

Lean Automation (LA)

This feature groups three features into one (IED Wizard, Creating your Own IED & Mimic Panel/Face plate of the IED).

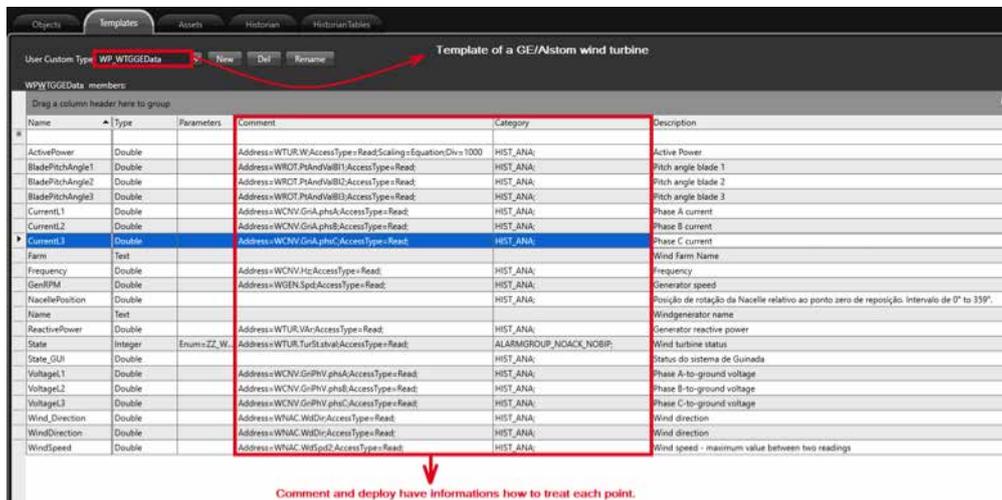
With the [LA methodology](#), developed by Spin and available in ActionNET, it is possible to configure complex automation applications for substations, power plants, wind farms, etc. in a [few minutes](#), guaranteeing the quality and absence of errors in parameterization. The purpose of the methodology is to automate an application such as several substations with the [following differentials](#):

- **Time: less** ↓ (months are converted into weeks);
- **Cost: low** ↓ (reduction of the staff, their specialization and the time of implementation, with less in expenses with travel tickets, car rental, hotel, meals, etc.);
- **Quality: high** ↑ (methodology minimizes failures and guarantees quality and finishing);
- **Flexibility: high** ↑ (from a solution you can get to new ones, easily).

The methodology is supported by the following five concepts and functionalities of Action.NET:

Template: template will be the basic information cell. For example, if the objective is to generate an application to create substations considering all [typical bays of a utility](#), you can have a template for each typical bay: feeder, transformer, line, reactor, etc. If the objective is an application to generate [wind farms](#), the templates will be of wind turbines, anemometrical towers, wind farms, etc.

One template is a record that contains all points of an information cell as a typical bay, a wind turbine, anemometrical tower, etc. However, in LA, each template point will have all information necessary to treat the point as, for example, its address inside the IED, alarm and historical conditions, nomination rules, etc.



Comment and deploy have informations how to treat each point.

Template of a Wind Turbine

Category: the software has an array of properties, created at design time that can be used in the LA projects to associate the points to functionalities. So, we can associate several of these available properties to three different functionalities:

a) Alarm: sets the type of alarm associated with one point. If, for example, my application has ten different alarm types, 10 category properties will be consumed to represents these alarms;

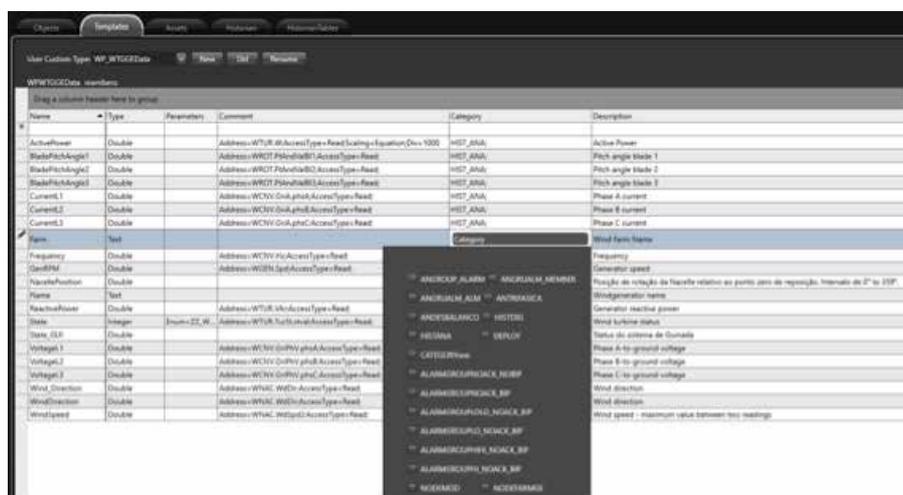
b) History: defines the conditions that cause the recording of the current state/value of the point in a history record. Again, as many properties will be consumed as are the historical recording conditions;

c) Points in Devices: the user must manually create the channels and nodes of the Application, according to the characteristics of the protocols used. This parameterization should be done according to the IEDs with which the project will communicate.

For each existing node, the system will automatically create a category with the prefix NODE_ followed by the node name.

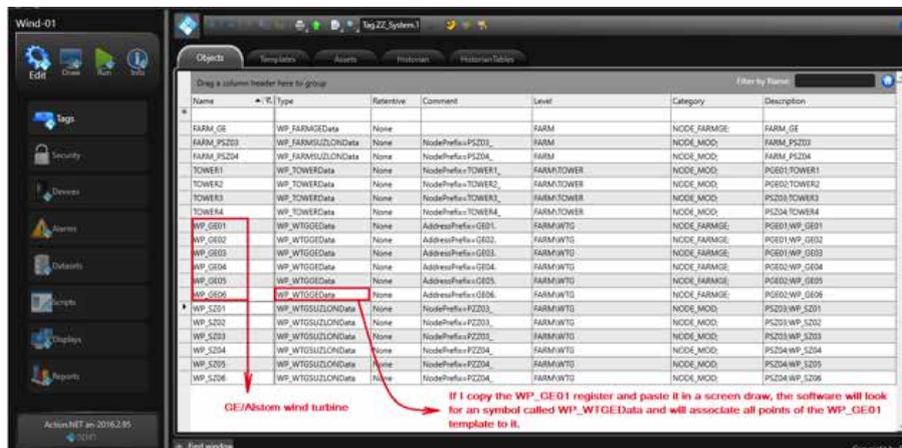
Category to Template Association: as shown in the figure below, in the templates, double clicking on the column category of each point opens a window where all the categories

created in this application are shown and the user associates one or more categories to each point of the template. Thus, each point has the respective alarm type defined, as well as the condition for its recording in historical files and the information referring to the devices, nodes, type of point in the protocol and address 2 of the point (address 1 is associated with the device).



Each point of the template can be associated to several categories

Linking the template name to the library symbol: Action.NET has a functionality oriented to the development of the methodology: if a template has the name of a symbol from the symbol library, it creates a link between them and the mapping of template variables associated with symbol mapping. Thus, in the wind farm application, WP_GE01 to WP_GE06 are associated to the template WP_WTGGEData and corresponds to six GE's wind turbines. If we copy one wind turbine register and paste in a screen draw, the symbol of a wind turbine, named "WP_WTGGEData" will be inserted in the screen and all points of it will be associated to that symbol.



There are six GE's wind turbines (WP_GE01 to WP_GE06)

Extensions Deploy: extensions are program libraries developed in the DotNET environment that, at project time, allow you to import, export, and process data associated with an application. In the case of Deploy, when it is executed it scans the Tag/Object lines and, for each line that has objects to be created (associated category), it automatically generates the alarm tables, history and points of devices. After that, it deletes the categories associated with that line. This allows that in the future, when creating a new span, only it will be generated, since the others do not have more associated categories. After the development of LA methodology, we developed the concept of component: a project or one module of a project with some functionalities. At the end, each LA project was transformed into a component.

Check some samples of LA components:

- [Lean automation – Component concept;](#)
- [Lean Automation – Customization of a power utility substation;](#)
- [Lean Automation – Wind power component;](#)
- [Lean Automation – Asset control component.](#)