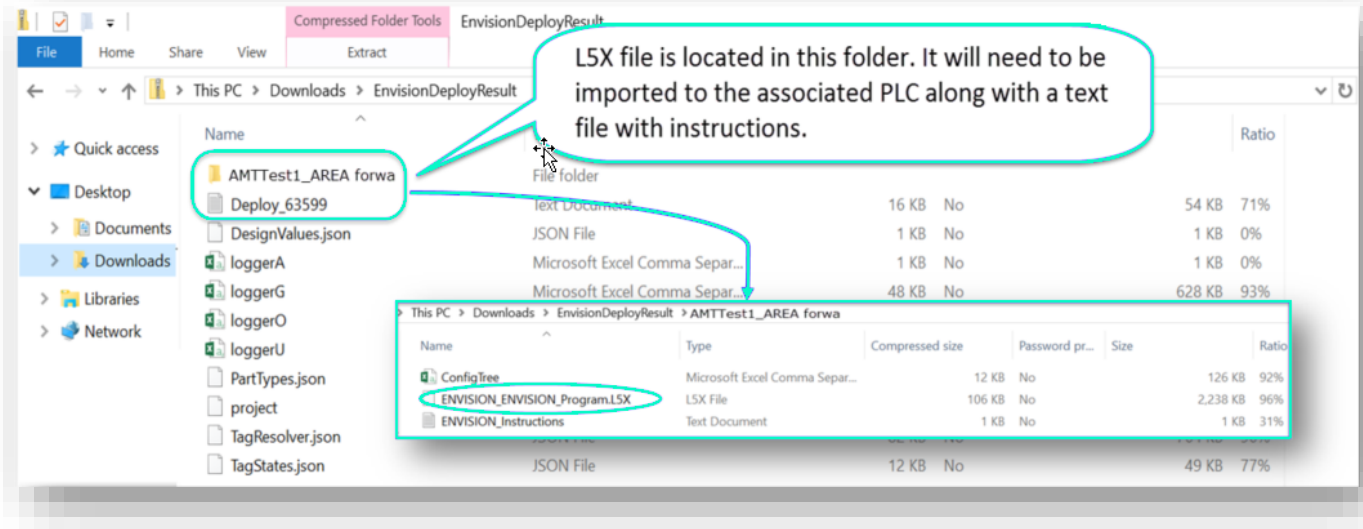
The deploy may take some time to complete. You will be prompted when it's finished.

When the Deploy finishes successfully, you will see a zip file created in your computers **Downloads** folder. It will have a name of **enVisionDeployResult (##).zip** where ## is a number to distinguish between multiple Deploy files. Unzip the folder to view the contents.





Inside each folder you will find a **L5X** file that will need to be imported to the associated **PLC** along with a text file that contains the instructions for this task. If there is a **ConfigTree.CSV** file that is in the folder, it can be ignored.



# Exporting

[Factory Window](#) | [Building a Factory Tree](#) | [Adding a Group or OPs](#) | [Start/Stop Signals](#) | [OPS Signals](#) | [Importing](#) | [Exporting](#) | [Configuration](#)

## Export

The Export button is a feature that allows you to export an object from a Plant level to Group level. First, select the object you would like to Export (e.g. **AREA Forward**). Now select the Export button. A green **Notice** window will open, showing that the Export has been initiated. A few moments later, a download dialog window will open in the lower left-hand side of the browser window. Select open to view the .enx file or view it in the Downloads folder.

**1.** Select the object you would like to Export. Once it is highlighted, you can now select the Export button.

**2.** Click on the Export button to export the selected object in the Factory Tree. This will start a download that will create a .enx file. A green Notice Window will open that will show that the download

**3.** Select the pull down tab to open the download menu.

**4.** Select the Open to view the .enx or "Show in folder" to view it in the Download folder.

Name	Date modified	Type	Size
AREA Forward	7/7/2016 8:25 AM	ENX File	10 KB
A20	7/7/2016 8:16 AM	ENX File	9 KB
ENVISION Administration Manual V23	7/6/2016 7:21 AM	PDF File	13,124 KB
ENVISION Administration Manual	6/30/2016 2:54 PM	PDF File	13,124 KB
ENVISION Administration Manual	6/30/2016 2:48 PM	PDF File	9,518 KB

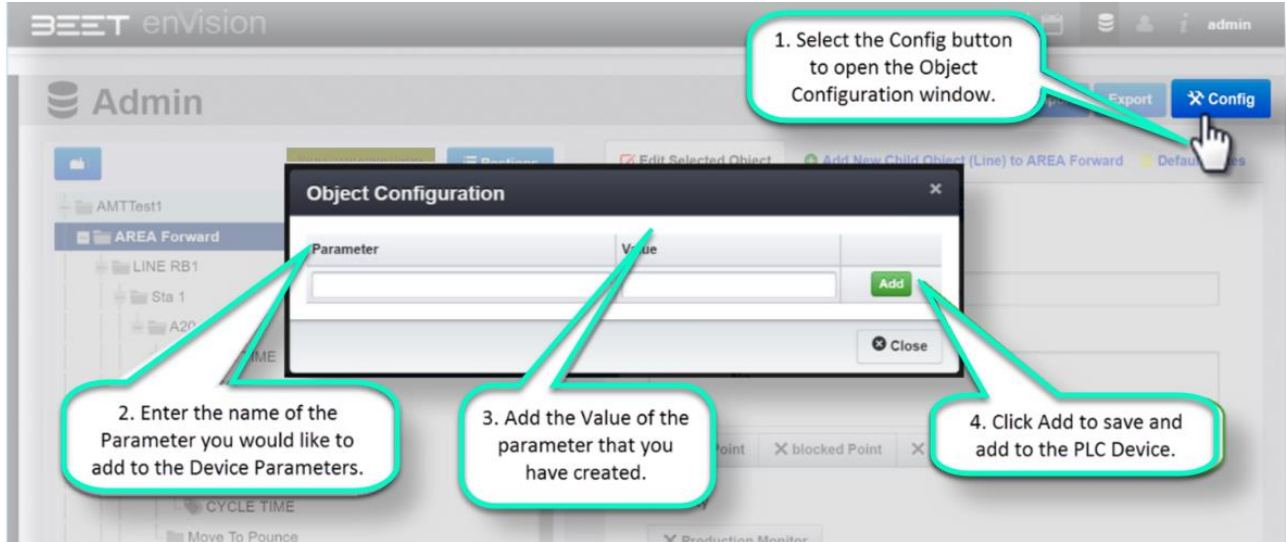
The .enx can be selected and view from Downloads folder.

# Configurations ▽

[Factory Window](#) | [Building a Factory Tree](#) | [Adding a Group or OPs](#) | [Start/Stop Signals](#) | [OPS Signals](#) | [Importing](#) | [Exporting](#) | [Configuration](#)



In the event of new installations, it may be needed that the Parameters used in the PLC may require a modification or a change to make it function correctly. For that, there is the **blue** **Config** button. The Config button is used to add Parameters for the Device section of the PLC. First, select the **Config** button on the upper right side of the browser window. An Object Configuration window will appear. Input the Parameter (name) and the Value for the created parameter. Then click the Add button.

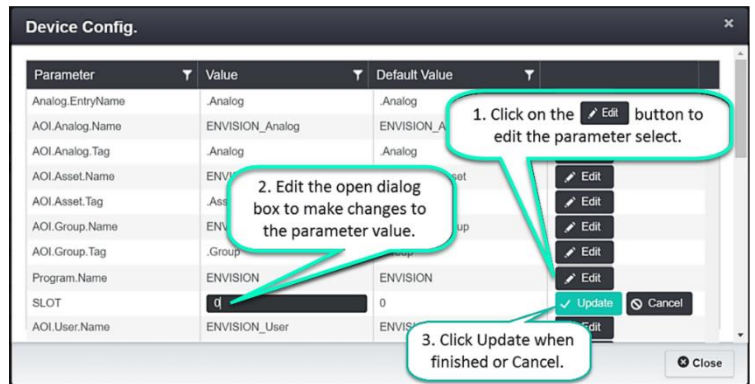
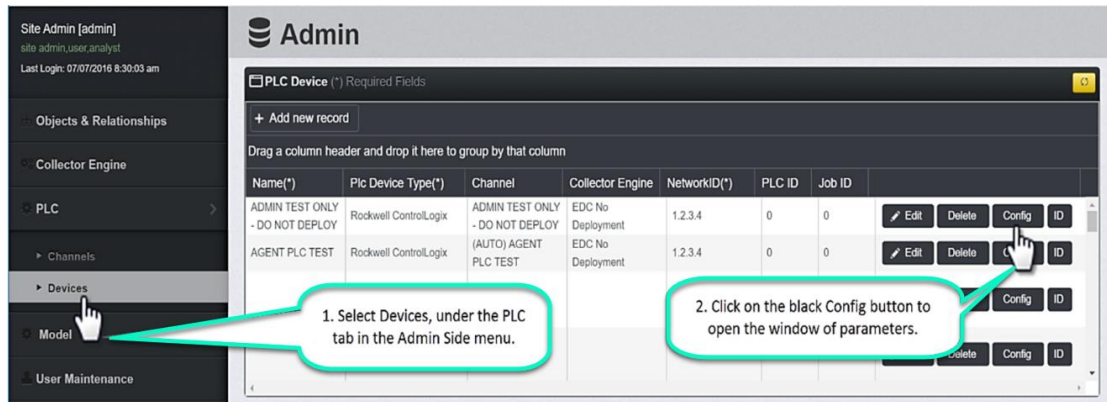


Go to the side menu and select Devices under the PLC tab. This will open the PLC Device page, that will list all the PLC Devices that are in the system currently. To view the created parameter, click on the **Config** button to open a window of the parameter for the selected PLC Device.

Sometimes an installed PLC device needs to have modified parameters to work correctly or to be *tuned* to the PLC. To make changes or modify a parameter you can use the **Edit** button.

Click the **Edit** button to open the **Device Config. Window**.

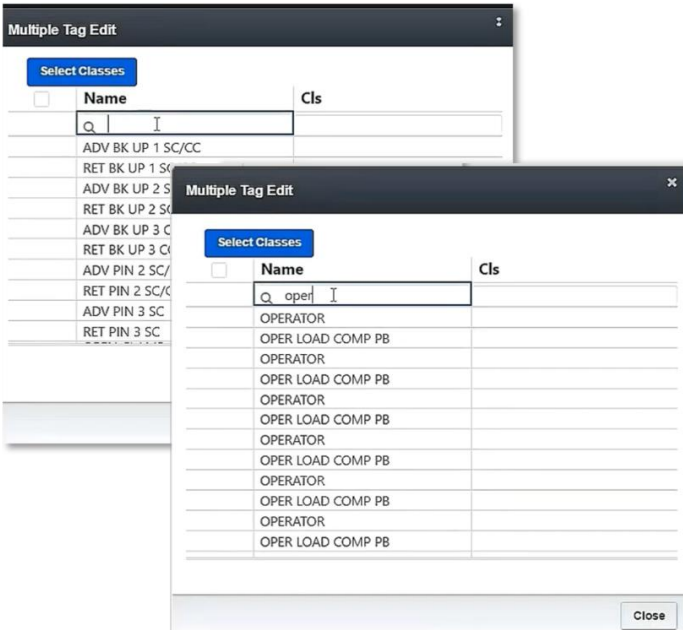
That will open an editing box for the parameter that has been selected. You can now make a change to the parameter to fine tune the parameter for a specific PLC Device.



# Multiple Classification Update

## Multiple Classification Update

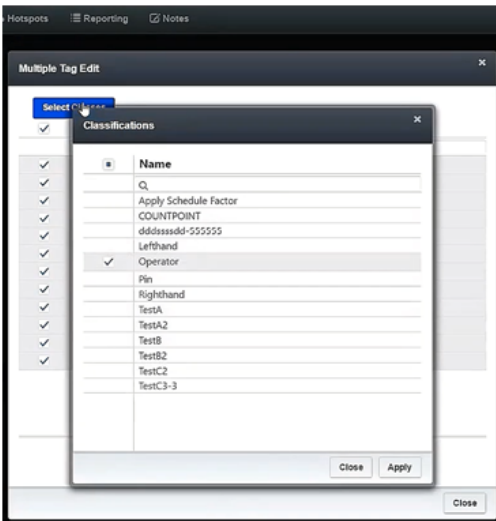
In the Admin section, you can drill down to the LINE level. At the LINE level, you can update Multiple Classification by selecting the yellow Multi Classification Update button. From there, a Multiple Tag Edit window will reveal.



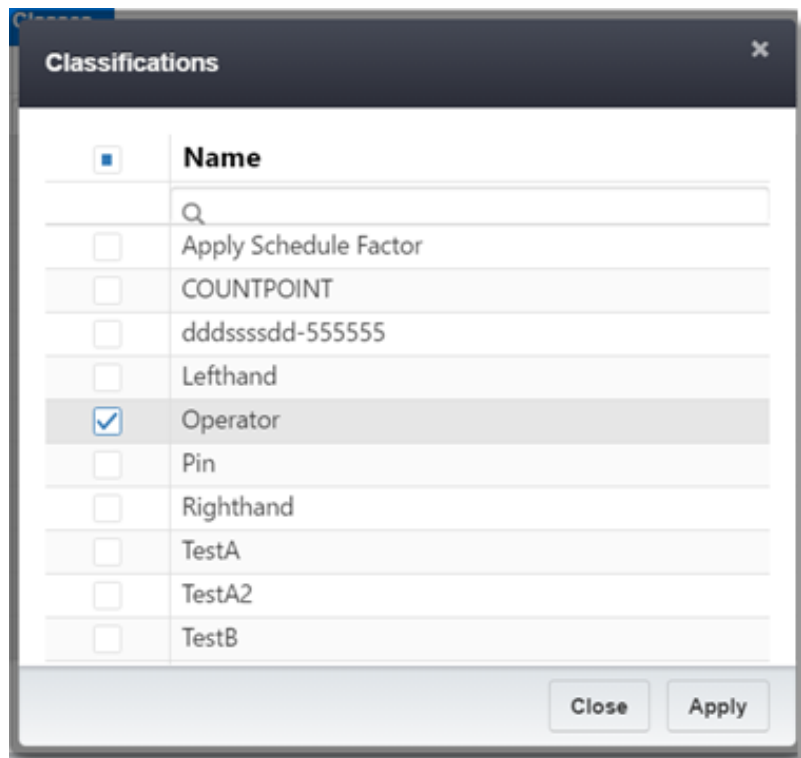
In the Multiple Tag Edit window, click on the edit box and start typing the tag name you would like to view. Typing in the word starting "oper" will cause the list below to populate with tags of the similar spelling (ex. OPERATOR, OPER LOAD COMP PB).

Once the list is revealed, you select individual names or simply select the check all box above the list to the left of the Name column. NOTE: You may also perform the same multiple function on the Cls (Classification) column as well, if the named tags

have several different classifications. Once your selection is completed, click the **Select Classes** button on top of the list.



After the selection of the Select Classes button, the Classification window will open. From there, you can select the classifications you would like to include with your selection (Operator). For this example, we click on the Operator Classification.

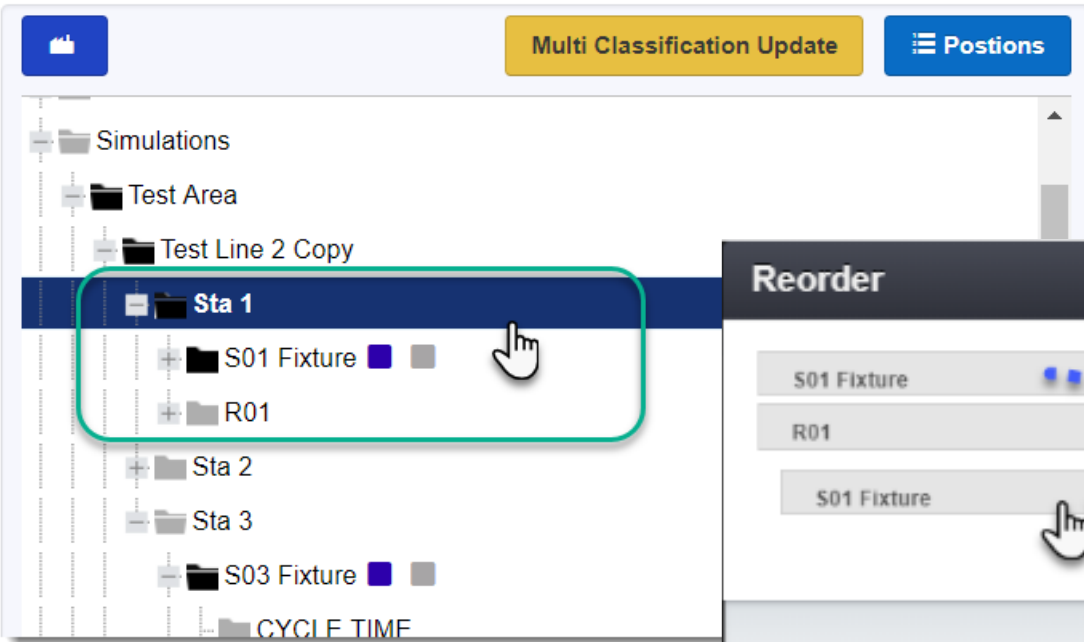




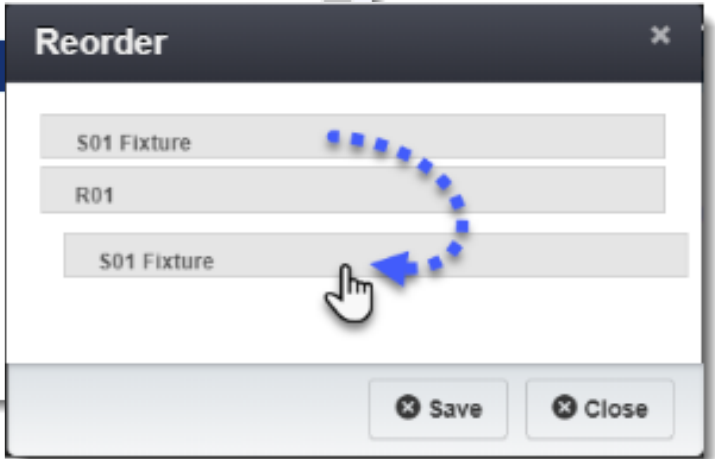
# Positions



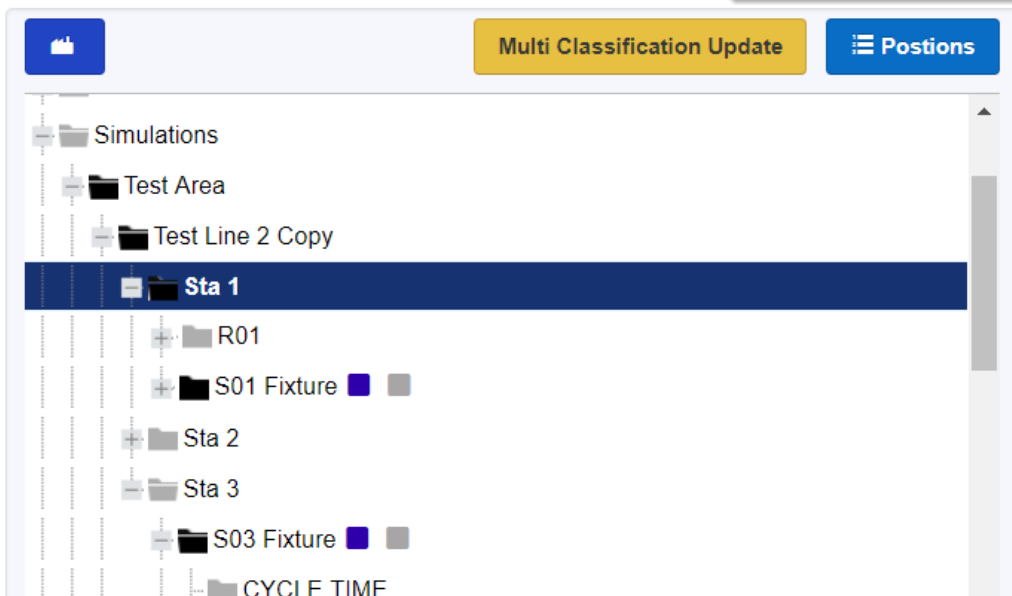
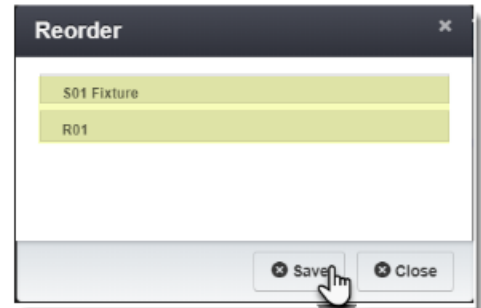
The Position button will allow the User to Reorder the positions of objects in the Factory Tree.



Clicking it will open a Reorder window. From there, the user can click and hold, the move the object in the order needed.



The objects new positions will be highlighted. Click Save to complete the Reorder.



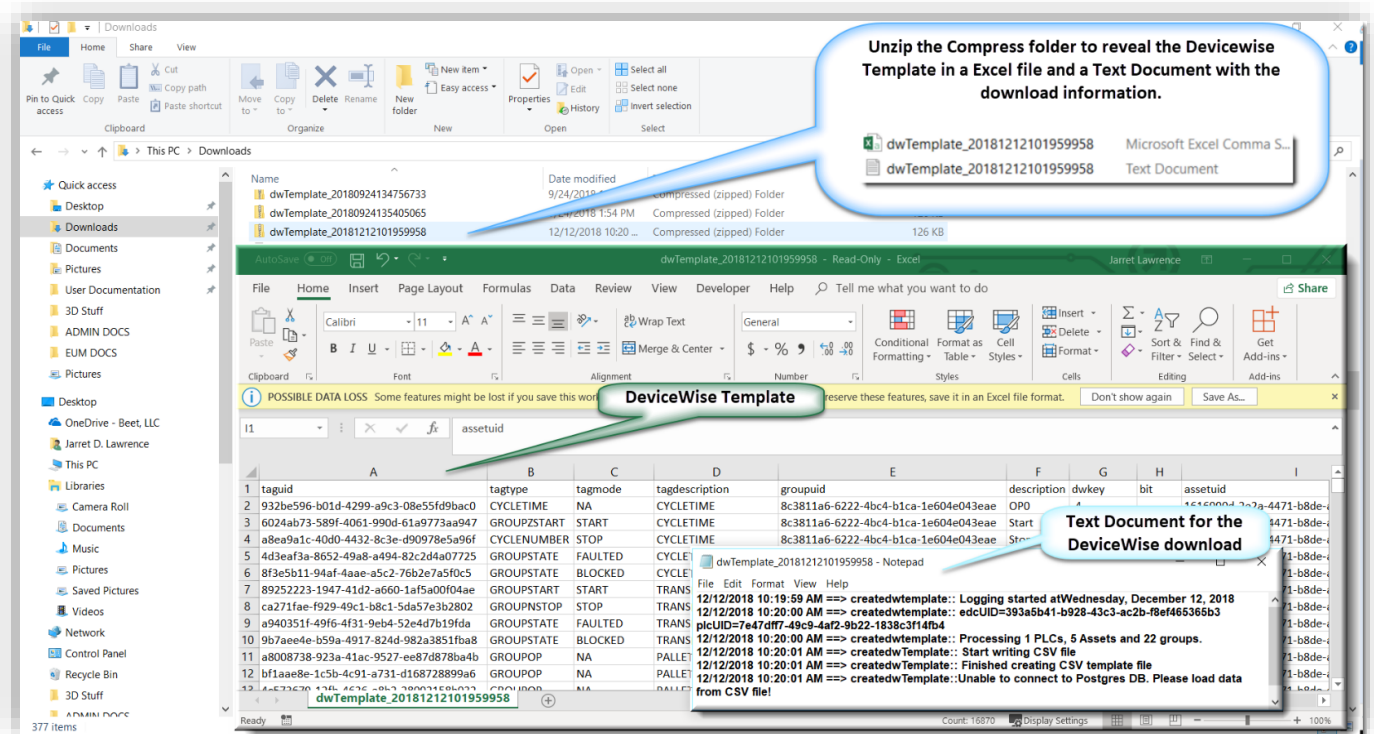
# DeviceWise Templates

In the Objects and Relationships sections, it is an option to generate a deviceWise template. At the Line Object level to the OP level, there is a deviceWise Template button located along with the other option button on the right side of the Objects and Relationships window.

Select an object on the Factory Tree (Object Tree), then select the deviceWise Template button. This action will initiate a download of the deviceWise Template and a text file of the download. A green Notice placard will appear, show that the download has initiated.



In the lower left side of the window will reveal the download window placard, showing the dwTemplate.zip file. From here, you can open, view in the folder, or cancel the download. In the download window, there will be an excel file and a text file.



# Add Default Note

When initially building a factory, it may be necessary to make and take notes that needed to be made during the building and shakedown of a factory tree. These notes will be available and attached to the object in which it was placed.

To access these notes, simply click on the Default Notes tab, on the right side of the Objects and Relationships window. When selected, the Add Default Note window will reveal. Any previous notes will be displayed below the note editing section.

Enter a detailed note in the “Enter Note Here” box. This sections allows a large amount of room for your note. Once the note is added, click on the pulldown in the Category section. Make a choice from the list of categories present. The user can attach a file to the note as a reference, with a file size up to 15MB. When finished, click the Save button to complete your note. The note will appear below the editing section.

To edit a note, simple click the editing icon and the “Edit Default Notes” window will appear. Make your changes, then click Save.

To Delete a not, click on the Delete Icon, a warning box will appear asking, “Are you sure you want to delete?” Click OK to delete or Cancel.



# I/O Expressions [▽](#)

## Code Explanation

### AssetStart

This bit is handled by the Cycle Time Group AOI (**Add-On Instructions**) and acts as echo of the Cycle Time Group Start. The Groups after the Cycle Time Group belonging to the same Asset will see the actual state of the AssetStart bit in the same PLC scan (only in the cases where AssetStart is used in any Group IO Expression), however, since the Asset is scanned before the Cycle Time Group, it will take one more scan for the Asset to see the transitions of the AssetStart bit.

### AssetStarted

This bit is handled by the Cycle Time Group AOI, it is turned on when an idle (stopped) group is started by the Cycle Time Group Start and turned off on the Cycle Time Group Stop.

The Groups after the Cycle Time Group belonging to the same Asset will see the actual state of the AssetStarted bit in the same PLC scan (only in the cases where AssetStarted is used in any Group IO Expression), however, since the Asset is scanned before the Cycle Time Group, it will take one more scan for the Asset to see the transitions of the AssetStarted bit

### AssetStartOS

This bit is handled by the Cycle Time Group AOI, it is turned on when the Cycle Time Group goes from not start to Started, stays on for one PLC scan and then turned off.

The Groups after the Cycle Time Group belonging to the same Asset will see the actual state of the AssetStartOS bit in the same PLC scan (only in the cases where AssetStartOS is used in any Group IO Expression), however, since the Asset is scanned before the Cycle Time Group, it will take one more scan for the Asset to see the state of the AssetStartOS bit

### AssetRunning

This bit is handled by the Cycle Time Group AOI, it is turned on together with the Cycle Time Group Started bit and turned off when the Cycle Time Group Op0 is triggered.

The Groups after the Cycle Time Group belonging to the same Asset will see the actual state of the AssetRunning bit in the same PLC scan (only in the cases where AssetRunning is used in any Group IO Expression), however, since the Asset is scanned before the Cycle Time Group, it will take one more scan for the Asset to see the transitions of the AssetRunning bit"

### AssetDone

This bit is handled by the Cycle Time Group AOI, it is turned when the Cycle Time Group Op0 is triggered and turned off on the Cycle Time Group Stop.

The Groups after the Cycle Time Group belonging to the same Asset will see the actual state of the AssetDone bit in the same PLC scan (only in the cases where AssetDone is used in any Group IO Expression), however, since the Asset is scanned before the Cycle Time Group, it will take one more scan for the Asset to see the transitions of the AssetDone bit.

### AssetStop

This bit is handled by the Cycle Time Group AOI and acts as echo of the Cycle Time Group Stop.

The Groups after the Cycle Time Group belonging to the same Asset will see the actual state of the AssetStop bit in the same PLC scan (only in the cases where AssetStop is used in any Group IO Expression), however, since the Asset is scanned before the Cycle Time Group, it will take one more scan for the Asset to see the transitions of the AssetStop bit.

### AssetStopOS

This bit is handled by the Cycle Time Group AOI, it is turned on when a STARTED Cycle Time Group is stopped, stays on for one PLC scan and then turned off.

The Groups after the Cycle Time Group belonging to the same Asset will see the actual state of the AssetStopOS bit in the same PLC scan (only in the cases where AssetStopOS is used in any Group IO Expression), however, since the Asset is scanned before the Cycle Time Group, it will take one more scan for the Asset to see the state of the AssetStopOS bit

## AssetMem

This bit is handled by the Asset AOI, it is turned on by the Asset SetUserMem Entry and turned off on either the AssetStop or the Asset ResetUserMem Entry.

All Groups after the Asset AOI belonging to the same Asset will see the actual state of the AssetMem bit in the same PLC scan (only in the cases where AssetMem is used in any Group IO Expression), however, if the AssetMem is used in any IO Expression of the same Asset, it will take one more scan for the AssetMem to show transitions

## GroupStarted

This bit handled by the Group AOI, it is turned on when an idle (stopped) group is started by the Group Start and turned off on the Group Stop.

If the GroupStarted bit is used in any IO Expression of the same Group, it will take one more scan for the GroupStarted to show transitions for those IO Expressions.

## GroupStartOS

This bit handled by the Group AOI, it is turned on when the Group goes from not started to Started, stays on for one PLC scan and then turned off.

If the GroupStartOS bit is used in any IO Expression of the same Group, it will take one more scan for the GroupStartOS to show transitions for those IO Expressions.

## GroupRunning

This bit is handled by the Group AOI, it is turned on together with the Group Started bit and turned off when the all the configured Ops in the Group are triggered.

If the GroupRunning bit is used in any IO Expression of the same Group, it will take one more scan for the GroupRunning to show transitions for those IO Expressions.

## GroupDone

This bit is handled by the Group AOI, it is turned when all the configured Ops in the Group are triggered and turned off on the Group Stop.

If the GroupDone bit is used in any IO Expression of the same Group, it will take one more scan for the GroupDone to show transitions for those IO Expressions.

## GroupOccurCount

This byte is handled by the Group AOI, it counts up when on the GroupDone and reset on the AssetStopOS.

If the GroupOccurCount bit is used in any IO Expression of the same Group, it will take one more scan for the GroupOccurCount to show changes for those IO Expressions.

## GroupStopOS

This bit is handled by the Group AOI, it is turned on when a STARTED Group is stopped, stays on for one PLC scan and then turned off.

If the GroupStopOS bit is used in any IO Expression of the same Group, it will take one more scan for the GroupStopOS to show transitions for those IO Expressions."

## GroupMem

This bit is handled by the Group AOI, it is turned on by the Group UserMem Entry and turned off on either the GroupStop.

If the GroupMem bit is used in any IO Expression of the same Group, it will take one more scan for the GroupMem to show transitions for those IO Expressions.

enVision IO Expression Shortcuts [▽](#)**Asset Status bits**

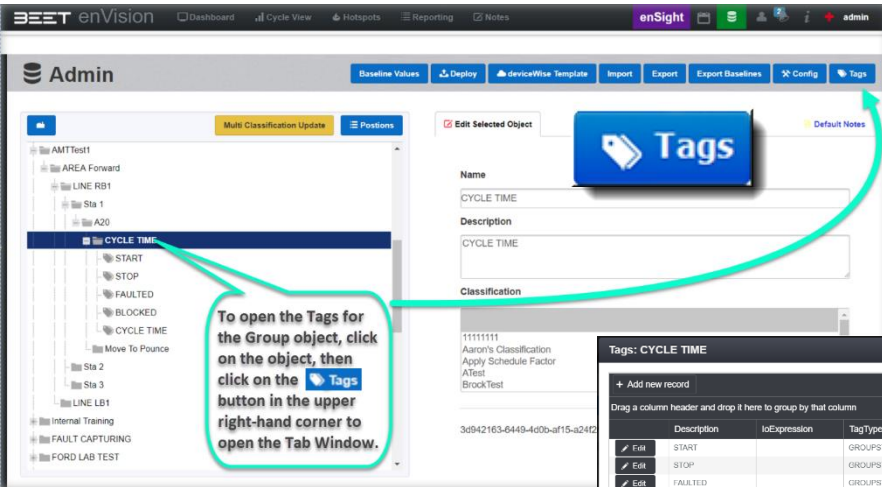
Shortcut (Not Case Sensitive)	Description
ASSETMEMORY or ASSETMEM	This is turned on by the Asset SetUserMem Entry and turned off by either the AssetStop or the Asset ResetUserMem Entry. All Groups that belong to the same Asset use the same AssetMem.
ASSETSTART	This is an echo of the CycleTime group's START conditions. All Groups that belong to the same Asset use the same AssetStart.
ASSETSTARTOS	This is a one-time trigger set on the first OFF→ON transition of the CycleTime group START following a STOP. All Groups that belong to the same Asset use the same AssetStartOS.
ASSETSTOP	This is an echo of the CycleTime group's STOP conditions. All Groups that belong to the same Asset use the same AssetStop.
ASSETSTOPOS	This is a one-time trigger set on the first OFF→ON transition of the CycleTime group STOP following a START. All Groups that belong to the same Asset use the same AssetStopOS.
ASSETSTARTED	This is turned on when an idle (stopped) Asset is started by the Cycle Time Group Start and turned off on the Cycle Time Group Stop. All Groups that belong to the same Asset use the same AssetStarted.
ASSETRUNNING	This is turned on together with AssetStarted and turned off when the Cycle Time Group Op0 is triggered. All Groups that belong to the same Asset use the same AssetRunning.
ASSETDONE	This is turned on when a STARTED Cycle Time Group Op0 is triggered and turned off on the Cycle Time Group Stop. All Groups that belong to the same Asset use the same AssetDone.

**Group Status bits**

Shortcut (Not Case Sensitive)	Description
GROUPMEMORY or GROUPMEM	This is turned on by the Group SetUserMem Entry and turned off by the GroupStop Entry
GROUPSTARTOS	This is a one-time trigger set on the first OFF→ON transition of the Group START following a STOP.
GROUPSTARTED	This is turned on when an idle (stopped) group is started by the Group Start and turned off on the Group Stop.
GROUPRUNNING	This is turned on together with GroupStarted and turned off when all configured Group OPs are triggered.
GROUPDONE	This is turned on when all configured OPs of a STARTED Group are triggered and is turned off by the Group Stop.
GROUPSTOPOS	This is a one-time trigger set on the first OFF→ON transition of the Group STOP following a START.

# Group Tag Explanation

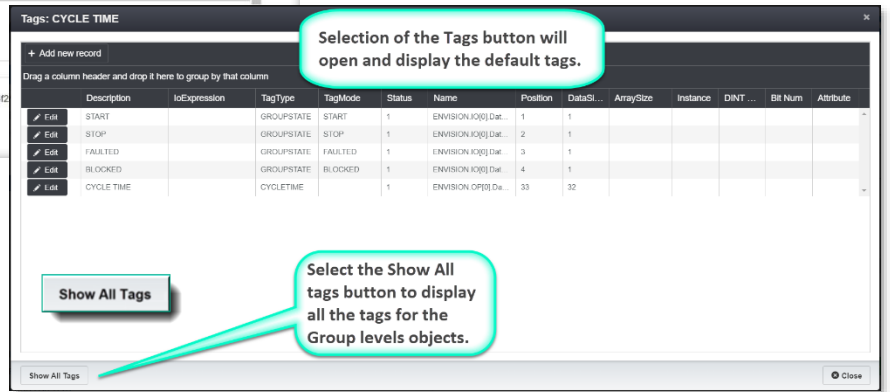
At the Group level, every object has a tag that can be configured to what you need it to achieve. To view and configure these Group Tags, you need to **select** the group object you wish to view. In the upper right-hand corner is the **Tags** Button.



To open the Tags for the Group object, click on the object, then click on the Tags button in the upper right-hand corner to open the Tab Window.

The Tags: (CYCLE TIME) window will open and display the default parameters associated with that object selected. Whenever a Group object is created, these Standard Tags are automatically attached to the Group objects created.

There are more Tags available for you to utilize. Simply **click** on the **Show All Tags** button, located on the lower left side of the Tags Window.



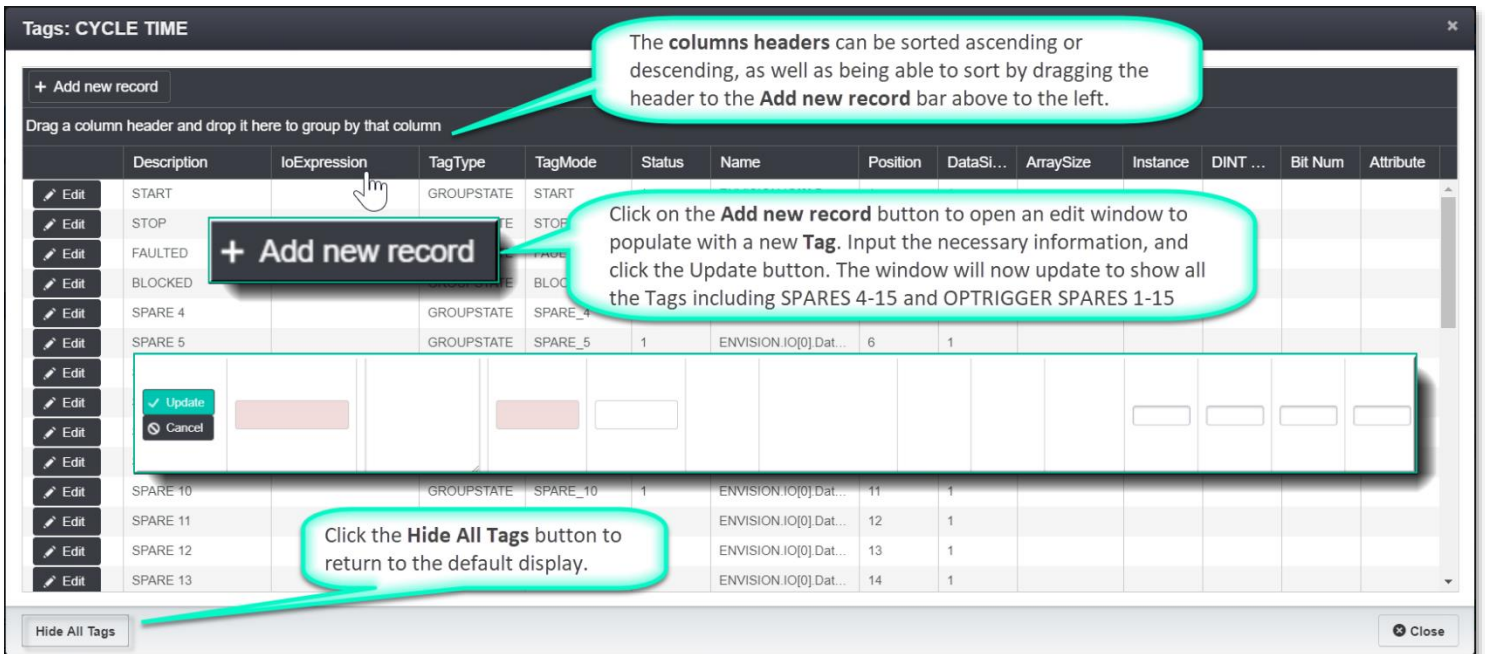
Selection of the Tags button will open and display the default tags.

Show All Tags

Select the Show All tags button to display all the tags for the Group levels objects.

The Tags window will now update with all the available tags to be utilized as needed. The tags provided, are additional tags labeled Spare 4-15 and OPRIGGER SPARES 1-16. You can also add a new Tag by selecting the + Add new record button. Fill out the parameter fields and click the **Update** button.

The **Column headers** can be dragged and dropped to sort the tags by the header selected. Simply **click** and **hold** the column header, then drag it up to the column field above the headers and then release. It will now sort the tags by the selected header field. They also can be sorted by ascending or descending, by simply clicking on the Column Header icon.



The columns headers can be sorted ascending or descending, as well as being able to sort by dragging the header to the **Add new record** bar above to the left.

+ Add new record

Click on the **Add new record** button to open an edit window to populate with a new **Tag**. Input the necessary information, and click the Update button. The window will now update to show all the Tags including SPARES 4-15 and OPRIGGER SPARES 1-15



Click the **Hide All Tags** button to return to the default display.

Hide All Tags

Close

## Add New Record

If another tag needs to be created or needs to be edited, you can select a tag from the list (SPARES or OPTRIGGERS SPARES), then click the **+ Add new record**. The selected window will expand into an editing field to make you changes. Under the **Description**, input a name to represent the tag. Input the **I/O expression**, **Tag type**, and **Tag Mode**. If everything is set and ready, click the **✓ Update** button, to save changes or **Cancel** to discard changes and go back to the list. If you want to change the I/O Expression only, you can select the Edit button and the selected tag will open an editing box to make your changes. When finished, click **✓ Update** to Save or **Cancel** to discard changes.

The screenshot shows a window titled "Tags: CYCLE TIME" with a table of tags. A callout points to the "+ Add new record" button, stating: "To add a new Tag, select one of the SPARES's or OPTRIGGER SPARES, then click the + Add new record button, and a editing field will appear as seen below." Another callout points to the expanded form, stating: "Input the **Description**, **I/O Expression**, **Tag Type**, **Instance**, **DINT**, **Bit Number**, and **Attribute**. If you select a default tag (e.g. START), it will allow the user to only change the I/O Expression. Click **Update** to Save the changes or **Cancel** to discard." The table has columns: Description, IoExpression, TagType, TagMode, Status, Name, Position, DataSI..., ArraySize, Instance, DINT..., Bit Num, Attribute. The expanded form has fields for Description, IoExpression, TagType, TagMode, Status, Name, Position, DataSI..., ArraySize, Instance, DINT..., Bit Num, Attribute, and buttons for Update and Cancel.

## Group Tag Definitions

### START

START is used to indicate that the step has been initiated. Solenoid valves, robot initiate, operator clear to enter, etc. are examples of common Start signals. The START signal is a retentive signal and goes true upon the first OFF / ON transition. It is not reset until the Group STOP signal goes true. The enVision PLC driver stores the current Timestamp during the scan that the transition is detected.

### STOP

STOP is used to reset the Group and clear all retentive signals and memory bits in the PLC driver that are associated with that Group. The only Group that **MUST** have an expression entered for its STOP signal is the CYCLE TIME Group. This STOP signal has special meaning to the enVision application. It turns on the Asset Stop signal inside the PLC driver that indicates to enVision that an Asset has completed a cycle and that all the data captured by the application since the Start of the Asset can be processed and stored as historical data. All Groups other than the CYCLE TIME Group have the option of leaving the STOP signal blank. The Deploy function that generates the PLC driver logic is designed to automatically use the Asset Stop signal to fill in any Group STOP signal that is left blank. It is strongly recommended to leave Group STOP signals blank IF the Group is expected to occur only once per cycle. This ensures that all Groups get reset at the end of a cycle.

There are a couple of exceptions touched upon earlier where it may be desirable to capture multiple occurrences of a Group during one cycle of an Asset such as monitoring how many times an operator moves in and out of a light screen and the gaps between each occurrence. For this scenario, it is necessary to enter an expression for the STOP signal in order for the Group to reset before the cycle ends.

### FAULTED

The object is faulted and stop working.

## BLOCKED

The object completes its work and is waiting for downstream objects to finish so the finished part can move out of the object.

## CYCLETIME

Amount of time between the start and stop of an object.

## SPARES (4-15)

The Spares are used to create custom Group Tags when you are required to have more than the default tags.

## OPTRIGGER SPARE (1-16)

The OPTRIGGER SPARE is used to create a custom Op Trigger

## Asset Tag Explanation [▽](#)

### AUTO

The object is in a full auto or normal operation mode.

### MANUAL

The Asset is running manually or being engaged for a maintenance or troubleshooting mode.

### FAULTED

Tooling or Automation failure, that may require manual intervention.

## BLOCKED

The Asset has finished its work and cycle but cannot transfer its finished product to the next position.

## PRIMARY STARVED

The overall time the object is in starved state. The object completed its job and the produced part has exited the object. The object is waiting for the next part to start working.

### RUNNING

The Asset is performing its designed functionality

### IDLE

The asset has done its function.

### SECONDARY STARVED

The system is waiting for parts that are loaded at this asset.

### INTERRUPTED

An asset cycle is interrupted due to violation of safety setup or other non-tooling and automation related stoppage.

### NON-PRODUCTION

The system is not engaged or not in production. During this time, no records will be kept.

### USER 01-04

### SPARE 11-27

The Spares are used to create custom Asset Tags.

# enVision Application Server (EAS) ▾

EAS receives the data and information in order to display the complete data taken. It receive data from the EDC which are in contact with the PLC's.

## Components of EDC

### OPC Server (OLE Process Control)

Object linking and embedding (OLE) is a Microsoft technology that facilitates the sharing of application data and objects written in different formats from multiple sources. Linking establishes a connection between two objects and embedding facilitates application data insertion.

Object Linking and Embedding (OLE) for process control (OPC) is a set of standards developed by a joint collaboration of leading automation industry suppliers. OPC's primary mission is to define a uniform interface for use with any organization or custom software package.

Makes it possible to understand and unify the communications between machines.

### Agent

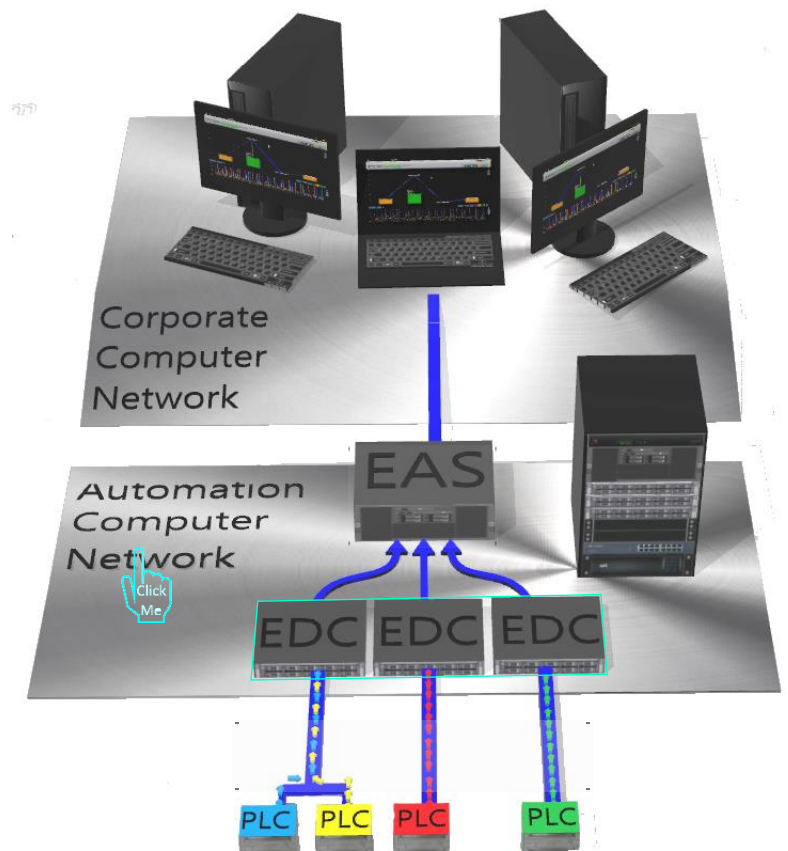
The Agent is a service program, that allows the EDC to talk to EAS. The EAS cannot talk to EDC directly. The agents initiate communication through the web API's (application programming interface) to the EAS to find out what it has to do. The Agent asks the EAS if it needs anything. If the EAS has a job to do, it tells the EDC, and the EDC will start to perform the job. If it needs more information on the job, it will ask the EAS for the necessary information (such as a deploy). When asked by the EDC, the EAS will say it needs to deploy, so the agent executes the Deployment instructions.

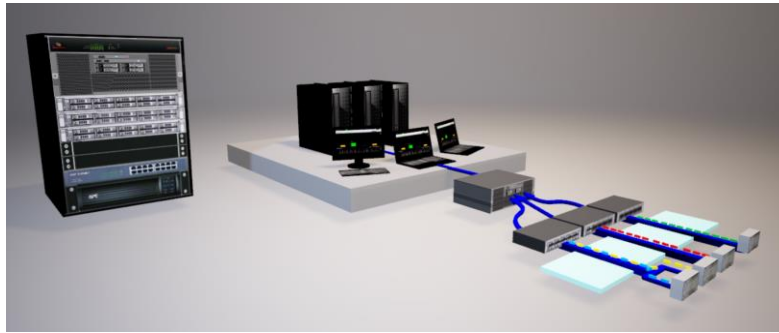
### Data logger

Data Logger is an application of the EDC that logs data from an OPC server to any ODBC-compliant database. Data logger monitors for any data changes in the PLC. It will log the changes through the database and staging the data in a temporary storage area. The data will be accessed and processed by the Data Processor when ready.

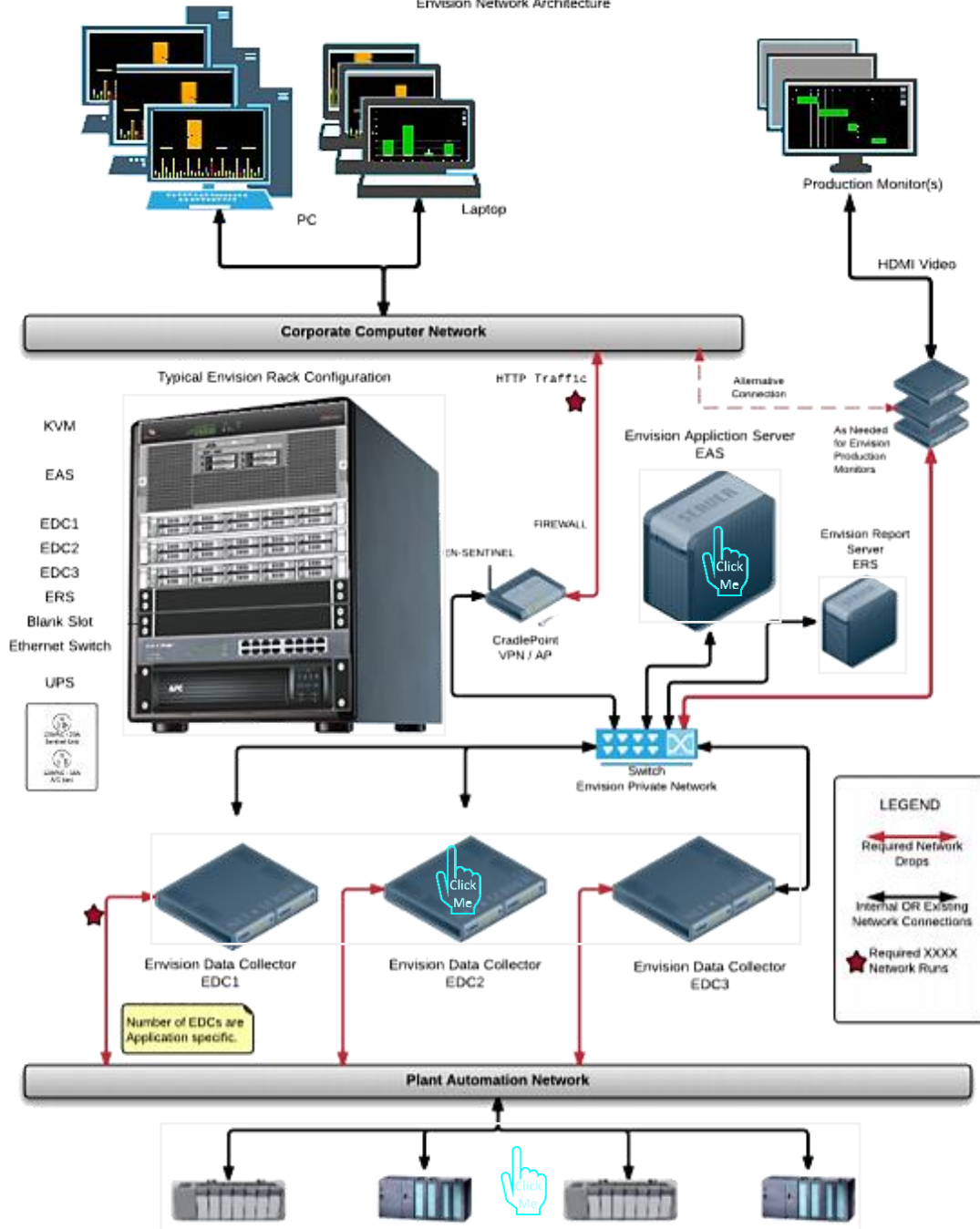
### Data processor

The Data Processor performs the real work and performed the complicated functions needed to complete the process. The Data Processor looks at the data that comes in through the system and the Data logger. It takes the data, organizes it, makes sense of it, and identifies the cycle data. It then analyzes the data, **enVisionizes** it (structures it) and sends it to the enVision Application Server. From there the data can be accessed through the enVision Browser.

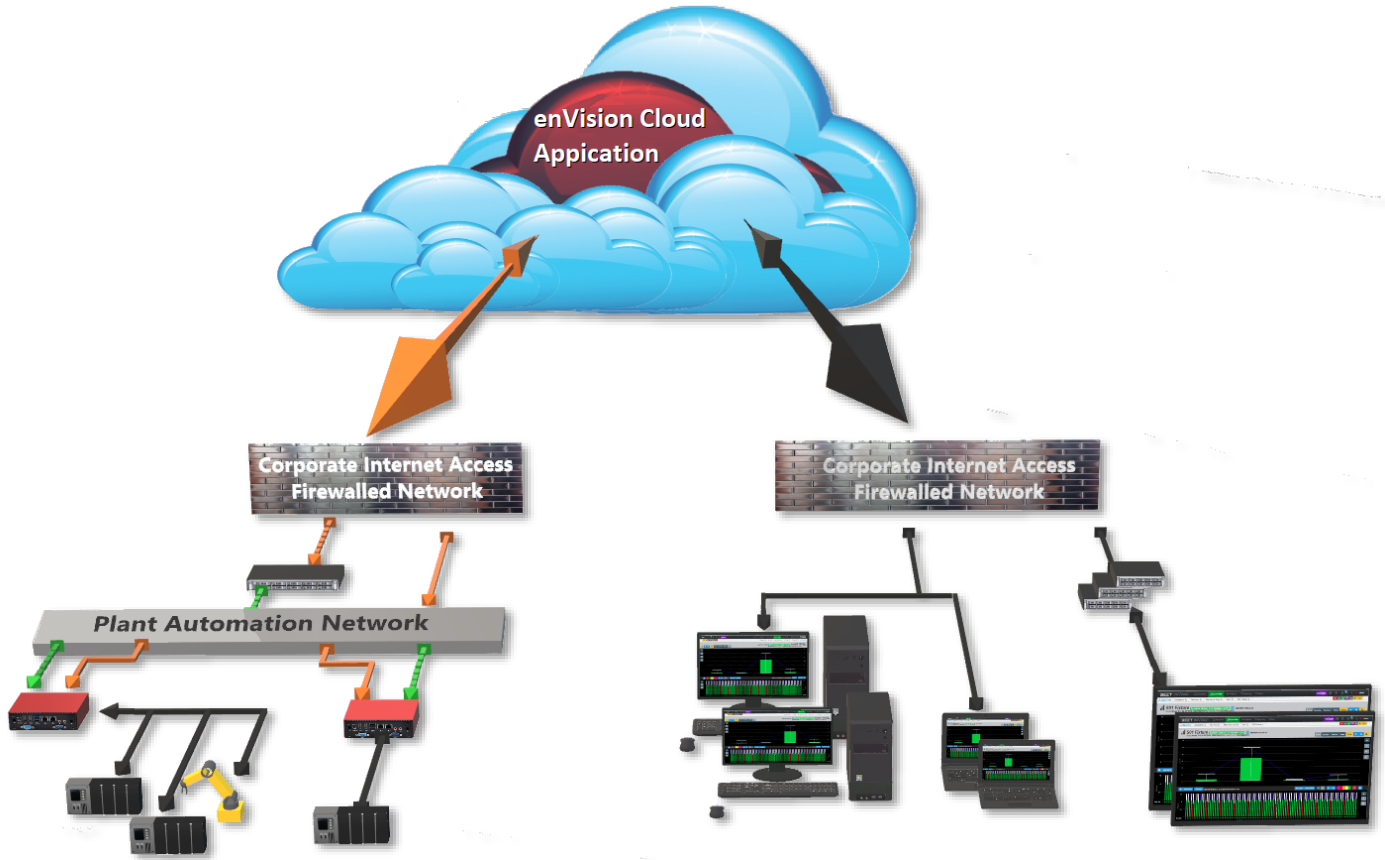




Envision Network Architecture







# Collector Engine

## Overview

The Collector Engine, in enVision, is representation of an **EDC (enVision Data Collector)**. Every collector engine is a representative of an EDC.

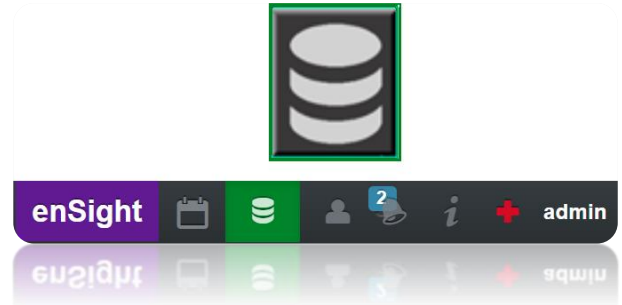
Before you can Deploy your enVision project, you need to configure how enVision will communicate with the PLC device that controls the machine/process that your project is monitoring.

First step is to create a Collector Engine. This defines the actual piece of hardware that will connect to the PLC network and gather data via an OPC (**Keeware** in this case) from the PLC.

Click on the Admin button on the top right-hand side of menu bar.

Slide the cursor over to the left-hand edge of window until side menu appears. This action will open the Admin menu tab.

Click the **Collector Engine** tab to open the Collector Engine Display Table.



**enSight** admin

BEET enVision Dashboard Cycle View Hotspots Reporting Notes

Site Admin [admin]  
site admin, user, analyst  
Last Login: 05/31/2018 9:13:13 am

**Admin**

Objects & Relationships

- Collector Engine
- PLC
  - Channels
  - Devices
- Model
- User Maintenance
- System
  - System Configuration
  - Audit
  - Application Logs
- Subscriptions/Notifications
- API Query

Simulations

- Test Area
  - Test Line 2 Copy
    - Sta 1
      - S01 Fixture
        - CYCLE TIME
        - TRANSFER IN
        - ROBOT LOAD
        - PIN CLAMPS CLOSE
        - TRANSFER OUT
  - R01
  - Sta 2
  - Sta 3
  - Test Line 5
    - DeviceWISE\_TEST
    - Training Master

Slide the cursor over to the left-hand edge of the browser window to open the Admin Menu.

Move the cursor over the menu and click on the **Collector Engine** Tab. This will reveal the Collector Engine Window.

# Adding a New Record

In the window that pops up you can create a new Collector Engine by clicking on the **+ Add new record** button, enter a name for the new engine, and then click the **✓ Update** button. From there, are the Agent Version, Address, and Last connect columns as well. Only the name needs to be added, no other data other than the name needs to be entered.

The user can create a new Collector Engine by clicking on the **+ Add new record** button.

Click the **✓ Update** button to complete your new record.

An editing box will appear. Input the name of the new Collector engine only. The other data will be added later. Click the **✓ Update** button when finished.

Name	Agent Version	Agent Address	Agent Last Connect	Update	Cancel	Edit	Delete	Status	ID
AMT CE 001				✓ Update	Cancel				
AAM G...EDC			09/20/2017 17:43:37				Delete	Status	ID
AAM...			10/12/2017 15:23:49				Delete	Status	ID
AAM TR...AX			05/22/2018 12:05:13				Delete	Status	ID
AGENT TEST	2.2.15316.1170	172.24.1.85	11/12/2015 3:20:20 PM			Edit	Delete	Status	ID
A_A_Rawn			07/11/2017 14:24:51			Edit	Delete	Status	ID
A_CollectEngine_Oliver			04/06/2017 16:09:19			Edit	Delete	Status	ID
ChinaTrainEDC_Lito			03/31/2017 13:14:53			Edit	Delete	Status	ID
DEPLOYMENT TEST			05/26/2016 09:50:55			Edit	Delete	Status	ID
DR_TEST						Edit	Delete	Status	ID
DTP-EDC2						Edit	Delete	Status	ID
Development EDC3						Edit	Delete	Status	ID
EDC - DD Assembly A						Edit	Delete	Status	ID
EDC - DDS						Edit	Delete	Status	ID
EDC - P552 LH BODYSIDE SEG 2			12/16/2016 07:55:05			Edit	Delete	Status	ID
EDC - Tipton			09/17/2015 07:03:42			Edit	Delete	Status	ID
EDC 1			06/08/2017 10:08:24			Edit	Delete	Status	ID
EDC FAULTS	2.1.4511.1178	172.24.1.20	10/20/2016 3:14:39 PM			Edit	Delete	Status	ID

After clicking the **✓ Update**, the Collector Engine Table now displays the new created Collector engine for "AMT CE 001". Use the **Edit** button to make any changes to the name field. Use the **Delete** button to Remove the Collector Engine from the table. **Clicking** the **Status** button will display the current status of the Collector Engine. The **ID** button will open a small window with the ID displayed. Click **OK** to close the window.

Select the ID button to display the id number of the Collector Engine.

The new Collector Engine "AMT CE 001" is now available on the Table.

To make any changes to the Name, **click** the Edit button, then click **Update** to save.

```
{ '_id': 'f587ccc5-0abf-421b-bc6a-51cf364e8751' }
```

OK

Name	Agent Version	Agent Address	Agent Last Connect	Edit	Delete	Status	ID
AGENT TEST	2.2.15316.1170	172.24.1.85	11/12/2015 3:20:20 PM				
DR_TEST			04/15/2016 13:30:25				
DTP-EDC2							
Development EDC3	2.2.15303.1115	172.24.1.31	5/5/2016 2:27:33 PM				
AMT CE 001							

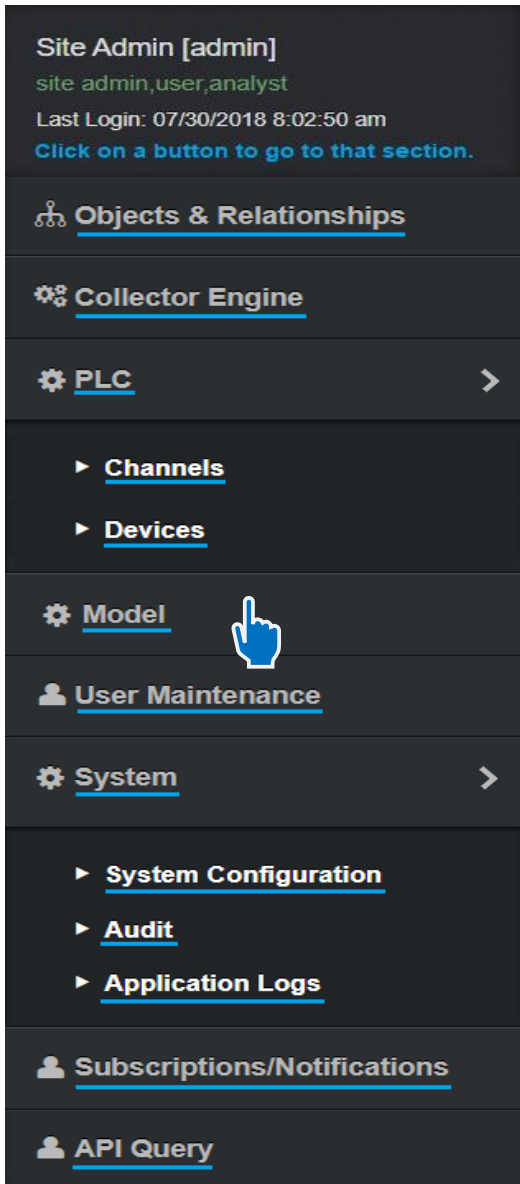
# PLC

[PLC](#) | [Overview](#) | [PLC Function](#) | [PLC Devices](#)

## Overview

The next step after creating the Collector Engine is to create and configure the PLC device itself. Based on the type of PLC (**Rockwell ControlLogix, Siemens, Mitsubishi**, etc), you are dealing with for your enVision project, you will need to enter certain information to tell the enVision application where inside the PLC memory it needs to read data from.

When you slide the cursor to the left-hand edge of the window, the Admin menu appears. Under the heading, **PLC** are two (2) selections that can be clicked on, **Channels** and **Devices**.



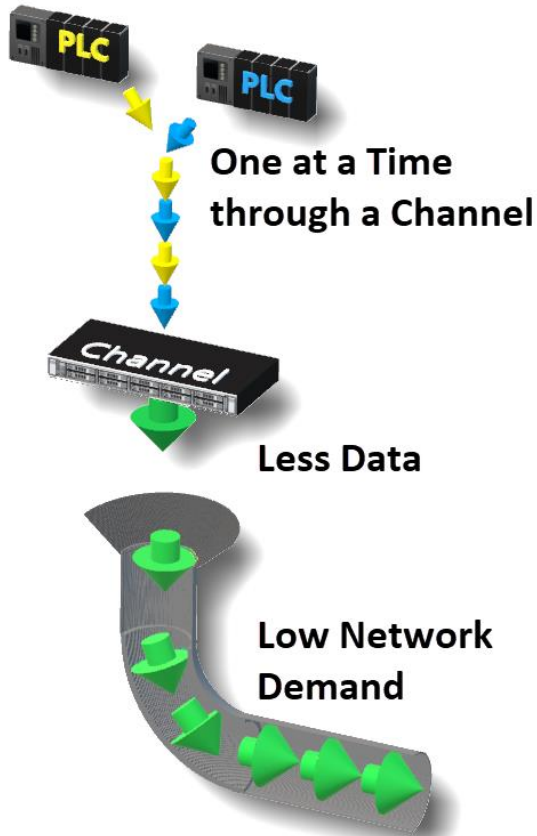
## PLC Function

- [One Channel setup with Multiple PLC's \(2 or More\)](#)
- [Two Channels Setup with 1 PLC Each](#)

The Channel information is used by the EDC server to setup how data is transferred between the different PLC devices and the EDC. Each channel is like a separate connection as far as the EDC is concerned. This means it can be communicating simultaneously with each PLC. This can be considered the recommended default setting. The trade off with having independent channels for each processor is that it adds more traffic to the network at any given point in time. When there are many PLC processors connected to an EDC it may become necessary to have PLC's share channels so as to reduce the amount of traffic on the network. When PLC's share a channel, the EDC will read data from them in a sequential fashion rather than simultaneously. The **Channels** selection is used to set up the sharing of channels between PLC's. The trade off with PLC's sharing a

channel is it takes longer for the EDC to read data from the PLC's since it is reading data from one PLC, then the next PLC, and so on, until it reads the last PLC on the channel and then repeats the sequence. Potential is that data may change in the PLC before the EDC was able to read it. There are options available to help reduce the risk of missing data. The **Channels** selection is usually only important when an EDC is going to be talking with many PLC's, say more than 4 or 5. The amount of data being read from a particular PLC could also play a role. These situations need to be addressed on a case by case basis.

### One Channel setup with Multiple PLC's (2 or More)



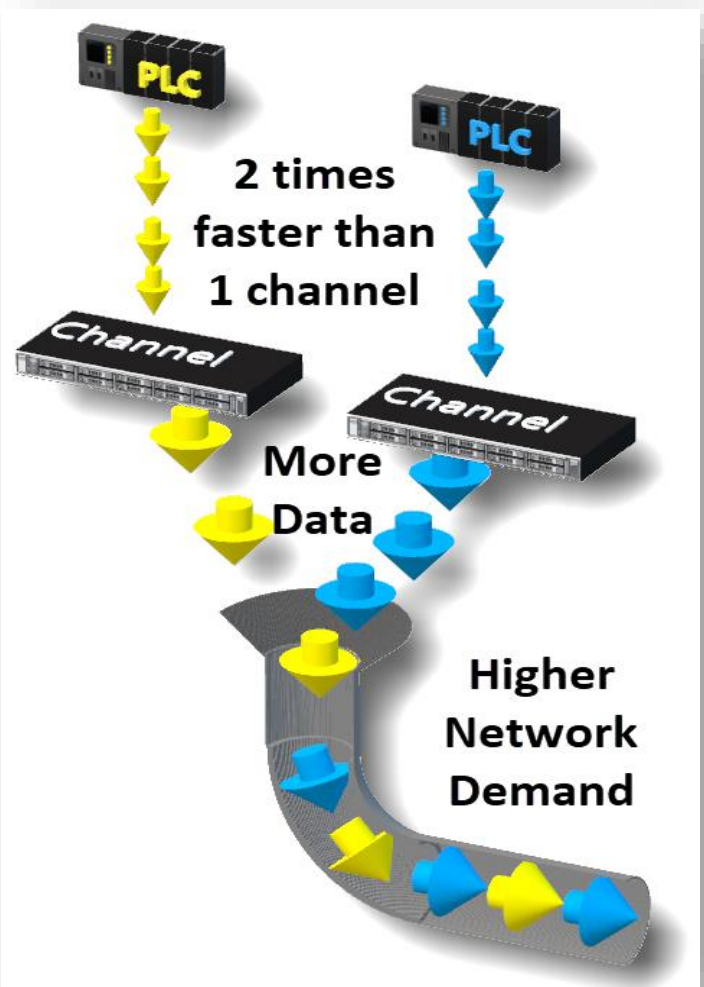
In the event that a cycle time would take several seconds to minutes, one channel can be used for multiple PLC's. Each PLC will send off a data package into the channel. The Channel will read each PLC one at a time in order. When the last PLC is read, it goes back to the 1<sup>st</sup> PLC.

Because the typical channel can only perform one action at a time. If there were 2 or more PLC's, then it would create a **bottleneck** at the channel input. It would only be able to process one action at a time. 1 action from PLC 1, then 1 (2) action from PLC 2 and so on. This setup is typically **slower** through the channel, but also creates **less demand or traffic**, on the network. If the cycle time is high and data requirements are low, then this would be a good setup to use. This system configuration is slower but has less demand on the network if used.

### Two Channels Setup with 1 PLC Each

In the event of a very low cycle time, where the cycle time would be millisecond to a few seconds, it would be better to use multiple channels with a single PLC. With 2 channels with a PLC for each, it would be able to process the data **faster**. The bottleneck would be eliminated. However, by using this setup, it would create **greater demand or traffic**, on the network.

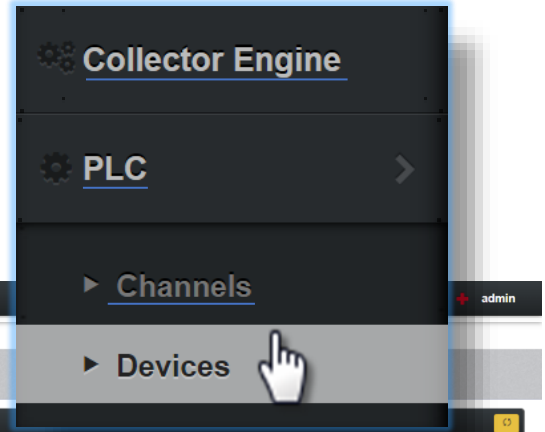
This would also be the case if you have different models of PLC's. Two different PLC's cannot run on the same channel. Therefore, it would be necessary to have a separate channel for each PLC. You have to take caution that too much demand on an Network connection can cause a series of issues such as flooding the network, saturation, and cause a backlogging resulting in packet losses as well as other problems.



# PLC Devices ▼

Click on **Devices** under PLC heading in side menu.

The following window appears. The data currently shown in the screenshot is data that was created in a test environment. It gives an idea of what a system with many PLC devices would look like. In the Admin menu, **click** the **+ Add new record** button.



Slide the cursor to the Left-hand side of the window to reveal the Admin Side Menu. Under the heading PLC are two selection that can be clicked on. Channels and Devices.

By selecting Devices or Channels, a window will open and display a table of your selection.

Name	Plc Device Type	Channel	Collector Engine	NetworkID	PLC ID	Job ID	Actions
A_A_Raw1	Rockwell ControlLogix	(AUTO) A_A_Raw1	DEPLOYMENT TEST	172.24.1.241	0	1	Edit Delete Config ID
AAA Basketball	Omron NJ	ADMIN TEST ONLY - DO NOT DEPLOY	EDC No Deployment	8.8.8.8.8.8	888	888	Edit Delete Config ID
AAM - TRMF D84 Front Axle Assy - CPU_OP010	Rockwell ControlLogix	AAM - TRMF D84 Front Axle Assy - CPU_OP010	AAM TRMF	10.5.140.27	1	1	Edit Delete Config ID
AAM - TRMF D84 Front Axle Assy - CPU_OP020A_020B	Rockwell ControlLogix	AAM - TRMF D84 Front Axle Assy - CPU_OP020A_020B	AAM TRMF	10.5.140.49	1	1	Edit Delete Config ID
AAM - TRMF D84 Front Axle Assy - CPU_OP030	Rockwell ControlLogix	AAM - TRMF D84 Front Axle Assy - CPU_OP030	AAM TRMF	10.5.140.71	0	1	Edit Delete Config ID
AAM - TRMF D84 Front Axle Assy - CPU_OP040_050	Rockwell ControlLogix	AAM - TRMF D84 Front Axle Assy - CPU_OP040_050	AAM TRMF	10.5.140.93	1	1	Edit Delete Config ID
AAM - TRMF D84 Front Axle Assy - CPU_OP060	Rockwell ControlLogix	AAM - TRMF D84 Front Axle Assy - CPU_OP060	AAM TRMF	10.5.140.115	1	1	Edit Delete Config ID
AAM - TRMF D84 Front Axle Assy - CPU_OP070A-OP070B	Rockwell ControlLogix	AAM - TRMF D84 Front Axle Assy - CPU_OP070A-OP070B	AAM TRMF	10.5.140.137	1	1	Edit Delete Config ID
AAM - TRMF D84 Front Axle Assy - CPU_OP110	Rockwell ControlLogix	AAM - TRMF D84 Front Axle Assy - CPU_OP110	AAM TRMF	10.5.140.203	1	1	Edit Delete Config ID
AAM - TRMF D84 Front Axle Assy - CPU_OP140_150	Rockwell ControlLogix	AAM - TRMF D84 Front Axle Assy - CPU_OP140_150	AAM TRMF	10.5.141.2	1	1	Edit Delete Config ID
AAM - TRMF D84 Front Axle Assy - CPU_OP170_180	Rockwell ControlLogix	AAM - TRMF D84 Front Axle Assy - CPU_OP170_180	AAM TRMF	10.5.141.52	1	1	Edit Delete Config ID
AAM - TRMF G&X Tube Welding - CPU_OP010LH	Rockwell ControlLogix	AAM - TRMF G&X Tube Welding - CPU_OP010LH	AAM TRMF G&X	172.21.72.24	0	1	Edit Delete Config ID

Click the **+ Add new record** button and a series of options will appear, text boxes to name the **Device** and **Network ID**. Drop down menus for **Plc Device Type**, **Channel**, and **Collector Engine**. And a selector for **PLC ID** and **Job ID** as well. Enter the information for the PLC device you are adding, then click the **Update** button.

Click on the **+ Add new record** button and a series of options will appear. Text boxes to name the Device and Network ID. Drop down menus for Plc Device Type, Channel, and Collector Engine. And a selector for PLC ID and Job ID.

After you have completed inputting your selection, click **Update** button to update the selection.

Name(*)	Plc Device Type(*)	Channel	Collector Engine	NetworkID(*)	PLC ID	Job ID	Actions
TEST 22	Mitsubishi Q	(AUTO) AGENT PLC TEST	AGENT TEST		0	0	Update Cancel
Test PLC A 48	Mitsubishi Q	Test Channel K	AGENT TEST		0	0	Edit Delete Config ID
Test PLC J	Modicon TSX Premium	Test_2_05/08/2015 17:29:03	DTP EDC2	1.2.3.4	0	0	Edit Delete Config ID
Test PLC C 422	Omron CS1	05/14/2015 16:44:05	Development EDC3	1.2.3.4	0	0	Edit Delete Config ID
Test PLC A	Rockwell	Test_2_05/14/2015 16:44:05	EDC - DD Assembly Area	172.24.1.241	0	2	Edit Delete Config ID
Test	ControlLogix	Training SIM	EDC - JE Test	1111	0	0	Edit Delete Config ID
Test Line 2 Copy PLC	Rockwell	Training SIM - 07/09/2015	EDC - JE Test Mitsu	1111	0	0	Edit Delete Config ID
Test Line 2 PLC	Siemens Nellink	11.15.28	JDC - P552 I I BODYSIDE	1.1.1.1	0	0	Edit Delete Config ID
Test Line 2 PLC	Siemens Nellink	Training SIM - 07/23/2015	SEG 2	10.109.204.83	1	1	Edit Delete Config ID

**Admin**

PLC Device (\*) Required Fields

+ Add new record

Drag a column header and drop it here to group by that column

Name(*)	Plc Device Type(*)	Channel	Collector Engine	NetworkID(*)	PLC ID	Job ID	Update	Cancel
TEST	Rockwell ControlLo		EDC No Deployment	136.129.6.4	0	0	Update	Cancel

**Callouts:**

- Name(\*) field:** The Name (\*) field is where you enter a name for the PLC.
- PLC Device Type(\*) field:** The PLC Device Type (\*) field. Choose a PLC that you are connecting to.
- Channel field:** The Channel field can be left blank. A channel with the same name will be auto created and inserted when the update button is clicked.
- Collector Engine field:** The Collector Engine field is where you enter the Collector Engine name that was created previously. Test is the Collector Engine name that was created for this example.
- NetworkID(\*) field:** The NetworkID(\*) field is where the Ethernet I.P. address for the PLC device is entered. For Rockwell this typically would be the I.P. address of one of the Ethernet modules in the chassis where the PLC resides. This would be based on how the EDC is networked into the PLC network.
- Update button:** Click the Update button when finished to update the PLC device.
- PLC ID and Job ID:** PLC ID and Job ID fields are currently not used and are ignored.

The **Name(\*)** field is where you enter a name for the PLC. For this example, **Test** is the processor name.

The **Plc Device Type(\*)** field utilizes a pulldown menu from which you select the type of PLC that you are connecting to. For this example, **Rockwell ControlLogix** was selected from the menu.

The **Channel** field can be left blank. A Channel with the same name as the Collector Engine will automatically be created and inserted into the field when the Update button is selected.

The **Collector Engine** field is where you enter the Collector Engine name that was created previously. **Test** is the Collector Engine name that was created for this example. The Collector Engine named "Test" communicates with the named PLC Device "Test".

The **Network ID (\*)** field is where the Ethernet I.P. address for the PLC device is entered. For Rockwell this typically would be the I.P. address of one of the Ethernet modules in the chassis where the PLC resides. This would be based on how the EDC is networked into the PLC network. For this example, **136.129.6.4** was used.

The **PLC ID** is an Integer reference number for the PLC device.

The **Job ID** is the number of the job agent that will process the collected data. For advanced users only, a performance increase may be possible by distributing the processing load among multiple job agents.

**PLC Device (\*) Required Fields**

+ Add new record

Drag a column header and drop it here to group by that column

Name(*)	Plc Device Type(*)	Channel	Collector Engine	NetworkID(*)	PLC ID	Job ID	Edit	Delete	Config	ID
PCB Test PLC 2	Omron CS1	Omron Simulation	EDC No Deployment	172.24.1.244	21	6	Edit	Delete	Config	ID
PLC2	Rockwell MicroLogix 1100	Demo Channel	EDC No Deployment	1.2.3.4	0	0	Edit	Delete	Config	ID
Rockwell 2.1 Test	Rockwell ControlLogix	Rockwell 2.1 Test	EDC No Deployment	172.24.1.241	1	1	Edit	Delete	Config	ID
Test	Rockwell ControlLogix	(AUTO) Test	Test	136.129.6.4	0	0	Edit	Delete	Config	ID
Test Line 2 Copy PLC	Rockwell ControlLogix	Test_2 - 05/08/2015 17:29:03 - 05/11/2015	EDCS	172.24.1.241	1	4	Edit	Delete	Config	ID
Test	Rockwell ControlLogic	(AUTO) Test	Test	136.129.6.4	0	0				
Test PLC A	Rockwell ControlLogix	Test Channel B	EDC No Deployment	172.24.1.241	1	4	Edit	Delete	Config	ID

**Callout:** This Collector engine communicates with this PLC.

Once all of the parameters are set and you approve, **click the Update** to save.

# PLC Channel ▾

- [Array.ANALOG.ArrayCount](#)
- [Array.IO.ArrayCount](#)
- [Array.OP.ArrayCount](#)
- [Array.USER.ArrayCount](#)
- [PATH](#)
- [Program.Name](#)
- [SLOT](#)

## Overview

The PLC Device, named **Test**, is created and added to the list of available PLC Devices. Note that the **Channel** field has an entry of **(AUTO) Test** shown. This indicates that a Channel named **Test** was automatically created when you created the PLC Device since the **Channel** field was left blank.

**Click** on the **Channels** button on the LH side menu and see that the channel **(AUTO) Test** was created.

**Admin**

**PLC Channels**

Name	Channel Type	Edit	Delete	ID
(AUTO) AGENT PLC TEST	ControlLogix Ethernet			ID
(AUTO) Dan Test	ControlLogix Ethernet			ID
(AUTO) FORD Template PLC	ControlLogix Ethernet			ID
(AUTO) IRN USER PLC	Siemens TCB/ID Ethernet			ID
(AUTO) IRN USER PLC	ControlLogix Ethernet			ID
(AUTO) KNIVAN	ControlLogix Ethernet			ID
(AUTO) Test	ControlLogix Ethernet			ID
(AUTO) Test - Omron PLC	Omron Ethernet			ID
(AUTO) Test	ControlLogix Ethernet			ID
TEST Channel	ControlLogix Ethernet			ID
PLC Channel	ControlLogix Ethernet			ID
ADMIN TEST ONLY - DO NOT DEPLOY	ControlLogix Ethernet			ID
ATATATATATATATATAT	ControlLogix Ethernet			ID
An ADD/Delete Test	ControlLogix Ethernet			ID
Analog Channel	ControlLogix Ethernet			ID
BC 4545	ControlLogix Ethernet			ID
BC RTRTR	Mitsubishi Ethernet			ID

**Admin**

**PLC Device (\*) Required Fields**

Drag a column header and drop it here to group by that column

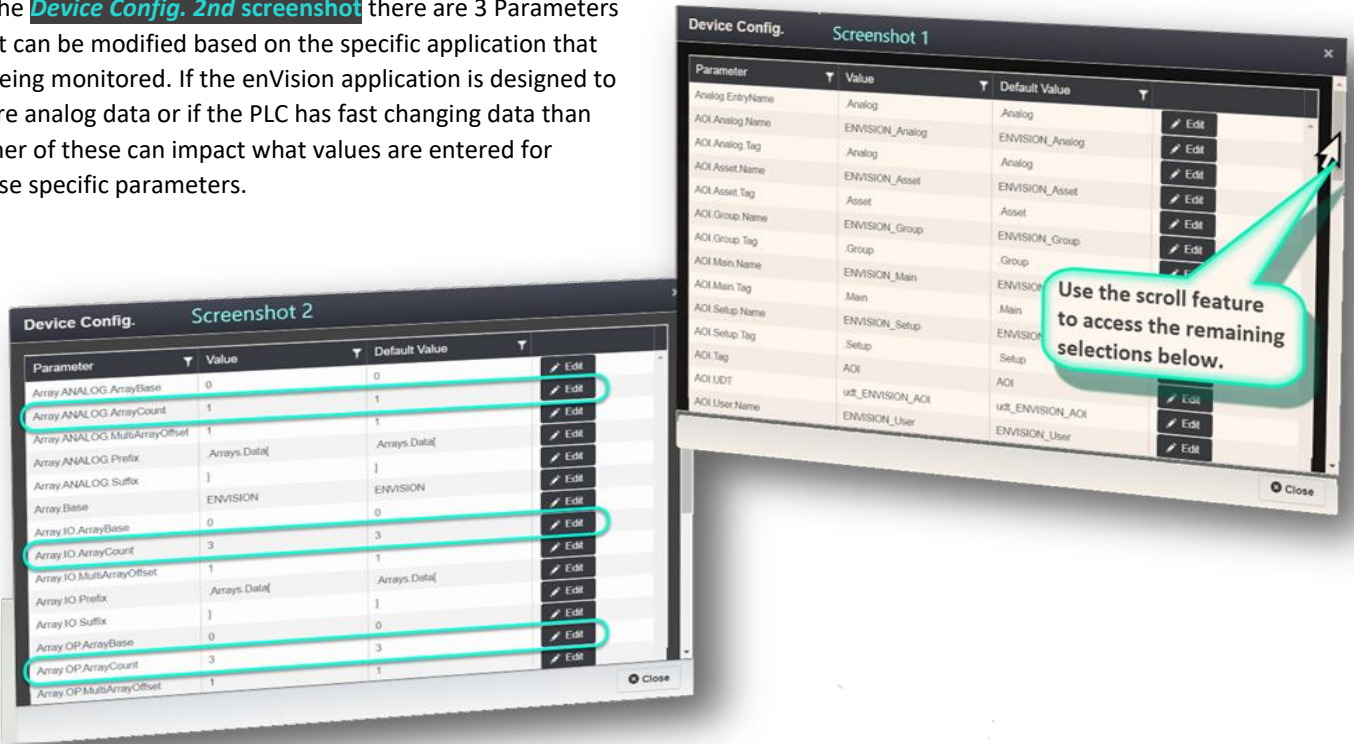
Name(*)	Plc Device Type(*)	Channel	Collector Engine	NetworkID(*)	PLC ID	Job ID	Edit	Delete	Config	ID
Mitsubishi_Test	Mitsubishi Q	Mitsubishi_Test	EDC No Deployment	172.24.1.241	0	1				ID
Rockwell v2.1 Test	Rockwell ControlLogix	Rockwell 2.1 Test	EDC No Deployment	172.24.1.241	1	1				ID
<b>Test</b>	<b>Rockwell ControlLogix</b>	<b>(AUTO) Test</b>	<b>EDC No Deployment</b>	<b>136.129.6.4</b>	<b>0</b>	<b>1</b>				ID
Test Line 2 Copy PLC	Rockwell ControlLogix	Test_2 - 05/08/2015 17:29:03 - 05/14/2015 16:44:05	Development EDC3	172.24.1.241	1	4				ID
Test Line 2 PLC	Rockwell ControlLogix	Test_2 - 05/14/2015 16:44:05	EDC No Deployment	172.24.1.241	1	4				ID
Test PLC A	Rockwell ControlLogix	Test Channel B	EDC No Deployment	172.24.1.241	1	4				ID
Test PLC C 4222	Rockwell ControlLogix	Test Channel K	EDC No Deployment	172.24.1.241	1	4				ID
Test PLC J	Rockwell ControlLogix	Test Channel J	EDC No Deployment	172.24.1.241	1	4				ID
TRAINING_SIM_FIXT	Rockwell ControlLogix	Training SIM	EDC No Deployment	172.24.1.241	1	4				ID



The next step is to configure the PLC Device. Go back to the Admin Display Menu, and select Devices under the PLC Tab. The configuration selections are based on the PLC type (Rockwell, Siemens, etc.). For this example, we'll be dealing with configuring a Rockwell ControlLogix PLC Device. Click on the **Config** button for the PLC Device named (**Test**) that you would like to configure. The following window pops up. Not all of the selections can be seen at one time in the window. There is a scroll bar to allow navigation to the remaining selections. Multiple screenshots will be utilized to show all the different selections that are available. There are actually only 6 or so selections that should ever be modified and so those will be focused on. The remainder should use the Default Value unless there is a sound reason to change them.

In the **Device Config. 1st screenshot** all the Parameters shown should be left with their Default Value. They are shown in the Device Config. list for informational purposes.

In the **Device Config. 2nd screenshot** there are 3 Parameters that can be modified based on the specific application that is being monitored. If the enVision application is designed to store analog data or if the PLC has fast changing data than either of these can impact what values are entered for these specific parameters.



## Array.ANALOG.ArrayCount Array.USER.ArrayCount | 1 | 1 | Edit

This value (*Array.ANALOG.ArrayCount*) can range from 1 to 5. If the specific enVision application is not designed to capture any analog data than this Value should be left at its Default Value of 1. If the enVision application is designed to capture analog data than it is recommended that the Value be changed to 5. The PLC driver will use 5 identical arrays that will store **Change Of State (COS)** data for all analog signals in a cyclical manner. 1<sup>st</sup> COS will be stored in array 1, 2<sup>nd</sup> COS in array 2, 3<sup>rd</sup> COS in array 3, 4<sup>th</sup> COS in array 4, 5<sup>th</sup> COS in array 5, 6<sup>th</sup> COS in array 1, 7<sup>th</sup> COS in array 2, etc. This allows the EDC to capture fast changing data even if the OPC (Kepware) is not as fast as the PLC scan time. The OPC will read all 5 arrays every time it accesses the PLC database.

## Array.IO.ArrayCount Array.IO.ArrayCount | 3 | 3 | Edit

This value (*Array.IO.ArrayCount*) can range from 1 to 5. The Default Value of 3 is recommended to be left unchanged. Data in the IO array can sometimes have COS in consecutive scans of the PLC. This usually occurs near the end of a cycle and the beginning of the next one. Since the COS of data in the IO array is "fast" only at certain points and only for 2 or 3 PLC scans having the number of arrays used set to 3 is usually sufficient. The PLC driver will use 3 identical arrays that will store **Change Of State (COS)** data in a cyclical manner. 1<sup>st</sup> COS will be stored in array 1, 2<sup>nd</sup> COS in array 2, 3<sup>rd</sup> COS in array 3, 4<sup>th</sup> COS in array 1, 5<sup>th</sup> COS in array 2, etc. This allows the EDC to capture data that may have changed in back to back scans of the PLC even if the OPC (Kepware) is not as fast as the PLC scan time. The OPC will read all 3 arrays every time it accesses the PLC database.

## Array.OP.ArrayCount Array.OP.ArrayCount | 3 | 3 | Edit

Array.OP.ArrayCount – This Value can range from 1 to 5. The Default Value of 3 is recommended to be left unchanged. The same explanation used above for Array.IO.ArrayCount holds true for Array.OP.ArrayCount.

In the **Device Config. 3** screenshot there is 1 Parameter that can be modified based on the specific application that is being monitored. If the enVision application is designed to store User data and the data is fast changing than that will impact what value is entered for this specific parameter.

Device Config.		Screenshot 3		
Array.IO.Prefix	.Arrays.Data[	.Arrays.Data[		Edit
Array.IO.Suffix	]	]		Edit
Array.OP.ArrayBase	0	0		Edit
Array.OP.ArrayCount	3	3		Edit
Array.OP.MultiArrayOffset	1	1		Edit
Array.OP.Prefix	.Arrays.Data[	.Arrays.Data[		Edit
Array.OP.Suffix	]	]		Edit
Array.USER.ArrayBase	0	0		Edit
Array.USER.ArrayCount	1	1		Edit
Array.USER.MultiArrayOffset	1	1		Edit
Array.USER.Prefix	.Arrays.Data[	.Arrays.Data[		Edit
Array.USER.Suffix	]	]		Edit
BlockSize	3840	3840		Edit
File.OutputMode	1	1		Edit
Path		\\10		Edit
Port	44818	44818		Edit

## Array.USER.ArrayCount Array.USER.ArrayCount | 1 | 1 | Edit

Array.USER.ArrayCount – This Value can range from 1 to 5. If the specific enVision application is not designed to capture any User data, then this Value should be left at its Default Value of 1. If the enVision application is designed to capture User data, then this value can be modified if the data is fast changing using the same explanation given for Array.ANALOG.ArrayCount.

**(Note: USER Array is currently unavailable. Future releases of enVision will incorporate this feature).**

In [Device Config.4 screenshots](#) there are 3 Parameters that can be modified based on the specific configuration of the PLC hardware and how the connection is made between the EDC and the PLC.

Array.USER.ArrayCount	1	1	Edit
Array.USER.MultiArrayOffset	1	1	Edit
Array.USER.Prefix	.UD[	.UD[	Edit
ConnectionTimeoutSeconds	3	3	Edit
FailAfter	3	3	Edit
Main Routine	z_Envision	z_Envision	Edit
PATH	1	1	Edit
PORT	44818	44818	Edit
Program.Name	ENVISION	ENVISION	Edit
ResponseTimeoutMilliseconds	1000	1000	Edit
SLOT	0	0	Edit
Tag.Scope	CONTROLLER	CONTROLLER	Edit

**PATH** PATH | 1 | 1 | Edit

This Value helps describe how the EDC (specifically the Kepware OPC) will connect to the PLC database after it has established communications to the Network ID entered for the PLC Device (136.129.6.4 for this example). For Rockwell a Value of 1 indicates that the PLC processor is accessed via the chassis backplane where the processor is installed. This is the most common method of connecting to a Rockwell PLC. This means the EDC is connected directly to an Ethernet communication module that is mounted in the same chassis as the PLC. There are other scenarios where the network between the EDC and the PLC is more complex and subsequently causes the PATH parameter to be more complex. You will need to refer to the Kepware manual that specifically addresses the different communication network setups that it supports for Rockwell processors to determine what the PATH value should be for these situations.

<https://www.kepware.com/products/kepserverex/drivers/allen-bradley-controllogix/documents/allen-bradley-controllogix-ethernet-manual.pdf>

**Program.Name** Program.Name | enVision | enVision | Edit

This parameter (Program.Name) defines the program name that the enVision PLC software driver will be called when it is imported into the ControlLogix project file. The Import file (\*.L5X) is created when a Deploy is performed. It is recommended that the Default Value of enVision is used unless it is absolutely necessary to change it. One situation where the Program.Name parameter would need to change is if there is more than one (1) enVision PLC driver program imported into a PLC. The enVision PLC program names will need to be unique in this scenario. One Program.Name could be enVision and the other enVision\_A for example. The Program.Name is used by EDC to find the program scoped tags that it needs to access.

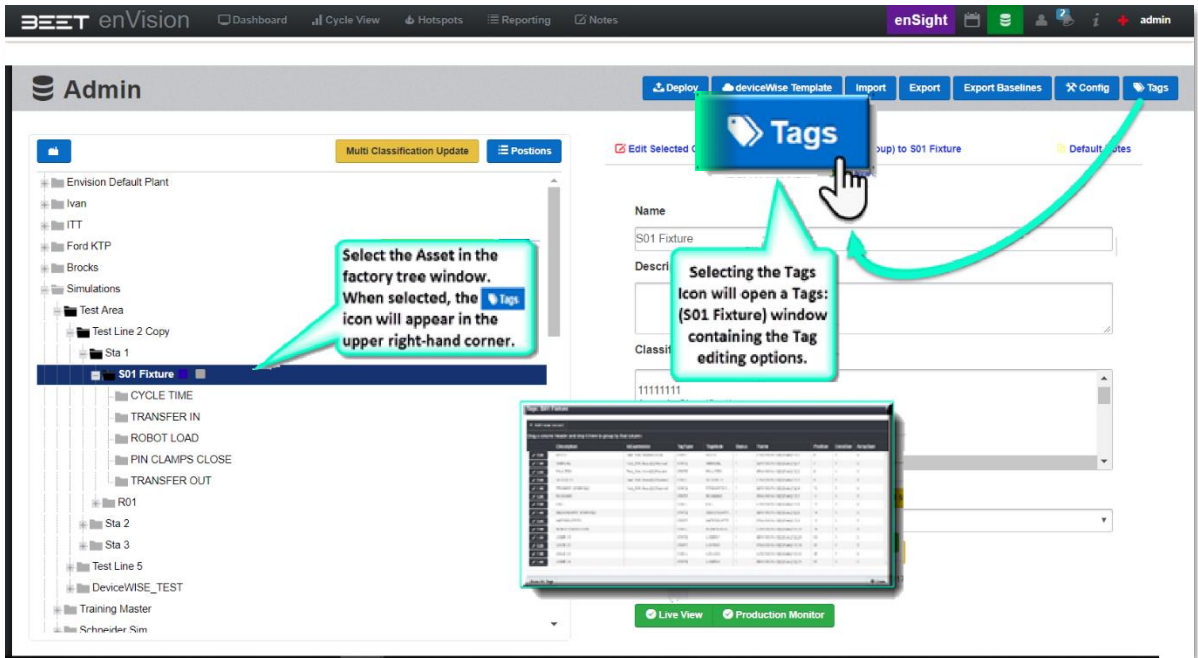
**SLOT** SLOT | 0 | 0 | Edit

This parameter (SLOT) is used to define the slot within the PLC chassis that the PLC processor is installed. This value is used by the EDC in helping define the communication path between the EDC and PLC. Typically PLC processors are mounted in slot 0 although they can be mounted in any slot. Modify this parameter as needed.

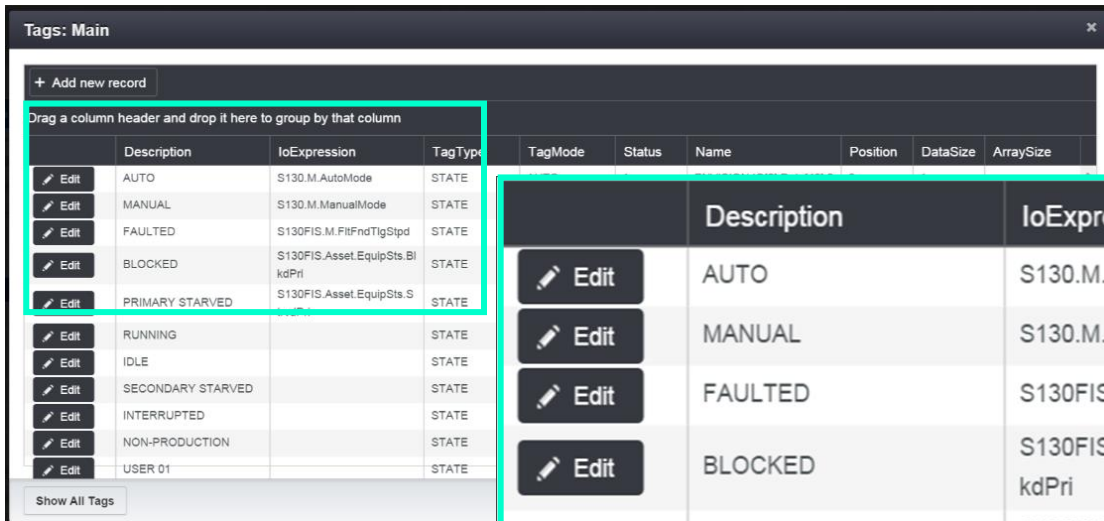
# Editing Asset Tags ▼

[Editing Asset Tags](#) | [PLC Tags](#) | [PLC Setup and Configurations](#)

Asset Tags are used to convey additional information to the enVision application for display purposes and for reporting calculations. To access the tags in Admin mode, highlight an object at the Asset level in the Factory tree window and then click on the Tags button.



The following window appears:



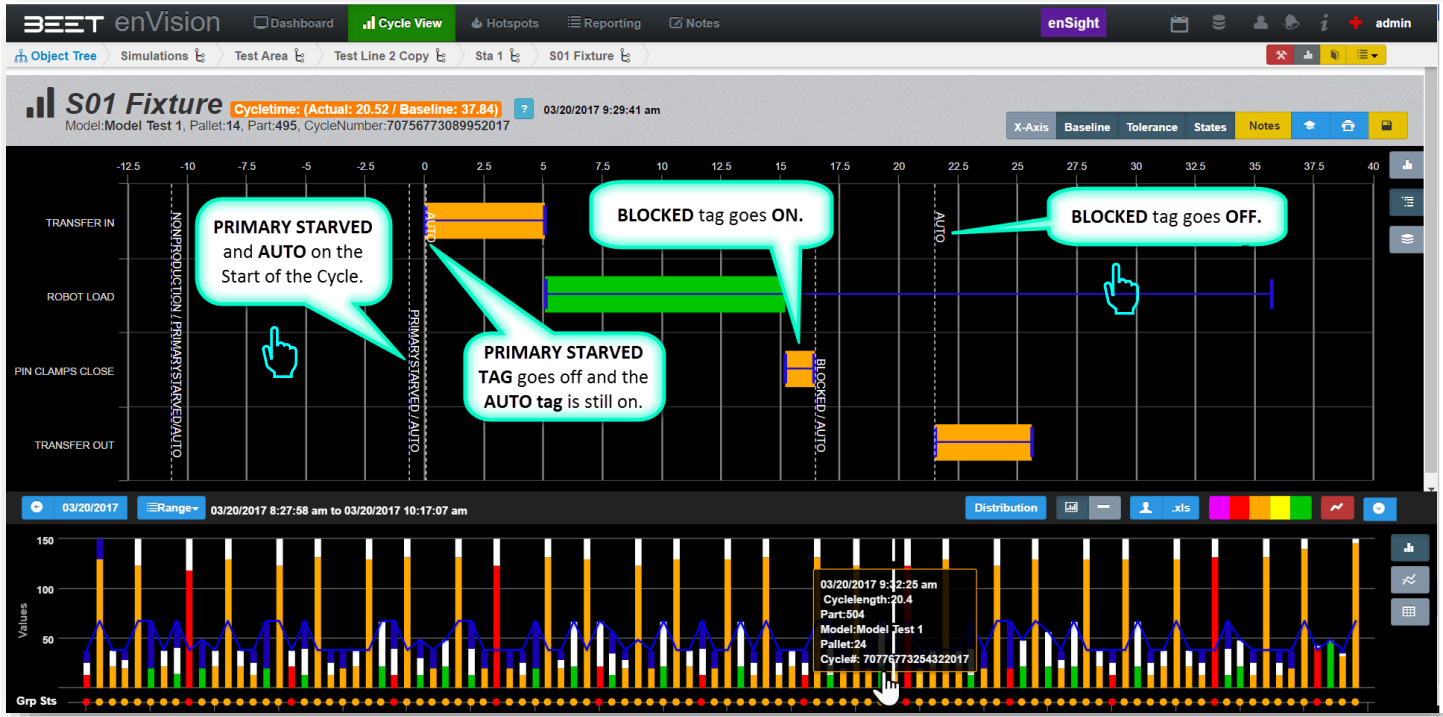
Here is a zoomed in of the most frequent tags used for cycle recording.

	Description	IoExpression	TagType
	AUTO	S130.M.AutoMode	STATE
	MANUAL	S130.M.ManualMode	STATE
	FAULTED	S130FIS.M.FltFndTlgStpd	STATE
	BLOCKED	S130FIS.Asset.EquipSts.BlkdPri	STATE
	PRIMARY STARVED	S130FIS.Asset.EquipSts.StarvdPri	STATE
	RUNNING		STATE
	IDLE		STATE

# PLC Tags

PLC tags have already been assigned in this example for the I/O Expressions. The Asset Tags themselves are pretty self-explanatory. AUTO, MANUAL, FAULTED, BLOCKED, PRIMARY STARVED (and also SECONDARY STARVED which isn't used in this example) are used to draw lines when looking at cycle history with Sequence View enabled to indicate when any of these signals changes state.

Below is an example of how Asset Tags are displayed in **Cycle view** screen. Note: Must be selected for sequence view.



There are other Asset Tags that can be edited. To see these, click the **Show All Tags** button located at the lower LH corner of the window.

Tags: S01 Fixture

+ Add new record

Drag a column header and drop it here to group by that column

	Description	IoExpression	TagType	TagMode	Status	Name	Position	DataSize	ArraySize
Edit	SPARE 13		STATE	STATE_13	1	ENVISION.IO[0].Data[12].13	19	1	0
Edit	SPARE 14					ENVISION.IO[0].Data[12].14	20	1	0
Edit	SPARE 15					ENVISION.IO[0].Data[12].15	21	1	0
Edit	SPARE 16					ENVISION.IO[0].Data[12].16	22	1	0
Edit	SPARE 17		STATE	STATE_17	1	ENVISION.IO[0].Data[12].17	23	1	0
Edit	SPARE 18		STATE	STATE_18	1	ENVISION.IO[0].Data[12].18	24	1	0
Edit	SPARE 19		STATE	STATE_19		ENVISION.IO[0].Data[12].19	25	1	0
Edit	SPARE 20		STATE	STATE_20			26	1	0
Edit	SPARE 21		STATE	STATE_21			27	1	0
Edit	SPARE 22		STATE	STATE_22			28	1	0
Edit	SPARE 23		STATE	STATE_23			29	1	0
Edit	SPARE 24		STATE	STATE_24			30	1	0
Edit	SPARE 25		STATE	STATE_25	1	ENVISION.IO[0].Data[12].25	31	1	0
Edit	SPARE 26		STATE	STATE_26	1	ENVISION.IO[0].Data[12].26	32	1	0
Edit	SPARE 27		STATE	STATE_27	1	ENVISION.IO[0].Data[12].27	33	1	0

Buttons: Hide All Tags, Show All Tags, Close

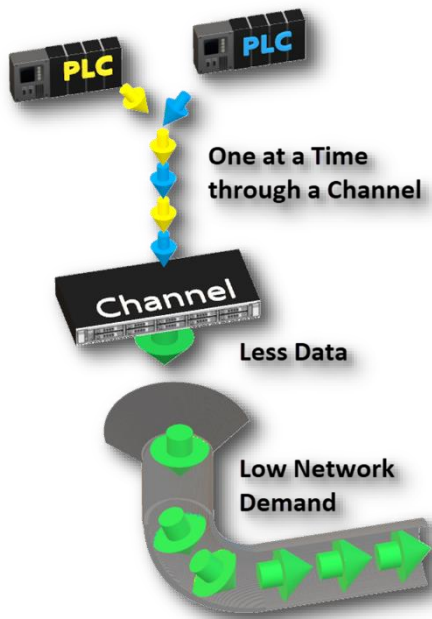
Callouts:
 

- Show All Tags
- Click the Hide All Tags, to re-open the Tags: Default (S01 Fixture) window.
- Click the Show All Tags, to re-open the Tags: S01 Fixture (default) window, to display all the available tags.

## PLC Setup and Configuration

[Editing Asset Tags](#) | [PLC Tags](#) | [PLC Setup and Configurations](#)

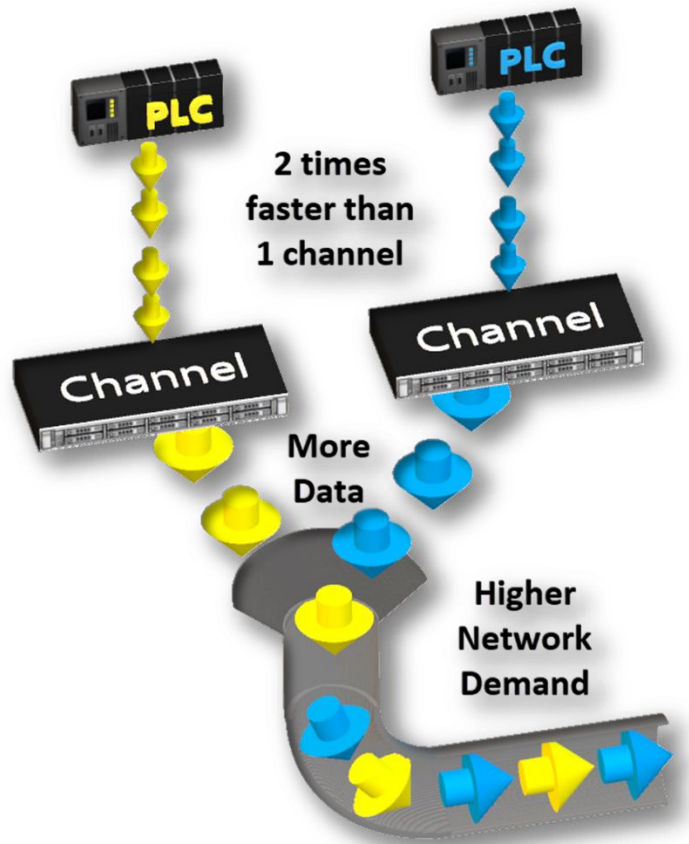
### One Channel setup with Multiple PLC's.



In the event, that a cycle time would take several seconds to minutes, one channel can be used for multiple PLC's. However, the typical channel can only perform one action at a time. If there were 2 or more PLC's, then it would create a bottleneck at the channel input. It would only be able to process one action at a time. 1 action from PLC 1, then 1 (2) action from PLC 2 and so on. This setup is typically slower but has less demand on the network. If the cycle time is high and data requirements are low, then this would be a good setup to use. This system configuration is slower but has less demand on the network if used.

### Two Channels Setup with 1 PLC

In the event of a very low cycle time, where the cycle time would be millisecond to multiple seconds, it would be better to use multiple channels with a single PLC. With 2 channels with on PLC a piece, it would be able to process the data faster. The bottleneck would be eliminated. However, by using this setup, the demand on the network would be greater. This would also be the case if you have different models of PLC's. Two different PLC's cannot run on the same channel. Therefore, it would be necessary to have a separate channel for each different PLC. You must take caution that too much demand on a Network connection can cause a series of issues such as flooding the network, saturation, and cause a backlogging resulting in packet losses and other problems.



# Models

Site Admin [admin]  
 site\_admin,user,analyst  
 Last Login: 07/30/2018 8:02:50 am  
 Click on a button to go to that section.

- Objects & Relationships
- Collector Engine
- PLC
  - Channels
  - Devices
- Model
- User Maintenance
- System
  - System Configuration
  - Audit
  - Application Logs
- Subscriptions/Notifications
- API Query

[Models](#) | [Overview](#) | [Adding, Editing, and Deleting](#)

## Overview

In enVision Models have significant importance. Models are used to signify a major motion path that needs independent baselines assigned to these motions. It must have a valid IoExpression entered for Asset cycles to be captured. The enVision application uses the MODEL tag as a pointer to the appropriate baseline cycle time values entered for each group of the Asset. Options are used to signify part type differences where the major motion remains the same and existing baselines and tolerances can be used. Options are generally used for reporting purposes.

At the top of the Cycle View window you will see the following tags now visible: CYCLENUMBER, PARTNUMBER, MODEL, OPTION and PALLET. These tags are 32-bit values. **The MODEL tag has significant importance. NOTE: It must have a valid IO Expression entered, in order for Asset cycles to be captured.** The enVision application uses the MODEL tag as a pointer to the appropriate baseline cycle time values entered for each group of the Asset. You define the models and their values in a different area of the Admin environment. This is shown in detail elsewhere but a short example of what the MODEL tag is used for and its significance is shown below.

Object Name: **Sta 30 In-Station**

Cycle Time / Baseline: **Actual: 39.76 / Baseline: 43.7**

Date and Time is actual: **10/26/2015 11:28:46 pm**

Model #: **Accepted Part Option: 2**

Pallet and Part #: **1518\*\*\*\*\***

CycleNumber: **1518\*\*\*\*\***

Model: Accepted Part OPTION: 2, Pallet: 324, Part: 29950655, CycleNumber: 151872878490992015

Slide the cursor to the left side of the screen to open the **Admin Display menu**. From the side menu, select the **Model (Tab)** from the side Admin Display Menu.

A list of the PLC devices that have been defined appears. You select the PLC that was entered for the PLC Device for the Asset. For this example, the PLC Device is **TRAINING\_SIM\_STA**. Click on it.

The following window appears. In this example, 3 models have been created. The **Part Code** values are the numbers that the enVision application expects to see returned by the tag name entered in the MODEL Asset Tag. **S130VRoll\_SR.Model** is the tag name that was entered for this example.

enVision Admin

Copy Checked to different PLC

Name	Part Code	Type	Priority	Action
2012 ND	3000000	GOOD	1	[Icons]
2012 PV	1000000	GOOD	1	[Icons]
Crew Cab	3	GOOD	3	[Icons]
Reg Cab	1	GOOD	3	[Icons]
Sup Cab	2	GOOD	3	[Icons]

**Use the pull down tab to choose the Type of Model. GOOD, DMGD, EMPTY, NULL or TEST.**

**To add a new Model, enter a Name, Part Code, Type and Priority. The click the + button to add to the list.**

**Select the PLC to view the details in the window to the right of the list.**

**Click on the Edit button to edit the Name, Part Code, Type and Priority fields. Make your changes and click the Save button when completed or X to cancel.**

**Select Delete button to delete the Model selected.**

Name	Part Code	Type	Priority	Action
2012 ND	30000000	GOOD	1	[Edit] [Delete] [Info]
2012 PW	10000000	GOOD	1	[Edit] [Delete] [Info]
Crew Cab	3	GOOD	3	[Edit] [Delete] [Info]
Reg Cab	1	GOOD	3	[Edit] [Delete] [Info]
Sup Cab	2	GOOD	3	[Edit] [Delete] [Info]

## Adding, Editing, and Deleting

From this **Model** screen, several types will be listed in a table. These are the types that were previously inputted earlier from when the test was setup. They can be edited or deleted. You may also add to the list by simply filling out the fields above the list. Enter the name that you would like to call it, the Part Code, Type, and Priority. Then click the + button to add it to the list.

### Name

Enter a name that is unique to the object you are using. For example, above, there are 3 different Cab configurations, Crew, Regular, and Super Cab. To add the **Extended Cab (Ext)**, we would enter **Ext** under the name dialog box.

### Part Code

A part code can be an actual part number or merely a designation number to show a difference between parts or similar parts. In the example above, the different types have a simple number to designate their Part Codes.

### Type

In this part of the Table, there is a pull-down tab to which will give you a selection of choices to choose from. They are GOOD, DMGD (Damaged), EMPTY, NULL, and TEST. These labels or strings are configurable to make it whatever you deem as needed. They are common types in manufacturing. The label can be tagged so the PLC can know what is in the system at that time.

In the case of manufacturing, the choice of **GOOD** would be picked in the production of parts. **DMGD** selection would represent a part or process that could not be completed due to an incomplete process or stoppage.

### GOOD

- DMGD
- EMPTY
- NULL
- TEST

**Click on the Objects & Relationships tab to go back to view the Factory Window**

**An integer value is used to select which model to use when multiple models are present.**

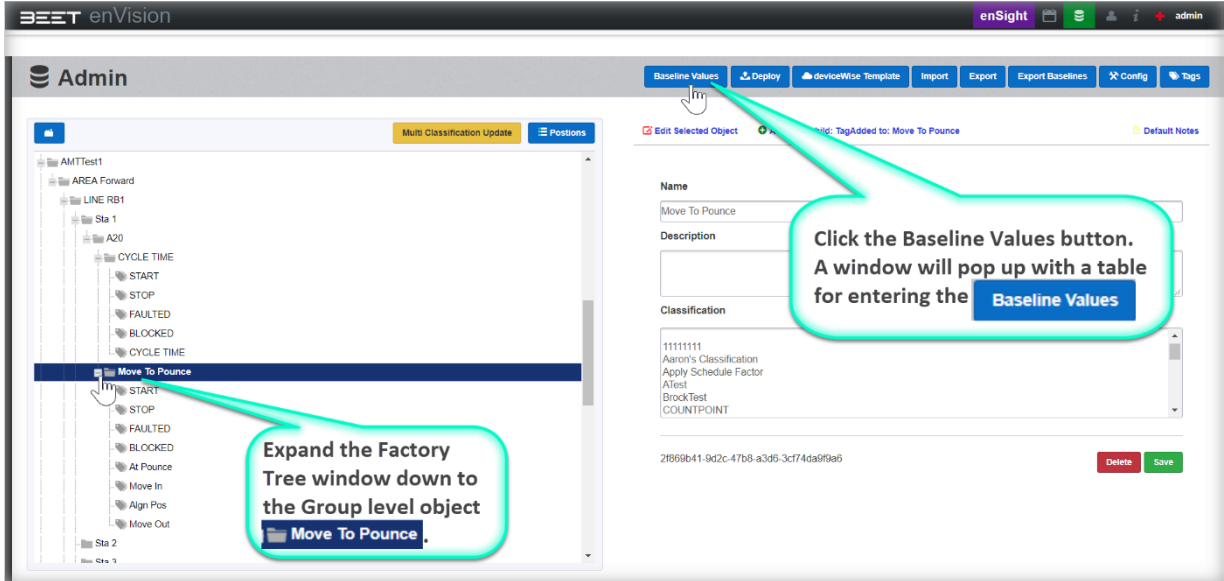
Name	Part Code	Type	Priority	Action
2012 ND	30000000	GOOD	1	[Edit] [Delete] [Info]
2012 PW	10000000	GOOD	1	[Edit] [Delete] [Info]
Crew Cab	3	GOOD	3	[Edit] [Delete] [Info]
Reg Cab	1	GOOD	3	[Edit] [Delete] [Info]
Sup Cab	2	GOOD	3	[Edit] [Delete] [Info]



# Priority

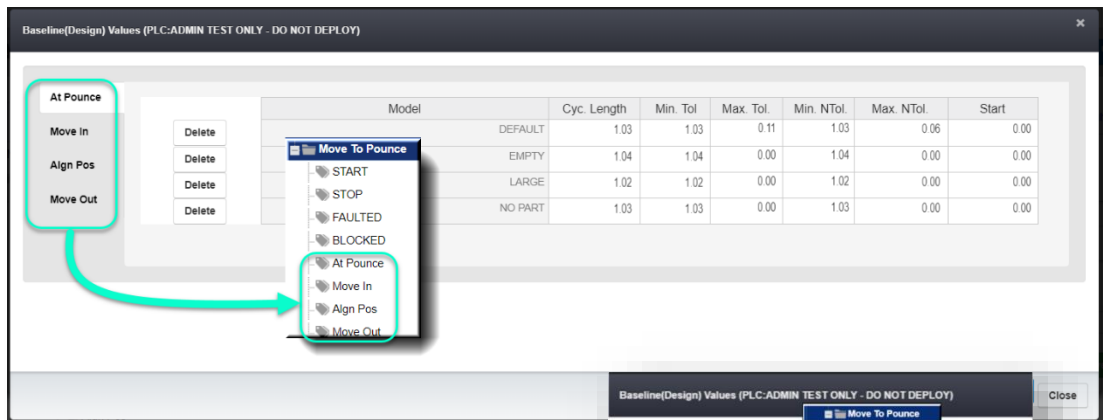
Integer value used to select which model to use when multiple models' codes are collected during a cycle. It is since a lower number is a higher priority. As in the number 1 priority would be higher than 2 or 3.

Once the Model data has been created, baseline values can be entered for the Groups that comprise the Asset. This is shown in the following screenshots. Drag the cursor to the left-hand edge of the screen for the Admin menu to appear. Click on **Objects & Relationships** to go back to the **Factory** window.

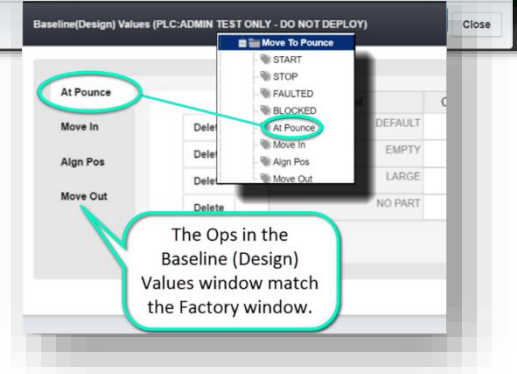


Expand the tree in the Factory window as shown. Then highlight the **CYCLE TIME (GROUP level)** and click on the **Baseline Values** button. A window will pop up with a table for entering Baseline data parameters.

The Baseline (Design) Values window will reveal. From here, the values of the Group Ops are inputted. In this example, all the values are inputted in for **At Pounce**.



There are 3 other objects, **Move In**, **Align Pos**, and **Move Out**, that will need to have the values inputted as well. Click on the object tabs to input your values.



# Baselines and Tolerances

[Baselines and Tolerances](#) | [Baseline Values](#) | [Tolerances](#)

Baselines are used to signify the expected or designed duration each motion will experience. Tolerances are used to denote how much variation from these baselines there can be before the motion is flagged. Motions will be flagged as “Watch” – the cycle duration exceeds the “NTol” value – or they will be flagged as “Warning” – the cycle exceeds the “Tol” value – which will display these motions on the “Hotspots” screen and are also used for reporting purposes.

There are 3 generally recommended – *but not fully inclusive* – methods of baseline and tolerance selection:

Choose a baseline value slightly above the normally expected cycle duration with tighter tolerance values.

- Useful for setting standard tolerance values for common pieces of tooling.
- Generally tight variance cycles where a small overage denotes an immediate issue.

Choose a baseline value of an average expected cycle duration with wider tolerance values.

- A more variable normal cycle duration useful for tracking cycles which exceed normally expected variance.
- Generally tight variance cycle where a larger overage is allowed.

Choose a baseline value as a hard limit with zero values for tolerances.

- CycleTime group where an overage constitutes a machine over cycle condition.
- Operator motions where a maximum time is given and any overage is considered an over cycle condition.
- An agreed upon maximum limit for a common piece of tooling used to track tooling degradation for predictive maintenance purposed.

## Object Classifications

Object Classifications are used to label common types of objects for easier usage for reporting purposes. By creating and assigning classifications to individual Group motions for each Asset, these objects can quickly be sorted and combined for reporting and data analysis. Commonly used classifications are Operator, Tooling, Robot, and Joining. These classifications are fully user definable, and any group may be assigned multiple classifications.

The screenshot shows the 'Admin' interface for 'enVision'. The main content area is titled 'Maintain Object Classifications'. It features a table with the following columns: Name, Cycle Length, Min. Tol., Max Tol., Min. Normal Tol., and Max. Normal Tol. The table lists various object classifications such as 'Apply Schedule Factor', 'ATest', 'COUNTPOINT', 'FakeObject', 'Lefthand', 'Operator', 'Pin', 'Righthand', 'Robot', 'TestA', 'TestA2', 'TestB', 'TestB2', 'TestC2', 'TestC3-3', and 'Tooling'.

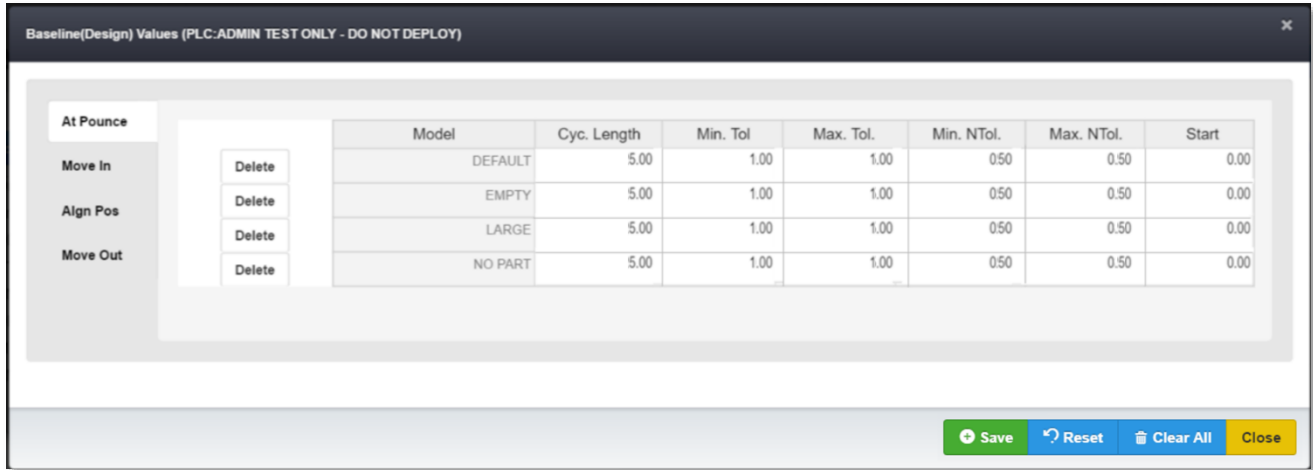
Name	Cycle Length	Min. Tol.	Max Tol.	Min. Normal Tol.	Max. Normal Tol.
Apply Schedule Factor	0	0	0	0	0
ATest	12	0.2	0.2	0.2	0.2
COUNTPOINT	0	0	0	0	0
dddssddd-555555	0	0	0	0	0
FakeObject	10	0	0	0	0
Lefthand	0	0	0	0	0
Operator	0	0	0	0	0
Pin	12	1	0	0	0
Righthand	0	0	0	0	0
Robot	0	0	0	0	0
TestA	12	0.2	0.2	0.2	0.2
TestA2	0	0	0	0	0
TestB	0	0	0	0	0
TestB2	0	0	0	0	0
TestC2	0	0	0	0	0
TestC3-3	10	0.1	0.1	0.1	0.1
Tooling	1.5	0.15	0.1	1.5	1.5

## SmartNotes

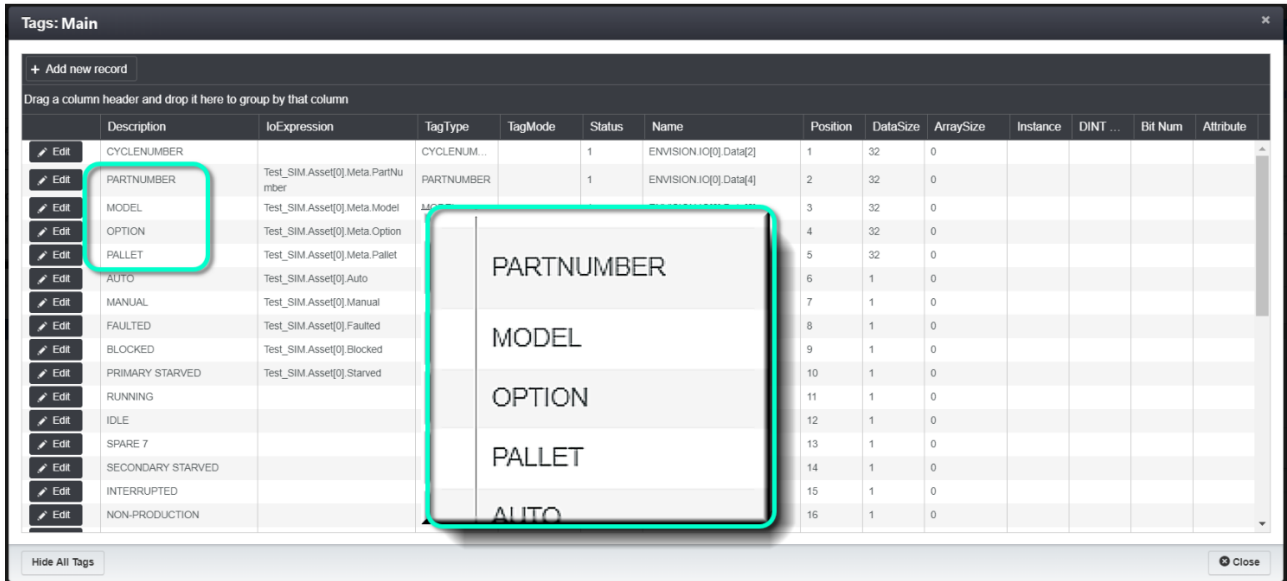
Notes are used to add descriptions to a given cycle. Note categories and subcategories can be created or modified in the admin area of enVision. When a note is created on a given cycle, any previously created note category can be assigned to this note, and these notes can then be sorted and summarized based on their category. Notes are very useful for describing reasons and conditions for abnormal cycles; these notes remain persistent and are available to all users. See [Note Categories](#).

# Baseline Values

When the **Baseline Values** button is selected, the following window pops up. It is here where the data is entered for baseline (Cycle Time) and tolerances for the models that were created previously. This is the data that the enVision application uses to create the graphical representation of the cycles. For this reason, the MODEL Asset tag is of critical importance. enVision needs a **value** for the MODEL Asset tag or it will not know how to “draw” the cycle.



Every Group of the Asset will have a similar window that pops up when the Group object is highlighted, and the **Baseline Values** button is clicked. For Groups objects with 2 or more OP’s, each OP will need values entered individually as required. There were additional Asset Tags that appeared when the Show All Tags button was clicked. These tags can be used to display additional information for each cycle.

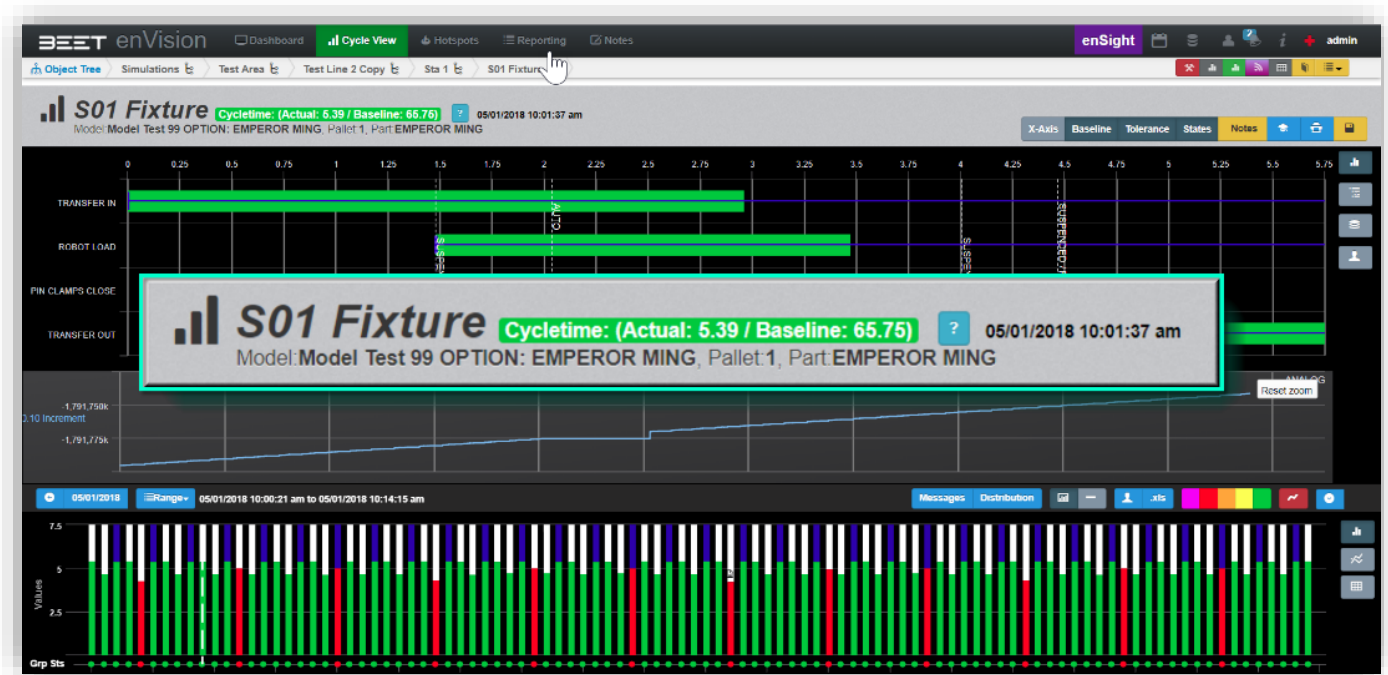


The **CYCLENUMBER** tag does not need an **IoExpression** and will be **ignored** if one is entered. This tag is automatically updated by the enVision PLC driver logic. The tag is displayed for informational purposes only.

The **PARTNUMBER** tag can be used to store a part tracking value such as a sequence number or rotation number if the process utilizes some sort of part monitoring. This is optional. If no IoExpression is entered, then enVision will display a zero.

The **OPTION** tag is currently not used by the enVision application for any display or calculation purposes. It is for future use. A value can be entered for the IoExpression but it will be ignored by the application.

The **PALLET** tag is fairly self-explanatory. It can be used to store part conveyance data such as a conveyor carrier I.D. number or a roller bed pallet I.D. This is optional. If no IoExpression is entered, then enVision will display a zero.



To edit model baselines and tolerances, go to any group object, then click the baseline tab at the top of the window. **CycleTime** baseline values will determine when the Asset starts and stops all other baselines will only affect Groups.

[Baseline Values](#)
[Deploy](#)
[Import](#)
[Export](#)
[Config](#)
[Tags](#)

Rough Part 1 Pick From Infeed Complete		Model	Cyc. Length	Min. Tol	Max. Tol.	Min. NTol.	Max. NTol.	Start
Gripper 1 Closed	Delete	10.7L	5.00	1.00	1.00	0.50	0.50	2.50 <sup>0</sup>
	Delete	10.7L (0)	5.00	1.00	1.00	0.50	0.50	2.50 <sup>0</sup>
	Delete	10.7L (180)	5.00	1.00	1.00	0.50	0.50	2.50 <sup>0</sup>
	Delete	10.7L NOK	5.00	1.00	1.00	0.50	0.50	2.50 <sup>0</sup>
	Delete	12.8L	5.00	1.00	1.00	0.50	0.50	2.50 <sup>0</sup>
	Delete	12.8L (0)	5.00	1.00	1.00	0.50	0.50	2.50 <sup>0</sup>
	Delete	12.8L (180)	5.00	1.00	1.00	0.50	0.50	2.50 <sup>0</sup>
	Delete	12.8L NOK	5.00	1.00	1.00	0.50	0.50	2.50 <sup>0</sup>

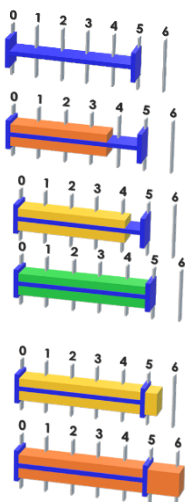
Baseline tables will differ from one another in terms of how many models each asset has, and how many OPs each group has. The values inputted in each cell will depend on how each model tied to the asset is intended to function.

# Tolerances

[Baselines and Tolerances](#) | [Baseline Values](#) | [Tolerances](#)

## Overview

The Tolerance's are preset in enVision to give a visual representation of the health of the cycle with a glance. In sequence the base of the cycle is the Cycle Length. Cycle Length is represented by a Blue bracket on the graph. In a recorded cycle, it will show the length as compared to the Baseline tolerance. It will show a color based upon the performance of that cycle. Green for **Good Cycle Length**, **Yellow to Watch**, and **Orange** to give the operator a **Warning**. Below are the definitions of each in the Sequence and Heartbeat Views.



**Cycle Length** baselines are represented by a **|—blue—|** bracket in enVision. Ideally, group length will match this bar exactly.

**Minimum Tolerance** is the amount a group or cycle can deviate below the cycle time before becoming an **Orange (Warning)** record.

**Minimum Normal Tolerance** is the amount a group or cycle can deviate below the cycle time before becoming a **Yellow (Watch)** record. The NTol number should always be smaller than the regular Minimum Tolerance (Min Tol).

A **Good Cycle Time** record will be **Green (Good)** if the amount of deviation above or below ideal cycle time is smaller than any tolerances.

**Maximum Normal Tolerance (Max NTol)** is the amount a group or cycle can deviated above ideal cycle time before turning **Yellow (Watch)**. The MaxNTol number should always be smaller than the Maximum Tolerance (Max Tol) number.

**Maximum Tolerance (Max Tol)** is the amount a group or cycle can deviate above the cycle time before becoming an **Orange (Warning)** record.

## All in Heartbeat View

Cyc. Length is the expected duration of the motion.

### Blue Line in Heartbeat Mode

Min Ntol and Max NTol are the allowed deviation from the cycle length to remain green, outside of this it becomes yellow. This is input into the box as an offset from the Cycle Length; it is not an absolute number.

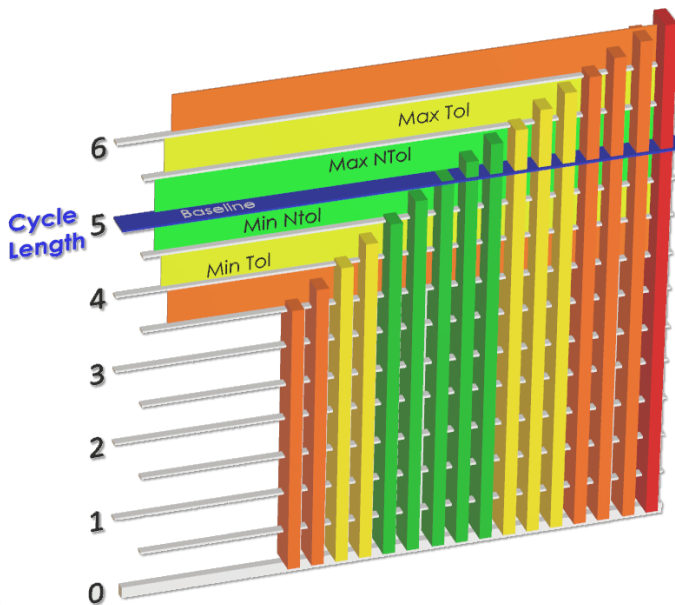
Example: Cyc Length of 5s with Min/Max NTol of .5. What this means is that the cycle will remain green from 4.5 to 5.5 seconds and yellow at  $T < 4.5$  and  $T > 5.5$

Min/Max Tol are the allowed deviation from the cycle length to remain yellow. The Tol value are expected to be a larger value than the NTol. To continue the example above, if we set the min/max Tol to be 1s then the color of the bars is set as follows:

Green cycle =  $4.5s \text{ (cyc. Length - Min Ntol)} < T < \text{(cyc. Length + Max Ntol)} \text{ 5.5s}$

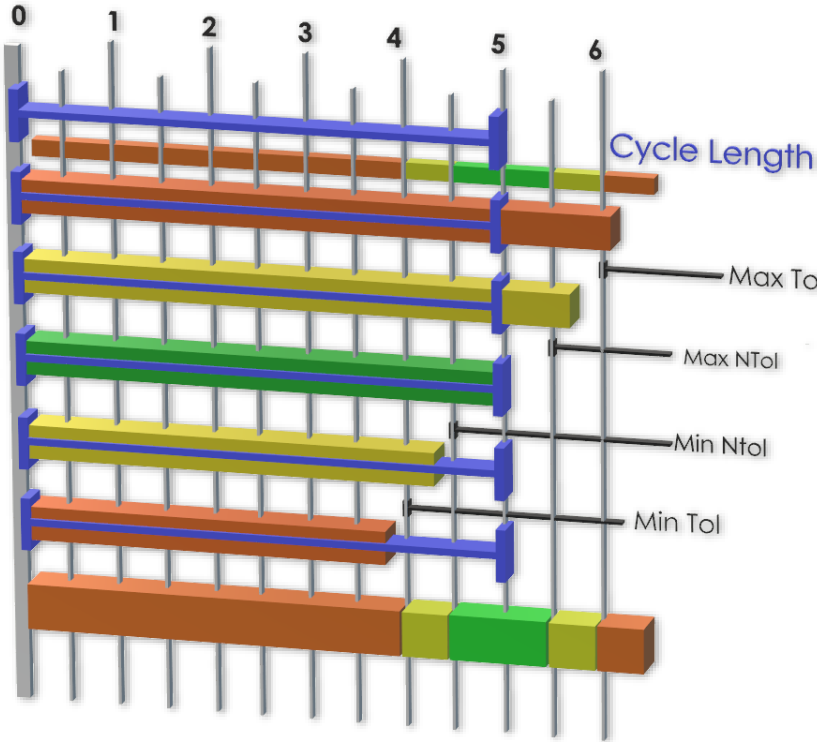
Yellow cycle =  $4s \text{ (cyc. Length - Min tol)} < T < \text{(cyc. Length - Min Ntol)} \text{ 4.5s OR } 5.5s \text{ (cyc. Length + Max Ntol)} < T < \text{(cyc. Length + Max tol)} \text{ 6s}$

Orange =  $T < \text{(cyc. Length - Min tol)} \text{ 4s OR } T > \text{(cyc. Length + Max tol)} \text{ 6s}$



Tolerance Reference sheet

This example shown in Sequence mode also shows the actual details if you hovered the cursor over the cycle bar.



Cyc. Length is the expected duration of the motion.  
Blue bracket in Sequence Mode

Min Ntol and Max NTol are the allowed deviation from the cycle length to remain green, outside of this it becomes yellow. This is input into the box as an offset from the Cycle Length; it is not an absolute number.

Example: Cyc Length of 5s with Min/Max NTol of .5. What this means is that the cycle will remain green from 4.5 to 5.5 seconds and yellow at  $T < 4.5$  and  $T > 5.5$

Min/Max Tol are the allowed deviation from the cycle length to remain yellow. The Tol value are expected to be a larger value than the NTol. To continue the example above, if we set the min/max Tol to be 1s then the color of the bars is set as follows:

Green cycle =  $4.5s \text{ (cyc. Length - Min Ntol)} < T < \text{(cyc. Length + Max Ntol)} \text{ } 5.5s$

Yellow cycle =  $4s \text{ (cyc. Length - Min tol)} < T < \text{(cyc. Length - Min Ntol)} \text{ } 4.5s \text{ OR } 5.5s \text{ (cyc. Length + Max Ntol)} < T < \text{(cyc. Length + Max Tol)} \text{ } 6s$

Orange =  $T < \text{(cyc. Length - Min tol)} \text{ } 4s \text{ OR } T > \text{(cyc. Length + Max tol)} \text{ } 6s$

**S01 Fixture**  
 Model: Model Test 1 OPTION: 559, Pallet: 38, Part: 559, Cycle Number: 187469733647572017  
 08/16/2017 5:39:49 am  
 Cycle time: (Actual: 22.61 / Baseline: 37.84)

**Cycle Length**  
 Cycle Length baselines are represented by a blue bracket in Envision. Ideally, group length will match this bar exactly.

**Min. Tol**  
 Minimum tolerance is the amount a group or cycle can deviate below the cycle time before becoming an Orange (Warning) record.

**Min. NTol**  
 Minimum normal tolerance is the amount a group or cycle can deviate below the cycle time before becoming a Yellow (Watch) record. The NTol number should always be smaller than the regular Min Tol.

**Good Cycle Time**  
 A record will be Green (Good) if the amount of deviation above or below ideal cycle time is smaller than any tolerances.

**Max NTol.**  
 Max normal tolerance is the amount a group or cycle can deviate above ideal cycle time before turning Yellow (Watch). The MaxNTol number should always be smaller than the Max Tol. number.

**Max. Tol**  
 Maximum tolerance is the amount a group or cycle can deviate above the cycle time before becoming an Orange (Warning) record.

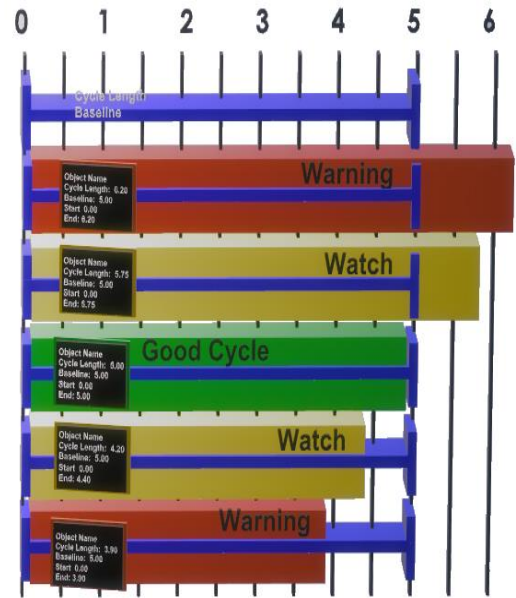
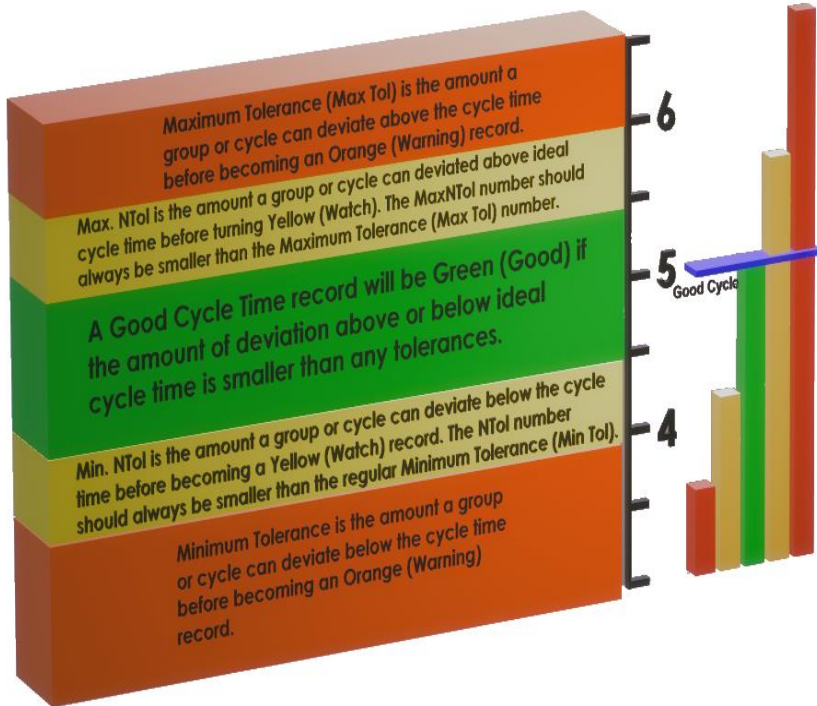
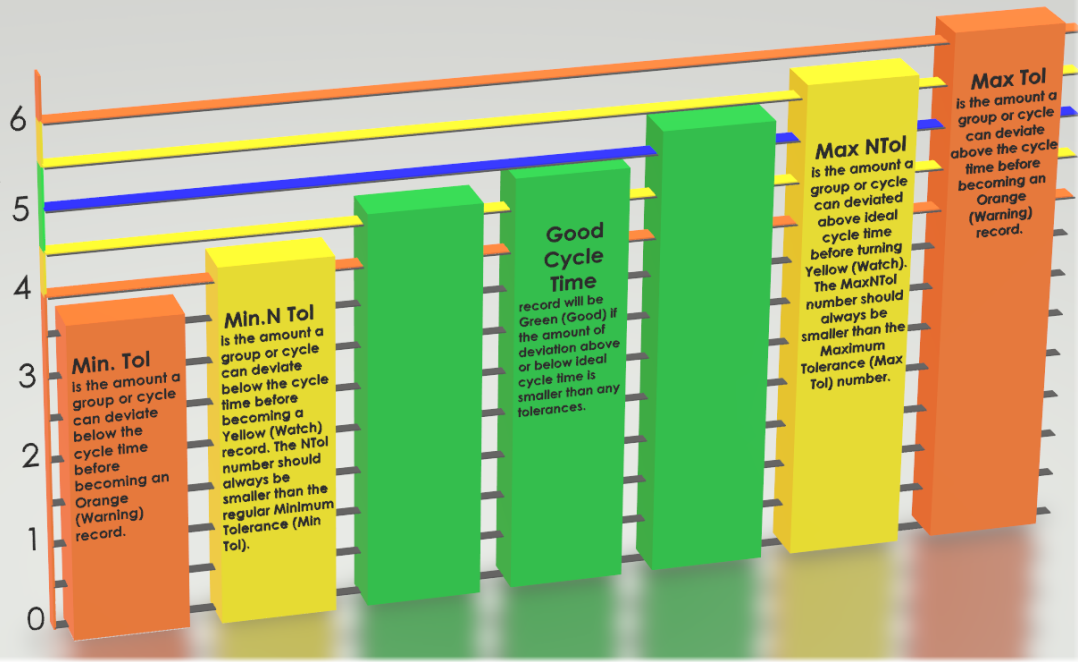
08/16/2017 Range 08/16/2017 4:40:30 am to 08/16/2017 6:00:41 am

Tolerances in Heartbeat View

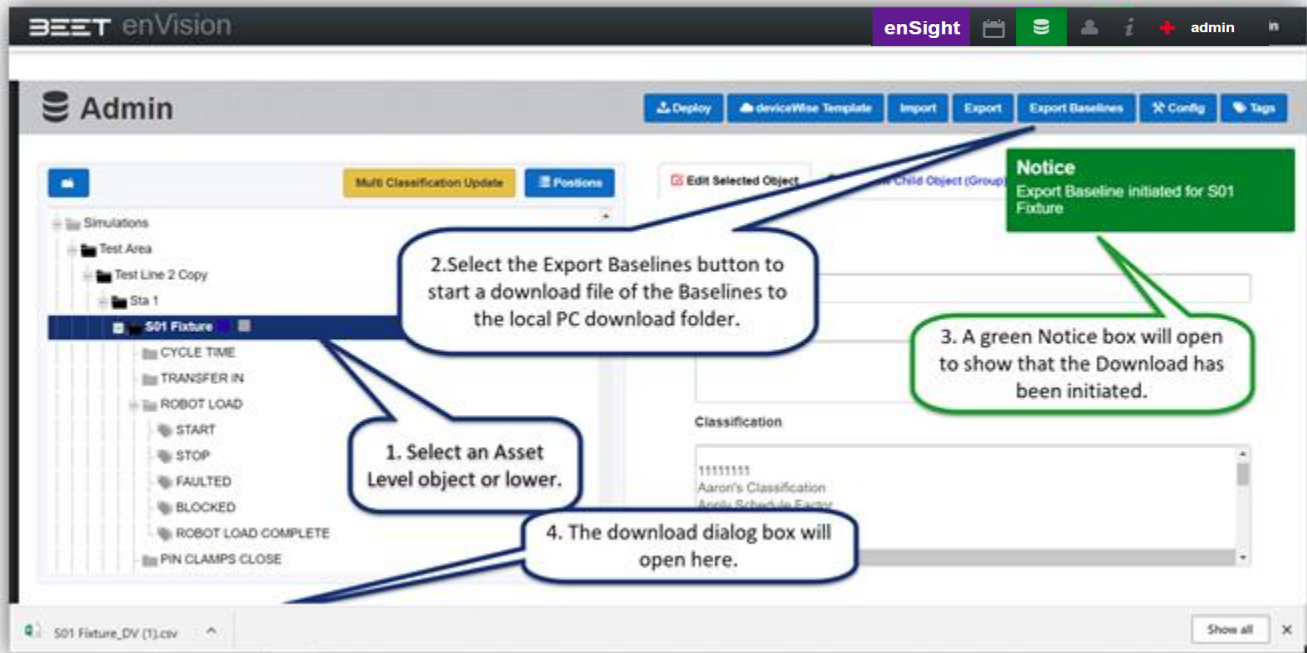
Cycle Length baselines are represented by a **Blue Line** in the Heartbeat Mode.

**Example Specs set for a 5 second cycle**

Cycle Length = 5 seconds  
 Min Tol = 1 second  
 Min. NTol = 0.5 second  
 Max NTol = 0.5 second  
 Max Tol = 1 second

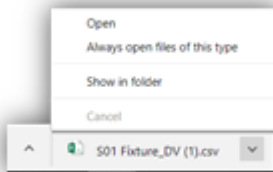


# Exporting Baseline and Tolerances

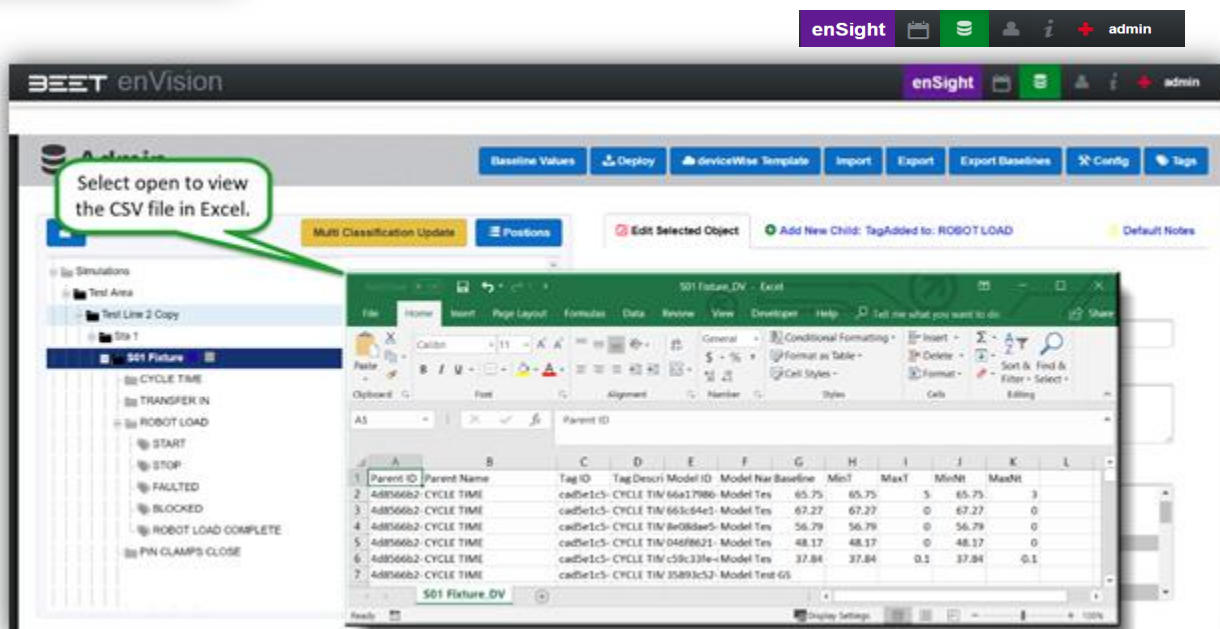


Select an asset level object, then click export baselines button to create a CSV download of all groups/ops/model combinations under that asset. A green Notice box will appear that will show that the download has been initiated.

A dialog box will appear below the browser window showing you the download in progress. Once it is complete, the user can open, or show in the downloads folder.

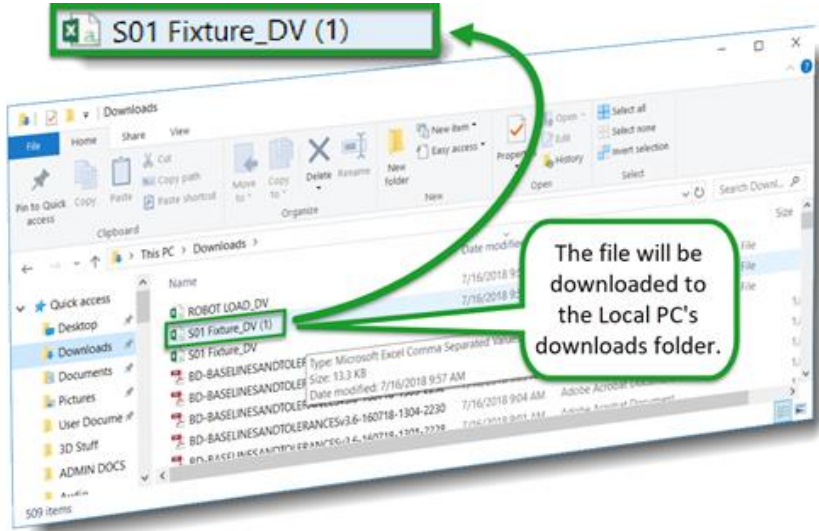


Click open and it will open the file in excel.

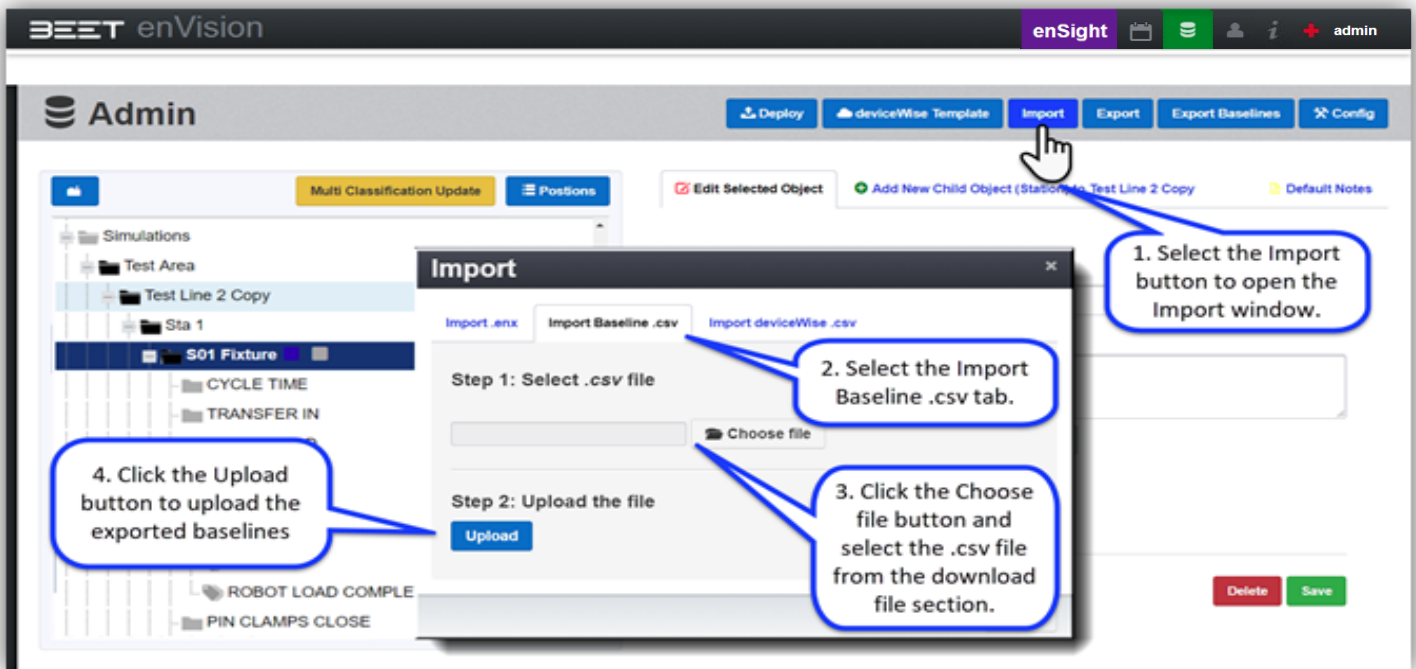




Click the Show in Folder tab to show it in the Download folder of the local PC.



You can enter baselines and tolerances, in the excel file, then upload back into enVision



Import screen has additional tabs to import baselines from the CSV generated from Step2



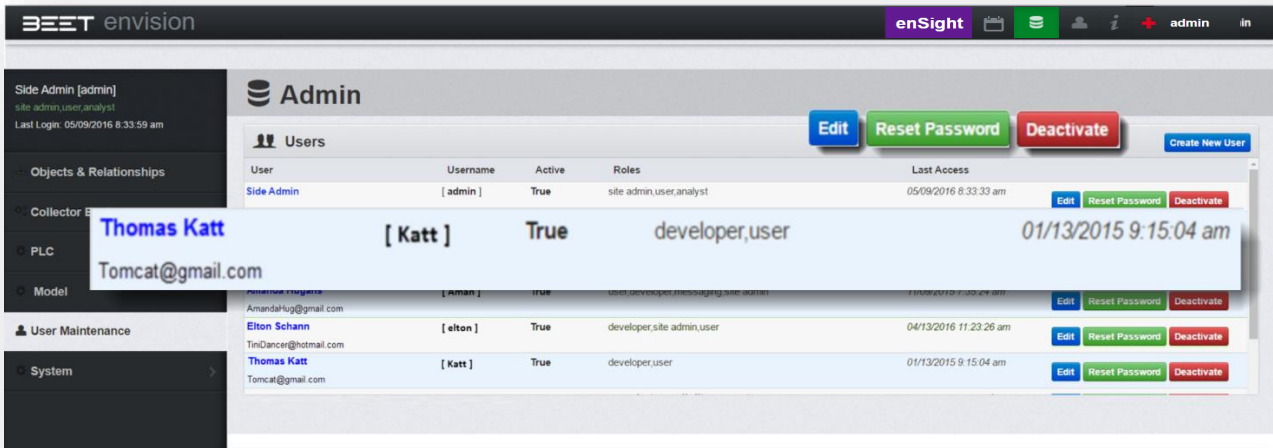
# User Maintenance ▼

[User Maintenance](#) | [Overview](#) | [Create New User](#) | [Changes and Editing Profile](#)

## Overview

This section of the Admin module contains the data table of all the created user accounts. To use or make changes to the administration module you will need to be registered in this database.

- Developers** - is mostly for the **BEET developers**, it allows the **hidden objects and information** to be seen when logged in at that level.
- Analyst** – Allows the user to fully use/editing of the **Dashboard and Reporting Modules**. You also need to have an **Analyst level** to utilize the **Conditional Notifications and Links to Subscriptions**.
- Site Admin** – allows the full control of the Admin module and will allow the user to make changes as well as **global changes**.
- Site Leader** – privilege for user – has same level as **Site Admin** except privilege to create new users.

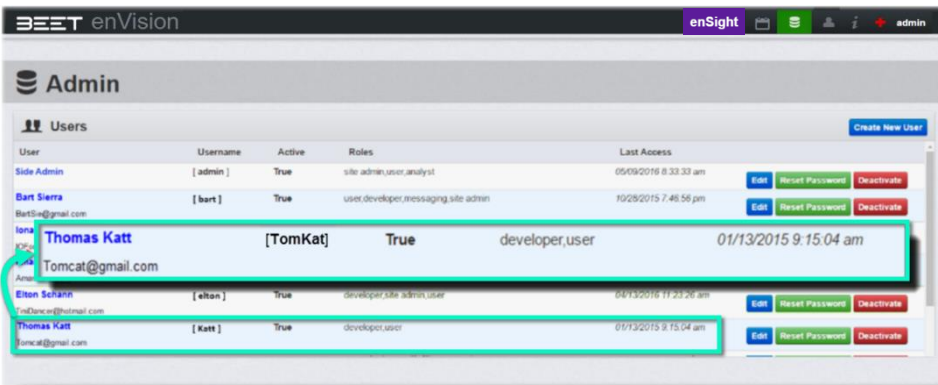
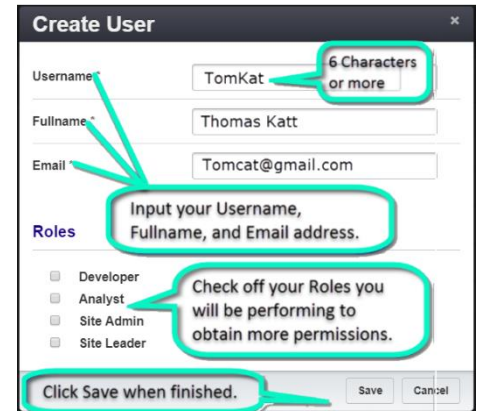


## Create New User



To add a new user, click the Create New User button in the upper right side of the User Maintenance Table window.

Once selected, a Create User window will appear. Input a Username, Full name, and Email address in the required fields. Below the edit boxes, is the Roles list. Check off the roles you will be performing. These roles will add permissions and allowances for using the Admin Module. After you are finished, click the save button.



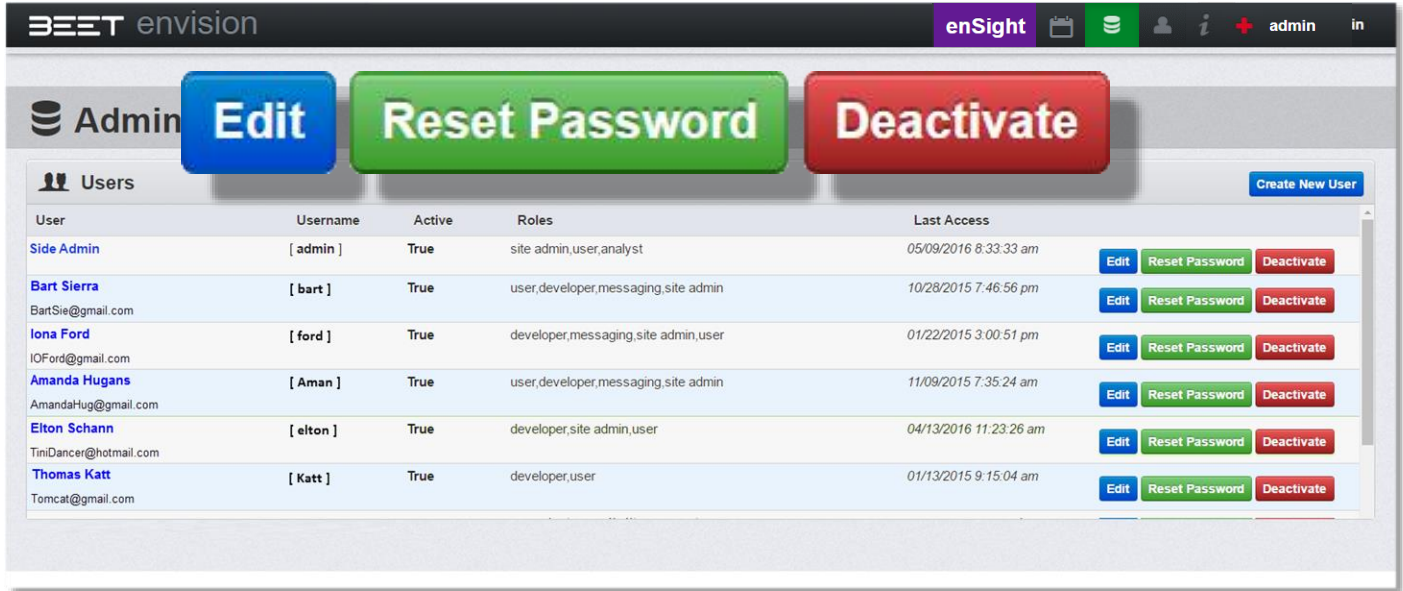
The User Maintenance List will now have the new user in the database. To make any changes, use the buttons to the right side of the User Maintenance window.

# Changes and Editing Profile

[User Maintenance](#) | [Overview](#) | [Create New User](#) | [Changes and Editing Profile](#)

**Edit**

On occasion, changes to a profile may be required or a profile may need to be removed. The User Management window has the options to make changes by using three different buttons, Edit, Reset Password, and the Deactivate buttons. All buttons are located on the right side of the User Management table.



Click Edit button and an Update User window will open. Make the changes to Full name, Email address, and check or uncheck additional Roles.

When your changes are completed, click the [Save](#) button.

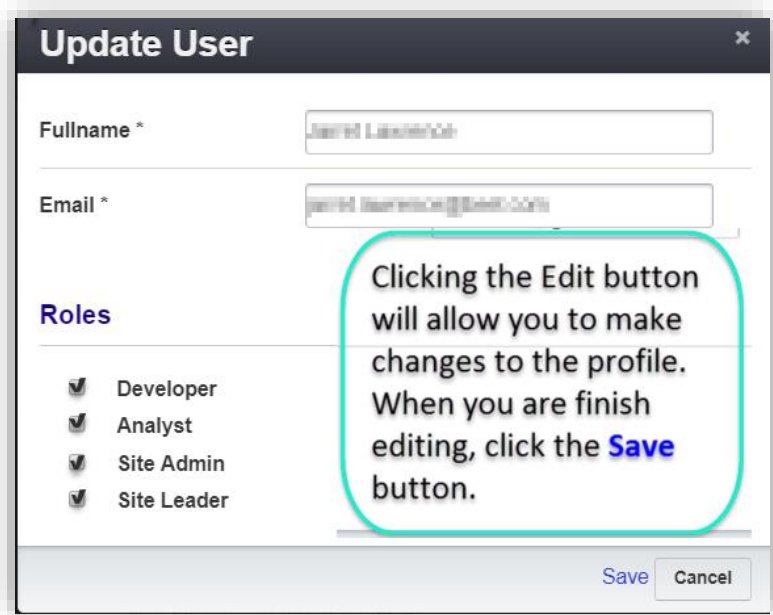
**Reset Password**

In the event you lose or forget your password, you can have it automatically reset if needed.

Clicking the **Reset Password** button will send you an email with a new password. Ensure that your email is correct. If you don't see it in your inbox, check the spam or junk folder.

**Deactivate**

Clicking the **Deactivate** button will disable and remove the selected profile from the database.



# System

[System Configuration](#) | [Warnings](#) | [State Display Names](#) | [Object Classification](#) | [Note Categories](#) | [Date and Time](#) | [Nomenclature](#) | [EDC Agent Settings](#) | [SMTP Settings](#) | [Subscriptions](#)

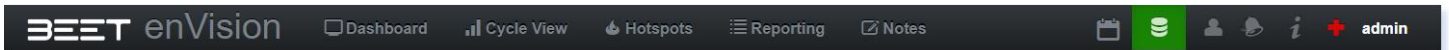
## System Configuration

The System section tab has many different sections that have the setting and configuration. The System Configuration tab will reveal the sub section to the right.



## Warnings

If this message appears anywhere on a window you are viewing, take caution. Any changes made will change all global settings immediately on the network. Be **very cautious** when performing any actions or changes on any window with this warning.



The screenshot shows the BEET Admin interface. On the left is a dark sidebar with a menu. The 'System' menu item is expanded, and 'System Configuration' is highlighted with a white mouse cursor. The main content area is titled 'Admin' and contains a list of configuration options: State Display Settings, Object Classifications, Note Categories, Date & Time, Nomenclature, EDC Agent Settings, SMTP Settings, and Other. A large red warning box is overlaid on the right side of the main content area, containing the text: **Stop!** Any change made are global and immediate!!!

# State Display Settings

[System Configuration](#) | [Warnings](#) | [State Display Names](#) | [Object Classification](#) | [Note Categories](#) | [Date and Time](#) | [Nomenclature](#) | [EDC Agent Settings](#) | [SMTP Settings](#) | [Subscriptions](#)

The **State Display Names** Tab is used to edit the default names, colors, and appearance of the objects and display names. **NOTE:** **These changes made are global and immediate!!!**

The names of the States all have a default name. If you wanted to use another name other than the default, simply use the edit box under the **Override** column, to the right of the **State (default)** name, to change it to what you desire.

**Override** allows a user to change from the default name of the state to a desired name. Use the edit box to change the state.

Click on the color to change and a color pallet will reveal for the user to change the color of the state.

Check or uncheck the State Display setting of what the user wants to Show in Live-View.

Click on the color you would like to change and use the color pallet to pick another color.

The states have a default color code for each state, but they can be modified as well. Simply click on the **colored box** and a **color pallet** will appear. Choose the color, tone, and hue you desire.

At the far right side is a column labeled **"Show in Live View"**. Check and uncheck the boxes to show this state in the **Live View**

**Module.** When everything is set, click the **Save** button.

The Re-named States appear here on the Live View.

# Object Classifications

[System Configuration](#) | [Warnings](#) | [State Display Names](#) | [Object Classification](#) | [Note Categories](#) | [Date and Time](#) | [Nomenclature](#) | [EDC Agent Settings](#) | [SMTP Settings](#) | [Subscriptions](#)

The Object Classification tab controls the Classifications on the Edit Select object window, under the Objects and Relationships window. Any changes that you make here in this window will display when viewing Group level objects in the Objects and Relationship window.

These Classifications created here, are used for Grouping, Sorting and for other purposes used in the reports. Simply double-click an Object Classification and then edit it to read what you desire.

If a field has a Lock icon, then this Object Classification cannot be changed in this window.

There is an open field to enter a new Object Classification if needed. Simply click on it and enter the Classification.

Name	Cycle Length	Min. Tol.	Max Tol.	Min. Normal Tol.	Max Normal Tol.
Apply Schedule Factor					
ATest	12	0.2	0.2	0.2	
COUNTPOINT	0	0	0	0	
ddddssddd-555555	0	0	0	0	
FakeObject	10	0	0	0	
Lefthand	0	0	0	0	
Operato	0	0	0	0	
Pin	0	1	0	0	
Righthand	0	0	0	0	
Robot	0	0	0	0	
TestA	12	0	0.2	0.2	0.2
TestA2	0	0	0	0	0
TestB	0	0	0	0	0
TestB2	0	0	0	0	0
TestC2	0	0	0	0	0
TestC3-3	10	0.1	0.1	0.1	0.1
Tooling	1.5	0.15	0.1	1.5	1.5

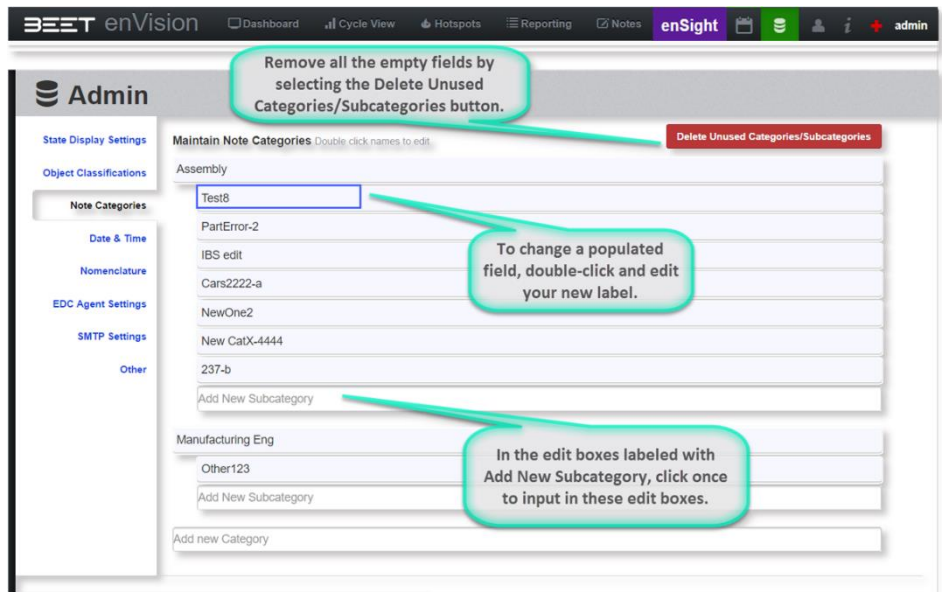
When you are finished, these Classifications will appear on the Objects and Relationships window, when you have a Group level object selected. The **Edit Selected Object** tab is selected, as seen here.

**Adding, Delecting, or Editing in the Object Classification section will show up here in the Edit Select Object section, in the Classifications Tab.**

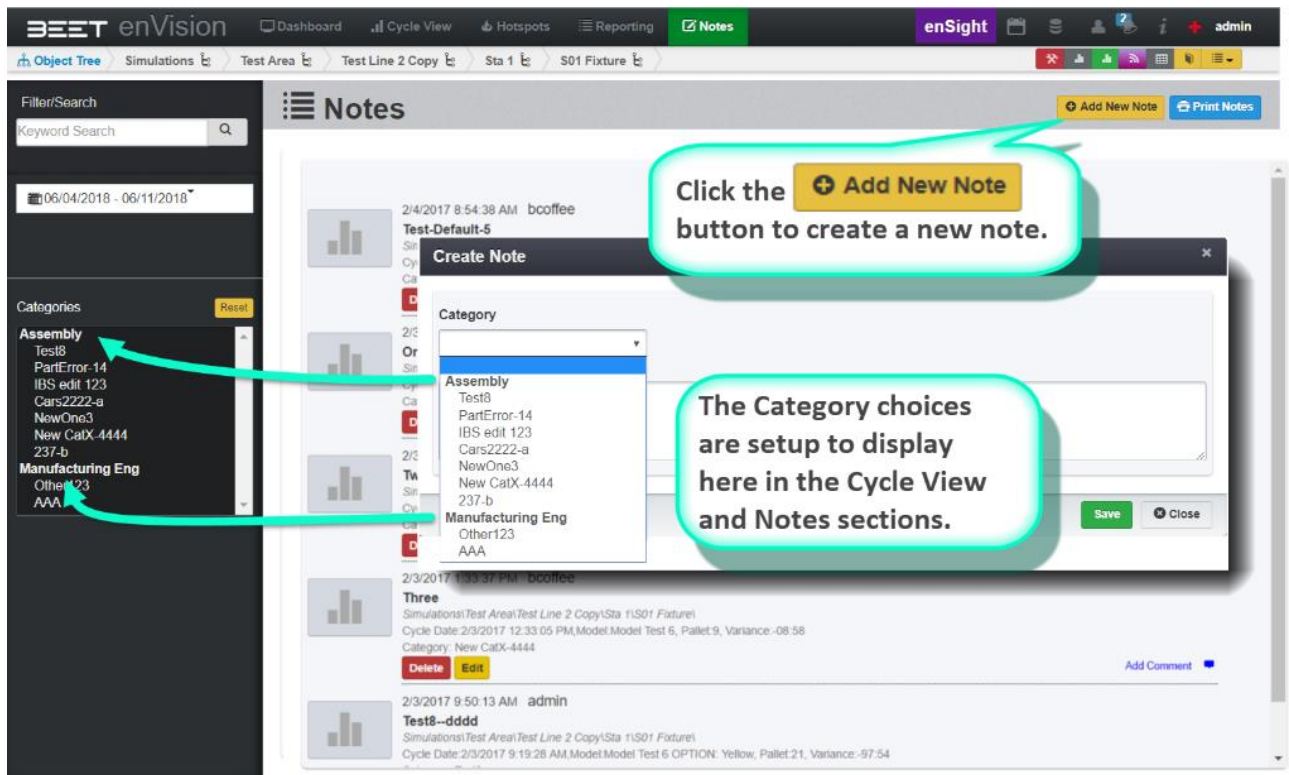
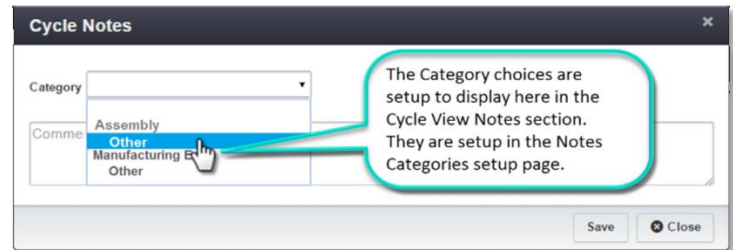
# Note Categories

[System Configuration](#) | [Warnings](#) | [State Display Names](#) | [Object Classification](#) | [Note Categories](#) | [Date and Time](#) | [Nomenclature](#) | [EDC Agent Settings](#) | [SMTP Settings](#) | [Subscriptions](#)

The Notes Categories tab holds the key to configuring the Notes Category list in the Cycle View Browser. From here, you can add, remove, or edit the Categories that will appear in the Cycle View Notes. To add a new Category or Subcategory, click on the editing box with the labels starting with "Add New". Input your new Category or Subcategory. For an already existing Category or Subcategory, you can double click on it to edit the field.



In the Cycle View browser, when viewing an **Asset** or **Group** level object you can click on the **+Add New Note** button to create a note. Shown here is a screenshot from the **Notes** browser section. When the Category pulldown is selected, it shows the selections you made in the Note Category List editor.

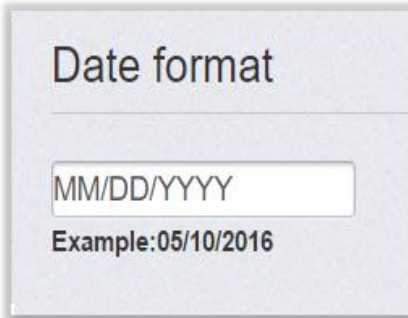


# Date and Time

[System Configuration](#) | [Warnings](#) | [State Display Names](#) | [Object Classification](#) | [Note Categories](#) | [Date and Time](#) | [Nomenclature](#) | [EDC Agent Settings](#) | [SMTP Settings](#) | [Subscriptions](#)

This section controls the Date and Time formats that appear in the enVision, Tables, and Data.

## Date Format

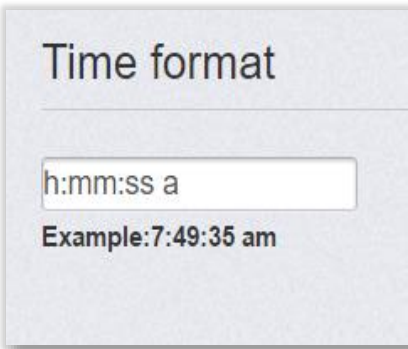


To change the date format, **click** in the Date Format editing box and delete the previous format, if there was one present before. Input a different format style in its place. If it is recognized by the program, then it will show an **Example** below the Edit box.

### Date formats examples:

- American MM/DD/YYYY
- European-DD.MM.YYYY
- Japanese-YYYY-MM-DD

## Time Format



To change the time format, it is very similar to changing the date format. **Click** in the Time Format editing box and delete the previous format, if there was one present before. Input a different format style in its place. If it is recognized by the program, then it will show an **Example** below the Edit box.

### Common Time Formats:

- h:mm:ss a / 1:30:00 p.m.
- hh:mm / 13:30
- h:mm / 1:30

When your changes are complete, click the **Save** button and the changes will update in the program.

**NOTE:** These changes made are global and immediate!!!

**Configure the formats for Date and Time on how it is displayed in the ENVISION Browser.**

**Common Date Formats:**  
 American: MM/DD/YYYY  
 European: DD.MM.YYYY  
 Japanese: YYYY-MM-DD

**Common Time Formats:**  
 h:mm:ss a /1:30:00 p.m.  
 hh:mm / 13:30  
 h:mm /1:30

**Stop!**  
Any change made are global and immediate!!!

**Save**



# Nomenclature

[System Configuration](#) | [Warnings](#) | [State Display Names](#) | [Object Classification](#) | [Note Categories](#) | [Date and Time](#) | [Nomenclature](#) | [EDC Agent Settings](#) | [SMTP Settings](#) | [Subscriptions](#)

Because of different industry standards, some companies have different definitions and keywords for certain parts of their company. Based upon company needs, it may be necessary to rename one or more of the **Default Objects**, **Part-type**, **Parts**, and **Pallet**. The Default Object names are common and/or typical names in manufacturing.

**NOTE:** **These changes made are global and immediate!!!**

In this case, we would like to change the name for the **Line** (default) object from **Line** to **Segment**. Click on the **Editing box** for **Line**. Delete the previous name (**Line**). Then input the name (**Segment**) you desire.

The screenshot shows the BEET enVision Admin interface. The top navigation bar includes 'enSight' and 'admin'. The left sidebar shows 'Admin' with various settings categories. The main content area displays the 'Nomenclature' settings for the 'Line' object. A 'Stop!' warning message is visible at the top. The 'Line' object is selected, and its name is being edited from 'Line' to 'Segment'. Callouts provide instructions: 'Click in the editing box.', 'Delete the previous name.', and 'Input your desired name.'

After your changes, review your changes and click **Save** to complete the process.

# EDC Agent Settings

[System Configuration](#) | [Warnings](#) | [State Display Names](#) | [Object Classification](#) | [Note Categories](#) | [Date and Time](#) | [Nomenclature](#) | [EDC Agent Settings](#) | [SMTP Settings](#) | [Subscriptions](#)

The EDC Agent Configuration screen is used to set the Heartbeat Update Delay. Since the Agent is continually trying to contact the EAS, a constant contact would cause a strain on the EAS. Therefore, this configuration setup feature will allow the EDC Agent to connect to the EAS and update at a specified interval (120 seconds in this example). The inputted number must be greater than 59 seconds.

**Admin**

- [State Display Settings](#)
- [Object Classifications](#)
- [Note Categories](#)
- [Date & Time](#)
- [Nomenclature](#)
- EDC Agent Settings**
  - [SMTP Settings](#)
  - [Other](#)

## EDC Agent Configuration

**Stop!**  
Any change made are global and immediate!!!

Heartbeat Update Delay  (Must be greater than 59 seconds. Default is 120 seconds.) [Save](#)

As a result of constant contact from an **EDC agent**, the **EAS** would experience too much demand from the **EDC Agent**. In the **EDC Agent Configuration** setup feature will allow the EDC Agent to connect to the EAS at a specified interval (120 seconds in this example). The inputted number must be greater than 59 seconds. Input the number of seconds you desire, then click the **Save** button.

**NOTE: THESE CHANGES ARE GLOBAL AND IMMEDIATE!!!**

Input the number of seconds you desire, then **click** the [Save](#) button. **NOTE:** **These changes made are global and immediate!!!**

# SMTP Settings

[System Configuration](#) | [Warnings](#) | [State Display Names](#) | [Object Classification](#) | [Note Categories](#) | [Date and Time](#) | [Nomenclature](#) | [EDC Agent Settings](#) | [SMTP Settings](#) | [Subscriptions](#)

**Simple Mail Transfer Protocol (SMTP)** is an Internet standard for electronic mail (email) transmission. SMTP by default uses TCP port 25. For a more secure port, it will allow you to change to a more secure port (e.g. 465 or equivalent).

This page allows you to input the Host Server Name (the computer representing the SMTP server), your choice of port to use, the email of the originating message, login information, and the option to use SSL (Secure Socket Layer) encryption.

**NOTE:** These changes made are global and immediate!!!

The screenshot shows the BEET enVision Admin interface. The top navigation bar includes the BEET logo, 'enVision', and user information 'enSight', 'admin', and 'in'. The left sidebar lists various settings categories, with 'SMTP Settings' selected. The main content area is titled 'SMTP Email Server Configuration' and features a prominent red warning box: 'Stop! Any change made are global and immediate!!!'. Below the warning, there are input fields for 'Host Server Name' (localhost), 'Port' (25), 'eMail From' (noreply@beet.com), 'SMTP User Id', and 'SMTP User Password'. A 'Use SSL' checkbox is also present. A blue 'Save' button is located to the right of the form. Five callout boxes provide additional information: 'The name of the computer name that is representing the SMTP Server.' points to the Host Server Name field; 'Typically use 25. For more secure, use a different port made for security.' points to the Port field; 'This is the address of the message originator.' points to the eMail From field; 'Log in and Password for the User.' points to the SMTP User Id and Password fields; and 'Check off to use Secure Socket Layer encryption.' points to the Use SSL checkbox.

# Others – Subscriptions and Default Dashboard

[System Configuration](#) | [Warnings](#) | [State Display Names](#) | [Object Classification](#) | [Note Categories](#) | [Date and Time](#) | [Nomenclature](#) | [EDC Agent Settings](#) | [SMTP Settings](#) | [Subscriptions](#)

In the enVision system, an email is automatically sent in the instances of certain events such as a fault, stoppage, or whatever is set in the configuration. For this reason, there is a configurable option to delay the automatic email. The Email Delay setting box is set to delay the email being automatically sent (in Minutes). This must be set for greater than 5 minutes. This would give you a preset time period that would give the opportunity to verify an issue, and make sure it is not a false positive or a false alarm, before an email is sent. The level is set based upon the time needed to verify the issue.

**NOTE:** These changes made are global and immediate!!!

The screenshot shows the 'Admin' section of the BEET enVision interface. The 'Other Settings' page is active, displaying a warning message: 'Stop! Any change made are global and immediate!!!'. Below this, there are two main settings:

- Subscription email delay:** A text input field containing the number '5'. To its right, a note states '(Must be greater than 5 Minutes.)'. A callout bubble explains: 'In the Envision Subscription service, an notification email would be sent in the event of a Fault, Warning, or other problems, depending on how the program is configured initially.'
- Show Default Dashboard:** A checkbox that is currently checked. A callout bubble explains: 'Checking off this box will cause the the selected default dashboard to load upon opening the Dashboard.'

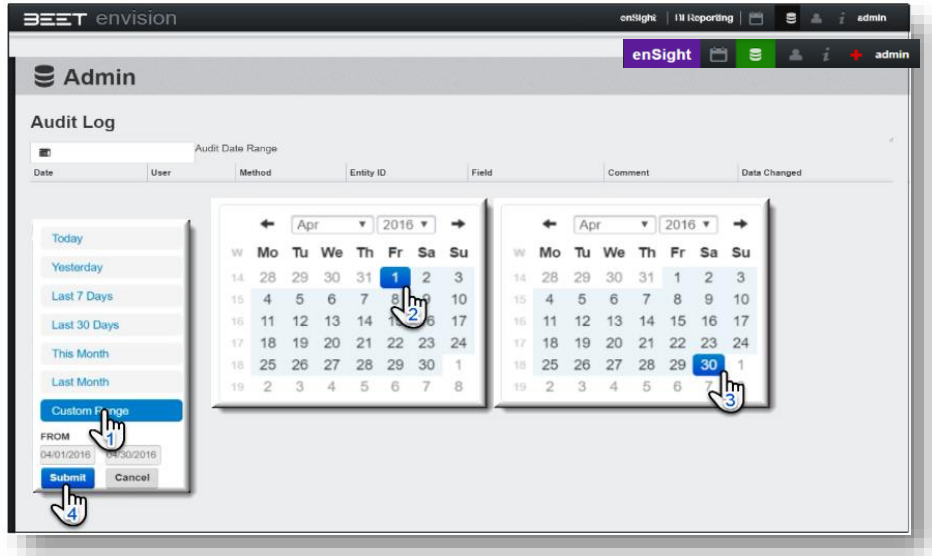
A 'Save' button is located to the right of the 'Subscription email delay' field. Another callout bubble explains: 'This option will allow you to delay sending a subscribed email. This may allow time to verify the issue and ensure that it is a real problem and not a false positive that would not require an email.'

# AUDIT

## Overview

The Audit Log contains all the changes made from the actual user that is logged in. This log will show the date of the action performed, User, Method, Entity ID, Field, Comments and the actual Data changed.

To use this feature, simply click on the Audit Date Range Icon to open a date range selection window. Choose from Today all the way to a custom range. In custom range, you select the start date and the end date, then click on Submit.



## Audit Date

When an action is performed in the Admin Module, that requires a Save or a change on a screen that makes instant changes, enVision records the event Date and actual time the action was taken.

## User

The user who logs into the enVision program

will have their User ID added to the record. For example, Thomas Katt (username TomKat) will show up as TomKat under user.

## Method

The Method is the way that the action was performed. By use of an action button (Save), command, process, update, or removal.

## Entity ID

The entity ID is a label assigned to the action that was performed?

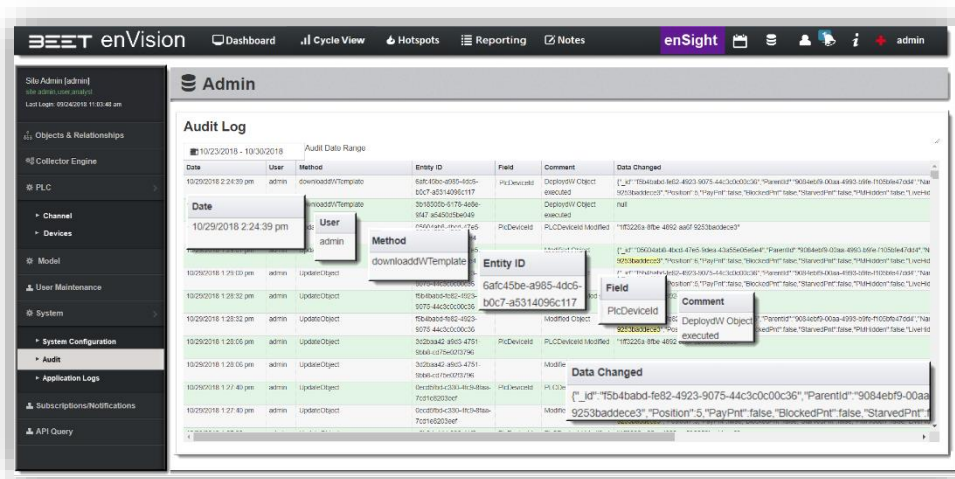
## Comments

Comments are inputted by the User when the action was taken.

## Data Changed

The data changed section shows what the original data (~~Original Data in Pink with STRIKETHROUGH~~) was before it was changed, then shows the new data (New Data Highlighted in Yellow) as well using a color code and format to show the changes. If the result of the changes was wrong, it preserves that original data to revert back to if needed. Click on any of the highlighted data in the Data Changed column and a window of the original and modified data will appear, as seen below.

In this example, you can click any of the **highlighted** area in the **Data Changed** column to open an Audit Data Details window - null {"\_id": "bdb595cb-0689-491c-a016-221a8d8bba52", "ParentId": "f4270b7



Click on the **Highlighted** portion in the **Data Changed** column to open a window of the **Original** and **Modified** changes to the Data.

The Audit Data Details window contains all the data from the previous setting. When new data is entered, it highlights the previous data in pink with a **strikethrough** and also shows the **new data in yellow highlight**. If a situation occurs where you need to revert back to the original configuration, this data would be available.

```

Audit Data Details
null{
  _id: "ca69e327-834e-44b8-a2f6-0124e0289996",
  ParentId: "655fb735-89ed-4d92-ae43-841151a283d8",
  Name: "IRN Ara 2",
  Description: "",
  Type: {
    _id: "2",
    Name: "Area"
  },
  PlcDeviceId: null,
  Position: 2,
  PayPnt: false,
  BlockedPnt: false,
  StarvedPnt: false,
  PMHidden: false,
  LiveHidden: false,
  Cls: null
}
    
```

# APPLICATION LOGS

[Application Logs](#) | [Overview](#) | [Log History Selector](#) | [Descriptions](#)

## Overview

In the Application Logs section, it keeps a record of certain actions that are happening to enVision. It records a time and date stamp for when certain actions happen in enVision. It also records when a user logs into enVision. As well if enVision is experiencing an error, fault, or something did not work correctly as designed, it will create a record of what happen.

Application Log View

Log History

Drag a column header here to group by that column

Level	Date	Logger	Message	User	Method
Info	06/12/2018 7:55:42 am	EnsigntLib.MessageRepo	Message Repo Initialized	admin	EnsigntLib.MessageRepo..ctor
Info	06/12/2018 7:55:42 am	EnsigntLib.MessageRepo	Message Repo Initialized	admin	EnsigntLib.MessageRepo..ctor
Info	06/12/2018 7:55:41 am	EnsigntLib.MessageRepo	Message Repo Initialized	admin	EnsigntLib.MessageRepo..ctor
Info	06/12/2018 7:55:41 am	EnsigntLib.MessageRepo	Message Repo Initialized	admin	EnsigntLib.MessageRepo..ctor
Info	06/12/2018 7:55:40 am	EnsigntLib.MessageRepo	Message Repo Initialized	admin	EnsigntLib.MessageRepo..ctor
Info	06/12/2018 7:54:42 am	EnsigntLib.MessageRepo	Message Repo Initialized	admin	EnsigntLib.MessageRepo..ctor
Info	06/12/2018 7:54:42 am	EnsigntLib.MessageRepo	Message Repo Initialized	admin	EnsigntLib.MessageRepo..ctor
Info	06/12/2018 7:54:41 am	EnsigntLib.MessageRepo	Message Repo Initialized	admin	EnsigntLib.MessageRepo..ctor
Info	06/12/2018 7:54:41 am	EnsigntLib.MessageRepo	Message Repo Initialized	admin	EnsigntLib.MessageRepo..ctor
Info	06/12/2018 7:54:40 am	EnsigntLib.MessageRepo	Message Repo Initialized	admin	EnsigntLib.MessageRepo..ctor
Info	06/12/2018 7:53:43 am	EnsigntLib.MessageRepo	Message Repo Initialized	admin	EnsigntLib.MessageRepo..ctor
Info	06/12/2018 7:53:43 am	EnsigntLib.MessageRepo	Message Repo Initialized	admin	EnsigntLib.MessageRepo..ctor

Level	Date	Logger	Message	User	Method
Info	06/12/2018 7:55:42 am	EnsigntLib.MessageRepo	Message Repo Initialized	admin	EnsigntLib.MessageRepo..ctor

## Log History Selector

Application Log View

Log History

When the Application Log is selected, it is defaulted to the last (1) day. To view back several days, simply click in the Log History edit box and enter the number of days you would like to look back to.

**Note:** A single day may have a large number of logs. Selecting multiple days may take a longer time period to load and display.

## Descriptions

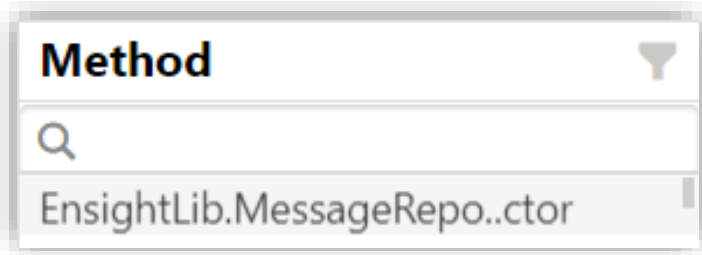
The records it creates contains the level (information, error, etc.), current date, time, Logger, **Message**, **User**, and **Method**. The **Level** column shows the kind of record it is representing. **Info** would show that there was a log on, deploy, or similar actions performed. **Error** would show that an inaction took place or an incorrect result occurred.

**Message**

Message Repo Initialized

The **Message** column contains the message recorded by the part of enVision that the Info or Error had occurred in. (**Message Repo Initialized**). The CE (Collector Engine and Address) could not connect or get access to the enVision Live Data.

The **Method** column displays ( **EnsignLib.**)what part of enVision that experienced the condition, and the action or inaction that was taking place. In the case of an inaction or something that was supposed to happen that didn't, it would see it that an error occurred, and will be listed as such. When logging onto or to different parts of enVision, it will show it as an action in the program and leave a record of an application engaging.



The **Level** column shows the kind of record it is representing. **Info** would show that there was a log on, deploy, or similar actions performed. **Error** would show that an inaction took place or an incorrect result occurred.

The **Message** column contains the message recorded by the part of Envision that the Info or Error action had occurred in.

Level	Date	Logger	Message	User	Method
Info	06/12/2018 7:55:42 am	EnsignLib.MessageRepo	Message Repo Initialized	admin	EnsignLib.MessageRepo..ctor

Date and Time stamp of the record.

The **Logger** displays which part of the program is recording to the log.

The **Method** column displays what part of Envision that experienced the condition, and the action or inaction that was taking place.

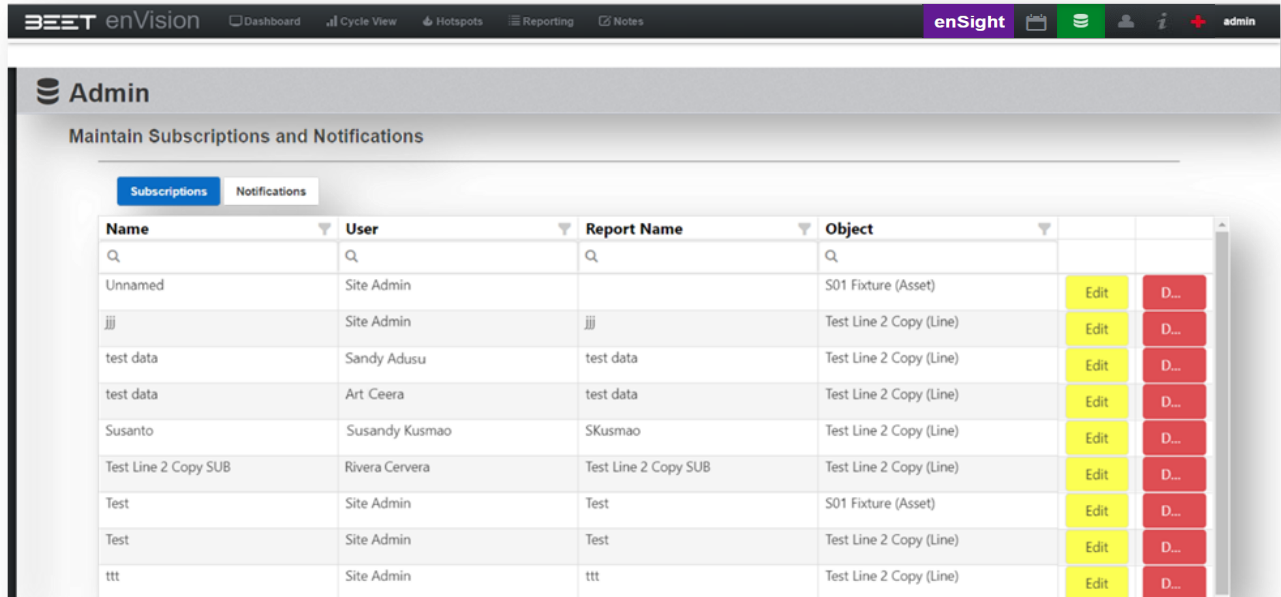


# Subscriptions and Notifications

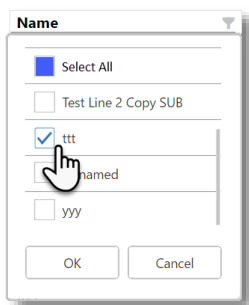
The Subscriptions and Notifications sections is an area of the Admin module where the Subscriptions and Notifications can be monitored and maintained. From here you can Edit or Delete them. Selecting the tab will open the Subscriptions screen (default).

## Subscriptions

In the Subscriptions window, the subscription button will be highlighted in blue. You can easily go to Notifications by selecting the button to the right of it. Below the button is a table of previously configured subscriptions.



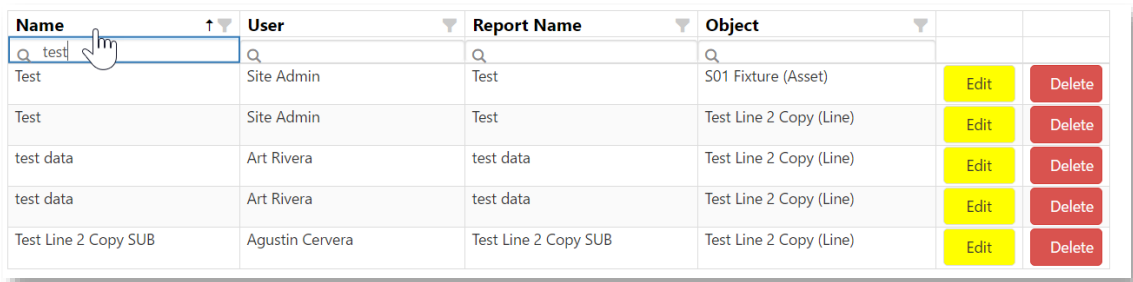
## Filter Bar



You can filter each column by selecting the filter icon, atop of each column. This action will open a window that will show a check off list that will include the items of each column and a Select All box. Either select check all or select the individual items you desire. When finished, select the OK button to close and update the table list.

## Search Bar

You can also sort by using the search dialog. Simply start typing the name of the item you seek and a list with the similar item name will appear below. From there, the items of the list can be Edited or Deleted.



# Notifications

The functions of the Notifications Table are the same as the Subscriptions table.

User	Object	Measure	Range		
Rustin Cerver	R01	overcycle,	shift	Edit	D...
Agust Fierra	S02 Fixture	overcycle, faulted,	hour	Edit	D...
Cera Carver	R03	overcycle,	day	Edit	D...
Site Admin	S01 Fixture	overcycle,	hour	Edit	D...
Site Admin	S01 Fixture	outofauto,	hour	Edit	D...

## Edit

Selecting the Edit button will open the Configure Subscription window. From here, changes can be made to the selected subscription. Once everything is edited, select the Save button to save the newly edited subscription or select Close to go back to the table list.

The 'Configure Subscription' window includes the following sections:

- Name\***: Text input field.
- Range\***: Dropdown menu with 'Current Shift' selected.
- Select Object\***: Tree view showing a hierarchy of objects including Simulations, Test Area, Test Line 2 Copy, Sta 1, S01 Fixture (selected), and various equipment like CYCLE TIME, TRANSFER IN, ROBOT LOAD, etc.
- Status (Optional)**: Dropdown menu with options: Good, Watch, Warning, Fault, Missing.
- Shift (Optional)**: Text input field.
- Shift Crew (Optional)**: Text input field.
- Classification (Optional)**: Text input field.
- Recipients**: List of recipients including Email Subject and Email Message.
- Calendar**: A weekly calendar grid with a green block at 2:00 PM on Wednesday.

See [Subscriptions and Notifications](#) in the End User manuals or at the Beet Knowledge Base.

The 'Conditional Notification' window for 'S01 Fixture' includes the following configuration options:

- Name**: Text input field.
- Type**: Radio buttons for 'Accumulation' (selected) and 'Single Event'.
- Range**: Radio buttons for 'Hour' (selected), 'Shift', 'Day', '7 Days', and '30 Days'.
- Condition**: Radio buttons for 'And' (selected) and 'Or'.
- Table**:
 

<input type="checkbox"/>	Measure	Accumulated Duration (Min.)	Occurrences
<input type="checkbox"/>	Over Cycle		
<input type="checkbox"/>	Faulted		

# API Query

The screenshot shows the BEET enVision Admin interface. At the top, there are navigation links for 'enVision', 'Dashboard', and 'Cycle View'. Below this is a grey header with the 'Admin' logo. The main section is titled 'API Generate' and includes a 'Cycle Data' sub-section. Underneath, there is an 'Assets' section with a scrollable list of items: 'FIXT B LACE OPER LOAD', 'FIXT B LACE WORK', 'FIXT LACE TT', 'Fixture', 'FIXTURE LASER CUT', 'FIXTURE LOAD - UNLOAD', 'Fixture2', 'GH1 ROBOT', and 'GH2 ROBOT'. Below the list is a 'Datetime Range' section with two radio buttons: 'Current Shift' and 'Last Shift'. A green 'Generate' button is located below the form. At the bottom, a URL is displayed: `http://development.beet.com/api/CycleHistory/GetCycles?Assets=35968f4e-6f1c-4e66-8d7d-2eab8cb847f6,c16e5c3d-e33d-4b1d-a5bd-eaba296d6711,b45f537c-1700-48c9-83be-7e2591f04beb,12b2e88d-4c5b-4a3d-85ad-45f8dac3e`

This screenshot is identical to the one above but includes four blue callout boxes with white text and arrows pointing to specific UI elements:

- Top Callout:** "Select the ASSETS that you would like added to the Generated Data File." (Points to the Assets list)
- Middle Callout:** "Make a selection from the Current Shift or the Last Shift" (Points to the Datetime Range radio buttons)
- Bottom Callout:** "Click the Generate button to create link to a API file address." (Points to the Generate button)
- Right Callout:** "The space below the Generate an address for the Data file." (Points to the URL area)

The URL at the bottom is: `http://development.beet.com/api/CycleHistory/GetCycles?Assets=35968f4e-6f1c-4e66-8d7d-2eab8cb847f6,c16e5c3d-e33d-4b1d-a5bd-eaba296d6711,b45f537c-1700-48c9-83be-7e2591f04beb,12b2e88d-4c5b-4a3d-85ad-45f8dac3e`

# Glossary

- **AOI**  
Add On Instructions
- **CE**  
Collector Engine
- **COS**  
Change of State
- **EAS**  
enVision Application Server
- **EDC**  
enVision Data Collector
- **xls**  
Excel spreadsheet
- **FIS**  
Factory Information System
- **OP**  
Operation
- **OPC**  
OLE Process Control
- **OEE**  
Overall Equipment Efficiency
- **PB**  
Push Button
- **PLC**  
Programmable Logic Controller
- **SSL**  
Secure Socket Layer
- **SMTP**  
Simple Mail Transfer Protocol
- **UID**  
Unique Identifier
- **VPS**  
Virtual Private Server

# Product Support

## Standard Support

Every licensed enVision user is entitled to: 60 days of free email based product support, [support@beet.com](mailto:support@beet.com)

Unlimited access to the on-line support materials available at Beet Analytics Technology support website  
<http://support.beet.com>

Critical bug fix updates for the version of software purchased.

## Service Maintenance Level Support

### Submitting Suggestions and Reporting Issues

Every licensed user who holds an active service maintenance contract for enVision is entitled to: Unlimited email based support for the duration of the subscription license [support@beet.com](mailto:support@beet.com)

Unlimited access to the on-line support materials available at Beet Analytics Technology support website  
<http://support.beet.com>

Critical bug fix updates for the version of software purchased

Product enhancement updates for the duration of the service maintenance contract

If you wish to make a suggestion or report an issue you have discovered using **enVision**, you can do so using our support website <http://support.beet.com>. Alternatively, you can email Beet Analytics Technology Support at [support@beet.com](mailto:support@beet.com)

The amount of information you can provide us with about the nature of a problem you are having will directly affect our ability to resolve it. The more information you can provide about your environment, the steps to reproduce and any other relevant information the better – *please be verbose!*

# Trouble Shooting

See the Product Support section for available content at Beet Analytics Technology support website:  
<http://support.beet.com>

For more information or questions, go to the enVision Customer Knowledge Base website:  
<https://docs.beet.com/display/EKB/enVision+Customer+Knowledge+Base>

See Troubleshooting articles on the enVision Customer Knowledge Base website:  
<https://docs.beet.com/display/EKB/Troubleshooting+articles>

# License and Copyright

## License and Copyright

### License Agreement

IMPORTANT! Do not install this SOFTWARE before you have read this license agreement. By proceeding to install this SOFTWARE you are indicating your acceptance of all the terms and conditions stated in this agreement.

#### 1. License Agreement

1.1 License Grant Beet Analytics Technology grants Licensee a non-exclusive, non-transferable (except as expressly permitted herein) license, solely for its internal business operations, to: a) Make and install the necessary number of copies of the applicable Licensed Software; b) Use the executable form of the software in the United State for the purpose of processing Licensee's own internal; c) Allow its authorized users to access and use the Licensed Software; d) Make one copy of each Licensed Software for back-up purposes.

This license is effective upon the Effective Date of the License and is subject to all the terms of this EULA. Licensee agrees to operate the Licensed Software in accordance with the terms and provisions of this EULA, the applicable Licensed Software terms in the Quote and to ensure that its Users comply with these terms. License keys or license tokens do not themselves grant the legal right to use the Licensed Software. Certain Licensed Software may either contain third party software components or may be third party software products to which certain Specific Terms for Third Party Software apply. Except as is expressly set forth in this EULA, no other express or implied right or license is granted to Licensee.

1.2 License Type Definitions the following License Types may be offered with respect to individual Software products or product families. The License Type will be specified in a Quote or similar ordering document provided by Beet Analytics Technology. (a) "Concurrent User" licenses means that access to the Software at any given moment will be limited to the maximum number of concurrent users for whom licenses have been validly acquired under this Agreement. (b) "Named User" licenses means that access to the Software will be restricted to those individuals within Licensee's organization that are named by Licensee and for whom licenses have been validly acquired under this Agreement. Licensee shall have the right to change Named User licenses provided that no individual Named User license may be changed more than once every thirty (30) days. Licensee agrees not to: (a) use the Licensed Software to develop software applications for use by or distribution to any third party, whether in whole or part.

1.3 License Limitations Licensee agrees not to: (a) use the Licensed Software to develop software applications for use by or distribution to any third party, whether in whole or part, whether as standalone products or as components; (b) rent, lease, sublicense, perform or offer any type of services to third parties relating to the Licensed Software, including but not limited to, consulting, training, assistance, outsourcing, service bureau, customization or development unless specifically authorized in the applicable Licensed Software terms; (c) correct errors, defects and other operating anomalies of the Licensed Software; (d) modify (except as permitted and described in the Documentation), adapt, reverse engineer, decompile, disassemble, or otherwise translate all or part of the Licensed Software; (e) provide, disclose or transmit any results of tests or benchmarks related to any Licensed Software; (f) use any software that may be delivered with the Licensed Software other than the Licensed Software ordered hereunder.

#### 2. Software Maintenance Services

2.1 Software Maintenance. Software maintenance services consist of (a) the provision of Software updates, (b) the provision of Error corrections, as defined herein, for the Software, and (c) the provision of telephone support in connection with the Software. Software maintenance services will be provided in accordance with the terms of this Schedule to those customers who have purchased maintenance services under this Agreement for the applicable Software. Software maintenance services are, and will continue to be, available under this Agreement only to the extent that these services are made available by Beet Analytics Technology with respect to the Software, or any portion of the Software, to its customer base in general.

2.2 Maintenance Term. Licensee may purchase Software maintenance services for an initial annual maintenance term or such other time period that is acceptable to Beet Analytics Technology. Thereafter, maintenance services will automatically renew for successive one (1) year terms unless terminated by Licensee by providing notice of termination in writing to Beet Analytics Technology at least 60 days prior to the expiration of the initial term or any renewal term. Beet Analytics Technology will provide Licensee a written maintenance quotation for each subsequent maintenance term 90 days in advance of the expiration of the then current maintenance term. If Licensee purchases additional Software licenses during the term of this Agreement, Beet Analytics Technology reserves the right to adjust the annual maintenance term and pro rate the annual maintenance fees for such add-on Software to be coterminous with the initial Software maintenance term and billing cycle.

2.3 New Releases of Software. New versions of the Software released by Beet Analytics Technology may contain Error corrections and/or new or enhanced functionality. A new version may be either a point release denoted by a change to the right of the first decimal point (e.g. V2.0 to V2.1) (a "Point Release") or a major release denoted by a change to the left of the first decimal point (e.g. V2.0 to V3.0) (a "Major Release"). A Point Release will generally consist of corrections to known Errors. A Major Release will generally consist of a new version of the Software that contains new or enhanced functionality. Licensee shall have the right to receive new Point Releases and new Major Releases of the Software that are released to Beet Analytics Technology's customers in general during any period of time for which Licensee has purchased maintenance services under this Agreement for the applicable Software. This right does not extend to any release, module, option, future product, or any upgrade in functionality or performance of the Software which Beet Analytics Technology develops as a customized product for a single customer or that Beet Analytics Technology develops and licenses as a separate product and not for release to customers in general as part of maintenance services. Licensee is responsible for the installation and implementation of any new version and any required data conversion. Licensee remains solely responsible for the configuration of its own equipment and software, including the compatibility of any additional equipment or software with the Beet Analytics Technology Software.

2.4 Support for Prior Versions of the Software. Once a new version of the Software is released, either a Point Release or a Major Release, Beet Analytics Technology will maintain the current version it just released and the most current Point Release that relates to the immediately preceding Major Release. For example, if V2.1 is released, Beet Analytics Technology will maintain V2.1 and V1.x, where x is the latest Point Release in the V1 series. If a known Error has been corrected in an update to the prior Major Release, Beet Analytics Technology retains the right to require the Licensee to upgrade to the requisite Point Release that contains the Error correction rather than providing a separate patch or workaround.

2.5 Error Corrections. An Error means the failure of the Software to conform substantially to the Documentation ("Error"). Licensee may report any suspected Error to Beet Analytics Technology and, upon Beet Analytics Technology's request, Licensee will provide Beet Analytics Technology with a detailed, written description and documentation of the suspected Error. Beet Analytics Technology will investigate the facts and circumstances related thereto and Licensee will cooperate with Beet Analytics Technology's investigation. If Beet Analytics Technology finds that the Software contains an Error, Beet Analytics Technology will use all commercially reasonable efforts to correct the Error. An Error correction may consist of a separate patch, a workaround or it may be included in the next available Point Release or Major Release of the Software, at the discretion of Beet Analytics Technology. If Beet Analytics Technology cannot provide an Error correction as defined above, after being notified of the Error by Licensee, Licensee may terminate the software maintenance and receive a refund under Section 7 below.

2.6 Telephone Support. Licensee shall have the right to receive telephone support in connection with the Software by calling the toll-free number provided by Beet Analytics Technology between the hours of 8:00 am and 5:00 pm, in local time zones of the contiguous United States, Monday through Friday, except on holidays recognized by Beet Analytics Technology. Licensee will also be provided with the ability, by means of an electronic channel via the Internet, to log Software support requests, report suspected Errors, monitor progress on the Licensee's prior requests, download Software fixes and workarounds, exchange information on a bulletin board, and obtain access to release notes and other Software information.

2.7 Limitation of Remedies. Beet Analytics Technology's sole and exclusive responsibility, and Licensee's sole and exclusive remedy, for a failure to provide error corrections for the Software in accordance with this Schedule will be that Licensee may terminate Software maintenance for the Software involved. Beet Analytics Technology will thereafter promptly refund the unused portion of the fees paid for the remainder of the then current term of the Software maintenance services for the applicable Software.

2.8 Initial and Renewal Fees. Purchase of Software maintenance services for a Licensee site will be subject to the purchase of such services for all Beet Analytics Technology supported Software modules licensed for use at that site. The fees for Software maintenance services will be set forth on Beet Analytics Technology's quotation for such services. For each of the first two renewal terms following the initial term, Beet Analytics Technology may adjust the Software maintenance fees provided that: (i) Beet Analytics Technology provides written notice to Licensee at least 60 days prior to the end of the then current term, (ii) the fee adjustment becomes effective upon expiration of the then current term, and (iii) the percentage of the fee adjustment does not exceed three percent (4%) per year. Following the first two renewal terms, Beet Analytics Technology's ability to increase maintenance prices shall not be subject to the cap or notice period requirements Licensed Software Maintenance Services automatically renew on an annual basis unless terminated in accordance with the provisions of Section 7.4. The obligation to provide Software Maintenance Services shall be undertaken by Beet Analytics Technology or a Beet Analytics Technology authorized service provider.

### 3. Orders, Delivery and Payment

3.1 Orders. Licensee may order from Beet Analytics Technology Licensed Software and related Software Maintenance Services under this EULA, by issuing an Ordering Document. Licensee agrees that the terms and conditions of this EULA shall apply to all such Ordering Documents, Licensed Software and Software Maintenance Services ordered thereunder.

3.2 Payment Terms: a) In consideration of the rights, licenses and services provided hereunder, Licensee shall pay Beet Analytics Technology the charges applicable to each license of a Licensed Software at the price identified in the applicable Quote. Unless otherwise agreed to in writing by Beet Analytics Technology, (i) all charges will be invoiced upfront and (ii) Licensee shall pay all invoices by within thirty (30) days from invoice date. b) Late Payments. Licensee shall pay interest for late payment at a rate of 1.5% per month or highest lawful rate on all sums unpaid at the due date, plus reasonable attorneys' fees and costs incurred by Beet Analytics Technology in collecting unpaid amounts. c) Taxes. All prices are exclusive of taxes. Licensee shall be responsible for payment of any and all taxes, duties, excises, import VAT or similar charges of any nature whatsoever, now in force or enacted in the future, that are levied, assessed, charged, withheld, or collected for or in connection with Licensed Software and/or in relation to the transfer or usage provided hereunder or otherwise arising in connection with this EULA, but excluding taxes based on Beet Analytics Technology net income. If Licensee is or may be required under any law or regulation of any governmental entity or authority, domestic or foreign, to withhold or deduct any portion of any payment due to Beet Analytics Technology pursuant to this EULA, then the sum payable to Beet Analytics Technology will be increased by the amount necessary to yield to Beet Analytics Technology an amount equal to the sum it would have received had no withholdings or deductions been made. Licensee shall indemnify Beet Analytics Technology against any losses or costs incurred by Beet Analytics Technology due to any failure of Licensee to make such deduction or withholding.

3.3 Delivery. Beet Analytics Technology will deliver to Licensee the Licensed Software ordered hereunder, or make the Licensed Software available electronically. Electronic delivery will be made by posting the Licensed Software ordered by Licensee on Beet Analytics Technology's site and providing Licensee with a user name, password, and instructions for accessing and downloading the Licensed Software from such site. Licensee is responsible for accessing Beet Analytics Technology's site and downloading the Licensed Software. Licensed Software ordered by Licensee from Beet Analytics Technology shall be delivered FCA (Incoterms 2000) at Beet Analytics Technology's premises identified in Beet Analytics Technology's Quote.

### 4. Intellectual Property

4.1 Ownership: Beet Analytics Technology and/or its suppliers retain ownership in all intellectual property rights in the Licensed Software and all modifications, enhancements or other derivative works thereof. The Licensed Software are licensed, not sold. Licensee shall preserve and reproduce all copyright, patent and trademark notices which appear in the Licensed Software on all partial or integral copies thereof. Licensee recognizes that the methodologies and techniques,

contained in or expressed within the Licensed Software are proprietary information or trade secrets of Beet Analytics Technology or its suppliers. Licensee shall treat them as confidential information and not disclose them. 4.2 Intellectual Property Indemnification: Unless provided otherwise in the Specific Terms for Third Party Software, Beet Analytics Technology will defend Licensee against any claims made by a third party that a Licensed Software delivered under this EULA infringes a copyright in any country or a patent of the United States, and shall pay all costs, damages and expenses (including reasonable legal fees) finally awarded against Licensee by a court of competent jurisdiction or agreed to in a written settlement agreement signed by Beet Analytics Technology arising out of such claim, provided that (i) Licensee provides Beet Analytics Technology with prompt written notice of the claim, (ii) Licensee gives Beet Analytics Technology sole control of the defense of the claim and any related settlement discussions and provides reasonable cooperation in the defense and settlement of the claim, and (iii) in the case of a patent infringement, the related patent has been granted as of the date of Licensed Software delivery to Licensee. If a claim is made or in Beet Analytics Technology's reasonable opinion, is likely to become the subject of such a claim, Beet Analytics Technology may at Beet Analytics Technology's expense, either secure the right for Licensee to continue using the Licensed Software or modify it so that it is not infringing, or replace it with another program which is functionally equivalent. If none of the foregoing options is available on terms which are reasonable in Beet Analytics Technology's judgment, Beet Analytics Technology may terminate the licenses to the Licensed Software and, if such licenses are perpetual licenses, Beet Analytics Technology will either refund or provide a credit to Licensee, at Licensee's option, in an amount equal to the corresponding one-time fee paid for the licenses, depreciated on a straight-line over three (3) years upon return or destruction of all copies of the affected Licensed Software as certified by an officer of Licensee. Beet Analytics Technology shall have no obligation to defend or indemnify Licensee against any claim related to (i) any modification of a Licensed Software by anybody other than Beet Analytics Technology, or (ii) the use of one or more Licensed Software in combination with other hardware, data or programs not specified by Beet Analytics Technology, or (iii) the use of corrective patches or Releases other than the most recent one provided by Beet Analytics Technology. This Section 4.2 states Beet Analytics Technology's entire liability and Licensee's exclusive remedy for any claim of infringement of intellectual property rights under this EULA.

## 5. Warranties, Limitations and Disclaimers

5.1 Warranty. Beet Analytics Technology warrants for ninety (90) days from delivery to Licensee that the Release of any Licensed Software will materially conform to its Documentation, provided that it is properly used in the operating environment specified by Beet Analytics Technology. If such Release of the Licensed Software does not conform, Beet Analytics Technology will attempt to make the Licensed Software perform as warranted. Beet Analytics Technology may request Licensee to install a corrective patch or a new Release. If, after sixty (60) days from notice by Licensee of the non-conformity received within the warranty period as defined above, Beet Analytics Technology has not provided a conforming Licensed Software, Licensee's exclusive remedy and Beet Analytics Technology's entire liability for any breach of such warranty is for Licensee to terminate the license related to the non-conforming Licensed Software within thirty (30) days after such sixty (60) day period and obtain a refund of fees paid for such Licensed Software hereunder.

5.2 Disclaimers. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, THE FOREGOING WARRANTIES ARE IN LIEU OF AND EXCLUDE ALL OTHER WARRANTIES, REPRESENTATIONS OR CONDITIONS FOR THE LICENSED SOFTWARE OR SERVICES, WHETHER EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE OR NON INFRINGEMENT. Beet Analytics Technology disclaims any liability for any use or application of any Licensed Software or the results or decisions made or obtained by users of the Licensed Software. Except to the extent required by applicable law, Beet Analytics Technology does not warrant that (i) the functions of Licensed Software will meet Licensee's requirements or will enable it to attain the objectives Licensee has set for itself, or (ii) they will operate in the combination or environment selected for use by Licensee, or (iii) the operation of the Licensed Software will be uninterrupted or free of errors. In all instances, Licensee shall be solely responsible for ensuring that the results produced by Licensed Software comply with quality and safety requirements of Licensee's products or services. No employee or agent of Beet Analytics Technology is authorized to give a greater or different warranty. Licensee shall have exclusive responsibility for (a) program selection to achieve Licensee's intended results, (b) Licensed Software installation, (c) taking adequate measures to properly test, operate and use each Licensed Software and (d) results obtained there from.

## 6. Limitation of Liability

Each party is independently and exclusively responsible for obligations undertaken by it under this EULA. No party can be held jointly and severally liable with another pursuant to this EULA. No party shall be deemed an agent of another party pursuant to this EULA.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, UNLESS OTHERWISE SPECIFIED IN ANY APPLICABLE SPECIFIC TERMS FOR THIRD PARTY SOFTWARE, BEET ANALYTICS TECHNOLOGY'S POTENTIAL LIABILITY TO LICENSEE, FOR ANY AND ALL CLAIMS IN ANYWAY ARISING FROM OR IN CONNECTION WITH THE SUBJECT MATTER OF THIS EULA, WHETHER BASED IN CONTRACT, OR OTHER THEORY OF LIABILITY, IS LIMITED AS FOLLOWS:

EXCEPT FOR BEET ANALYTICS TECHNOLOGY'S LIABILITY UNDER SECTION 4.2 HEREOF, BEET ANALYTICS TECHNOLOGY'S AGGREGATE LIABILITY FOR DIRECT DAMAGES SHALL NOT EXCEED IN THE AGGREGATE THE AMOUNT CORRESPONDING TO FEES ACTUALLY PAID BY THE CONCERNED LICENSEE IN THE PRECEDING TWELVE (12) MONTH PERIOD PRIOR TO THE OCCURRENCE OF THE CAUSE OF ACTION GIVING RISE TO THE CLAIM FOR THE USE OF THE LICENSED SOFTWARE WHICH CAUSED THE DAMAGES. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, LICENSEE EXPRESSLY AND IRREVOCABLY WAIVES, AND BEET ANALYTICS TECHNOLOGY SHALL HAVE NO LIABILITY IN RESPECT OF, ANY AND ALL CLAIMS FOR INDIRECT, INCIDENTAL, CONSEQUENTIAL OR PUNITIVE DAMAGES, INCLUDING WITHOUT LIMITATION CLAIMS FOR LOST PROFITS, BUSINESS INTERRUPTION AND LOSS OF DATA, THAT IN ANY WAY RELATE TO THIS EULA, LICENSED SOFTWARE, DOCUMENTATION OR SERVICES, WHETHER OR NOT BEET ANALYTICS TECHNOLOGY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES AND NOTWITHSTANDING THE FAILURE OF THE ESSENTIAL PURPOSE OF ANY REMEDY. LICENSEE WAIVES ANY AND ALL CLAIMS FOR ANY DIRECT, INDIRECT, INCIDENTAL OR CONSEQUENTIAL DAMAGES, FOR ANY REASON AND ON ANY BASIS, AGAINST ANY BEET ANALYTICS TECHNOLOGY PROVIDERS, OR.

All legal actions against Beet Analytics Technology must be filed with the appropriate judicial jurisdiction within two (2) years after the cause of action has arisen.

## 7. Term and Termination

7.1 Term: This EULA shall come into force on the Effective Date and shall remain in full force and effect for a period of three (3) years from its Effective Date, unless terminated earlier as provided hereunder. Each Affiliate Participation Agreement shall expire concurrently with the EULA, unless terminated earlier in accordance with the provisions herein.



7.2 Termination of the EULA a) Material Breach. Either Beet Analytics Technology or Licensee may terminate this EULA and/or any licenses granted under this EULA, if the other is in material breach of any of its obligations and has failed to remedy such breach within thirty (30) days of receipt of written notice. b) For Convenience. Licensee may terminate any license to any Licensed Software by providing written notice to Beet Analytics Technology. Such notice may be provided at any time for perpetual licenses. Notices for term licenses shall be provided sixty (60) days prior to the Anniversary Date of the License. Licensee shall not be entitled to any refund or credit for termination of a license for any reason.

7.3 Effect of Termination of the EULA a) Effect of Termination of the EULA for Non-Payment. If this EULA is terminated by Beet Analytics Technology for breach by Licensee of its payment obligations, termination of this EULA shall also result in the termination of (i) Licensee's rights to use any licenses for which payment has not been made; (ii) the right of Licensee and (iii) Licensee's right to submit any further Ordering Documents under this EULA. All licenses for which Licensee is not in breach of its obligations shall remain in full force and effect. b) Effect of Termination of the EULA for Other Breach. If this EULA is terminated by Beet Analytics Technology for material breach by Licensee of any other obligations hereunder, termination of this EULA shall also result in the termination of Licensee's rights to use all Licensed Software ordered hereunder.

7.4 Effect on Licensed Software of Termination or Expiration of the EULA and/or Affiliate Participation Agreement Upon expiration or termination of this EULA and/or any Affiliate Participation Agreement or any licenses granted hereunder as a result of Licensee's uncured material breach or by Licensee for convenience, Licensee shall immediately destroy or return all copies of the terminated or expired Licensed Software and associated Documentation in their entirety, and duly certify the same in writing to Beet Analytics Technology. Expiration or termination of this EULA and/or any Affiliate Participation Agreement or any license shall not relieve Licensee of its obligation to pay all fees that have accrued or are otherwise owed by Licensee under any Ordering Document. The expiration or termination will not prejudice the rights and remedies of the non-breaching parties.

7.5 Termination of Software Maintenance Services a) By Licensee: Licensee may terminate Software Maintenance Services for a Licensed Software subject to the following conditions: (i) Licensee notifies Beet Analytics Technology with at least sixty (60) days prior notice, and (ii) such termination shall apply to Software Maintenance Services related to all licenses of said Licensed Software held by Licensee under any license agreement then in force between Licensee and Beet Analytics. In such case for all such licenses described in the preceding sentence: (x) Licensee shall have no further obligation to pay the Software Maintenance Services fees related to the corresponding Licensed Software; (y) Licensee shall duly certify in writing to Beet Analytics Technology that all copies of all Releases of the Licensed Software other than those of the latest Release of the Licensed Software installed by Licensee, have been duly destroyed or returned to Beet Analytics Technology in their entirety; and (z) Software Maintenance Services for such Licensed Software will terminate at the expiration of the thirty (30) day notice period. Beet Analytics Technology shall have no further obligation to provide any services or deliver any Release in support of any such licenses, except for providing license keys if necessary. Licensee may reinstate Software Maintenance Services, provided such reinstatement is activated for all licenses of a given Licensed Software held by Licensee under any license agreement then in force between Licensee and Beet Analytics Technology or any other Beet Analytics Technology, and Licensee pays all fees that would have been due in respect of Software Maintenance Services from the date of termination of Software Maintenance Services to the date of reinstatement of such Software Maintenance Services plus a reinstatement fee???. b) By Beet Analytics Technology: Beet Analytics Technology may terminate Software Maintenance Services related to all Licensed Software under this EULA if Licensee fails to pay Software Maintenance Services fees when due, upon thirty (30) days prior written notice. Fees related to Software Maintenance Services shall be due by Licensee until termination of such Software Maintenance Services.

## 8. Miscellaneous

8.1 Purchase Orders. Licensee's purchasing terms and conditions shall not in any way supersede, modify, vary or otherwise supplement the terms of this EULA.

8.2 Notices. All notices required hereunder shall be in writing, in English and shall be deemed to have been given (i) the date delivered in person or by reputable express courier service, (ii) three (3) days after sending the notice if sent by certified or registered mail, (iii) the date sent by confirmed facsimile, addressed to the parties at their addresses in the Ordering Documents, or at such other address as either party may designate to the other by notice served as hereby required, or contained in the relevant order form.

8.3 Force majeure. Neither Party hereto shall be liable for any default in the performance of its obligations under this EULA resulting from (i) a case of force majeure as defined by the law governing this EULA and the courts in such jurisdiction and (ii) the following causes: strikes (whether previously announced), war (declared or not), riots, governmental action, acts of terrorism, acts of God (fire, flood, earthquake, etc.).

8.4 Severability. If any part of this EULA is found to be invalid, illegal or unenforceable in any respect, the remaining provisions shall nevertheless be binding with the same effect as if the invalid, illegal or unenforceable part was originally deleted.

8.5 Transfer, Assignment & Subcontract. Licensee shall not subcontract, assign, delegate or otherwise transfer (including without limitation, by way of merger or contribution) any or all of its rights, duties, benefits or obligations under this EULA, or sublicense Licensed Software to any third party without Beet Analytics Technology's prior written approval. Any approved transfer of licenses to another country may be subject to an adjustment in price, as prices are specific to each country or region. This EULA shall be binding upon, and inure to the benefit of Beet Analytics Technology and its successors and assigns. Beet Analytics Technology may assign, delegate or otherwise transfer (including without limitation, by way of merger or contribution), any of its rights or obligations hereunder and/or otherwise subcontract any of its obligations, in whole or in part, to any Beet Analytics Technology and/or to any third party, without Licensee's consent.

8.6 Amendments & Non-Waiver. No waiver, alteration, modification, or cancellation of any of the provisions of this EULA or of any Affiliate Participation Agreement shall be binding unless made by written amendment signed by all parties. A party's failure at any time or times to require performance of any provision shall in no manner affect its right at a later time to enforce such provision.

8.7 Audit. During the term of this EULA and for a period of three (3) years thereafter, Licensee shall establish and maintain accurate information records relating to the use, and when applicable, destruction of the Licensed Software. Beet Analytics Technology shall have the right at any time, at its own expense and under reasonable conditions of time and place, to audit and copy these records. Licensee also hereby authorizes Beet Analytics Technology to verify its compliance with the terms of the EULA. For such purpose, Beet Analytics Technology may conduct an audit on Licensee's premises during normal business hours, in a manner that

minimizes disruption to its business. Beet Analytics Technology may require Licensee to provide it or any third party Beet Analytics Technology engages to conduct such verification, with machine access, copies of system tools outputs, or other electronic or hard copy system information as appropriate. If the audit reveals unauthorized use of any Licensed Software, Licensee shall promptly pay to Beet Analytics Technology any amounts owed as a result of such unauthorized use at the then current list price. In the event such unauthorized use is five percent or greater of Licensee's authorized licenses for the applicable Licensed Software, then in addition to Licensee paying the applicable charges, Licensee shall reimburse Beet Analytics Technology for the cost of such audit. In a joint effort to prevent software piracy, Licensee shall comply with any changes in the Licensed Software licensing security mechanism that aims at preventing fraud. By invoking the rights and procedures described above, Beet Analytics Technology does not waive its rights to enforce this EULA or to protect its intellectual property by any other means permitted by law.

8.8 Export. Export to Licensee of Licensed Software and Documentation is subject to all applicable countries' export and re-export laws and regulations. Licensee shall not export or re-export, either directly or indirectly, Licensed Software when such export or re-export requires an export license or other governmental approval without first obtaining such license or approval. Licensee hereby certifies to Beet Analytics Technology that the Licensed Software ordered hereunder will not be used in violation of any applicable export laws, including for proliferation of any nuclear, chemical or biological weapons or missile delivery systems and will not be diverted. Beet Analytics Technology may terminate this EULA and all licenses hereunder upon written notice if Licensee violates these provisions.

8.9 Entire EULA; Order of Precedence. These General Terms together with the Ordering Documents comprise the complete agreement between the parties relating to the subject matter hereof and supersede all prior and contemporaneous proposals, agreements, understandings, representations, purchase orders and communications, whether oral or written. If there is a discrepancy, inconsistency or contradiction between any Licensed Software and terms contained herein, the provisions of the corresponding Licensed Software Terms shall prevail, but solely with respect to those Licensed Software described in such Licensed Software Terms. Licensee acknowledges that it has full knowledge of all terms herein and incorporated herein, and agrees to be bound by and to comply with such terms and has not relied on the future availability of functionality or product updates with respect to any Licensed Software in entering into this EULA thereunder. The terms of this EULA shall have no force or effect with respect to any claim based on the use of any intellectual property rights of Beet Analytics Technology outside the scope of the licenses expressly granted herein.

8.10 Governing law and jurisdiction. This EULA shall be governed and construed in accordance with the laws of, and the legal relations between the parties shall be determined in accordance with, the laws of State of Michigan, United States of America, without regard to any conflict of laws principles and excluding application of the United Nations Convention for the International Sale of Goods. The parties irrevocably waive all rights to trial by jury for any such litigation between them. All actions and proceedings arising out of or relating to this EULA shall be exclusively heard and determined by the Courts of the State of Michigan, United States of America. Notwithstanding the foregoing, Beet Analytics Technology may, in its sole discretion, bring any claim or dispute (including but not limited to seeking injunctive relief and/or equitable remedies) arising out of, or in connection with the validity, interpretation and/or performance of this EULA before any courts and or administrative authorities having jurisdiction over the subject matter of any such claim or dispute. This provision shall survive any termination or expiration of the EULA. Licensee acknowledges and agrees that the paragraph immediately above shall not prevent, restrict or otherwise limit in any manner, Beet Analytics Technology's rights to seek equitable remedies, including injunctive relief before any competent court in any jurisdiction.

8.11 Survival. The following sections of these General Terms shall survive termination thereof: "License Limitations", "Intellectual Property", "Warranties, Limitations and Disclaimers", "Limitation of Liability", "Term and Termination", "Miscellaneous", "Glossary"..

8.12 U.S. Government Restricted Rights Legend. If Licensee is an agency or unit of the U.S. Government, the Licensed enVision User Manual 11 ©2012 Beet Analytics Technology. All Rights Reserved. Software and the related Documentation are "commercial items," specifically "commercial computer software" and "commercial computer software documentation," and, consistent with FAR 12.212 and DFARS 227.7202, as applicable, are licensed to Licensee only with those rights as are granted pursuant to this EULA. This provision shall survive any termination or expiration of the EULA.

8.13 Counterparts. This EULA may be executed simultaneously in two (2) or more counterparts, each of which will be considered an original, but all of which together will constitute one and the same instrument.

## 9. Glossary

"Authorized Users" means (i) the employees of Licensee, including employees of Licensee's directly and indirectly wholly-owned subsidiaries within the USA that are controlled by Licensee and unincorporated divisions of Licensee, but not employees of other legal entities (including employees of any legal entity that is a part of a group of companies affiliated with Licensee, but that do not qualify as direct or indirect wholly owned subsidiaries of Licensee), and (ii) Licensee's consultants, agents and contractors who are working on Licensee's premises, provided they are not competitors of Beet Analytics Technology and they have agreed in writing to restrictions on the use of the Software and obligations of confidentiality no less stringent than those set forth in this Agreement. For the purpose of this definition, "controlled" is defined as the direct or indirect ownership of more than 50% of the voting securities of an affiliate. Licensee shall hold Beet Analytics Technology harmless and shall indemnify Beet Analytics Technology for any loss, cost, damage or expense (including reasonable attorney's fees) incurred by Beet Analytics Technology as a result of the failure by an Authorized User to abide by the terms of this Agreement. Documentation means, at any time, the current user documentation in any form or media as delivered together with the Licensed Software by Beet Analytics Technology for use in connection with Licensed Software.

Effective Date of the License means, for any license for a Licensed Software, the latest of the following (i) the date on which such Licensed Software is shipped or made available electronically to Licensee by Beet Analytics Technology or, if applicable (ii) the date on which Licensee is informed by Beet Analytics Technology that the associated license key can be requested or is available. Licensed Software means (i) any data processing program for which a license is ordered by and provided to Licensee pursuant to a Quote, consisting of a series of instructions or databases in machine readable form, (ii) associated Documentation, (iii) corrective patches and (iv) Releases to which Licensee is entitled to the extent it has paid the applicable fees. Licensed Software do not include new versions of a Licensed Software, including any successor product which significantly differs in architecture, user interface or mode of delivery. Ordering Document means the order placed by Licensee and accepted by Beet Analytics Technology. Quote means the final commercial proposal containing a quote for Licensed Software made to Licensee by Beet Analytics Technology. Release means a periodic update of the same version of a Licensed Software if and when made generally available to the market. Specific Terms for Third Party Software means the specific terms relating to certain third party software components or products not developed by

or for a Beet Analytics Technology and licensed to Licensee to be used in connection with or within the Licensed Software. Software Maintenance Services means the maintenance, enhancement and other support services referred to in Section 2 hereof.

## **END OF END USER LICENSE AGREEMENT**

### **Copyright Information**

Information in this document, including URL and other Internet Web site references, is subject to change without notice. Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place or event is intended or should be inferred. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Beet Analytics Technology.

Beet Analytics Technology may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Beet Analytics Technology, the furnishing of this document does not give you any license to these patents, trademarks, enVision copyrights, or other intellectual property.

Copyright 2012 Beet Analytics Technology. All rights reserved.

ActiveX, Internet Explorer, Microsoft, Visual Basic, Visual Basic .NET, Visual C# .NET, Visual C++ .NET, Visual Studio, Visual Studio, Windows, Windows 95, Windows 98, Windows 2000, Windows Millennium Edition, Windows .NET server family, Windows NT, Windows XP, Windows Vista and Windows 7 are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.  
The names of actual companies and products mentioned herein may be the trademarks of their respective owners.