

# ActionWise

Software SCADA / ADMS

Configuration and Operation Manual

Version 1 Revision 6

SpinEng - Automation

January 2019

## Revisions

1. January 2019 – Version 1.6 – English Version
2. August 2019 – Version 1.7 – Changed chapter 6.1

## Summary

1. Introduction.....	4
1.1 Presentation.....	4
1.2. Objectives.....	5
2. System Configuration.....	5
2.1 Electrical Network Data Generation.....	5
2.2 Historical Database.....	7
2.3 Using ActionWiseToolkit application.....	8
2.4 Configuring SCADA from Network Data.....	19
3. SCADA and ADMS Modules Databases Integration.....	25
3.1 Basic Design for an Electric Utility Company.....	26
3.2 Creation of a New SCADA / ADMS / FLISR Project.....	28
3.3 Procedures Performed on Importing Data from SinapRemote.....	29
3.4 Creating Graphical Screens in the ActionWise SCADA Module.....	33
4. Real-time interface (HMI).....	37
4.1 Functional Screens.....	37
4.3 Logic Blocks One-wire Diagrams.....	37
4.4. FLISR operation log screen.....	42
4.5 FLISR Event Reporting Screen.....	43
4.6 Operating Windows.....	44
5. ActionWise as Simulator.....	51
5.1 Simulator Operation.....	51
5.2 Simulation Interventions.....	52
5.3 Switch Opening.....	53
5.4 Switch Blocking for the FLISR Function.....	54
5.5 Change of Maximum Current.....	55
5.6 Return to Base Case (Normalization).....	55
6. ActionWise Simulation - Network Simulator.....	55
6.1 The Simulation Environment.....	56
6.1.1. Creating the Action.NET Gateway project.....	58
6.2. Running ActionWise / Simulator / Gateway.....	59
6.3 Executing ActionWise/ADMS/FLISR.....	62
6.4. Toolbar of the Simulator.....	62
7. Other Features.....	66
7.1 Simple Commands.....	66
7.2 Data Types in the protocol.....	66

# 1. Introduction

## 1.1 Presentation

ActionWise is a SCADA / ADMS application software implemented by the companies Spin Engineering of Automation and Sinapsis Energy Innovation.

Like all SCADA / ADMS, the solution includes a software module with SCADA (Supervisory Control and Data Acquisition) functions oriented to electrical applications and a module with ADMS functions (Advanced Distributed Management System).

The ADMS module includes a topological processor, a state estimator, a power flow module and a set of advanced functions for Distribution Network Management, such as the FLISR function (fault location, isolation and service restoration), highlighted in this document.

The two main differentials of **ActionWise**, when compared to other products on the market that also perform the FLISR function, are:

- 1) Electrical network modeling allows the integrated and complete representation of any topology, involving all segments of the system, making it ideal for the analysis of intelligent networks, where measurement, monitoring, control, automation and generation points are present at any voltage level;
- 2) The software, as shown in the figure below, works both in stand-alone mode, communicating with the field equipment, and in extension mode of the existing SCADA, in case the CLIENT already has a SCADA. In either of these modes, the solution can be deployed in the control center (COS) or in a substation or regional.

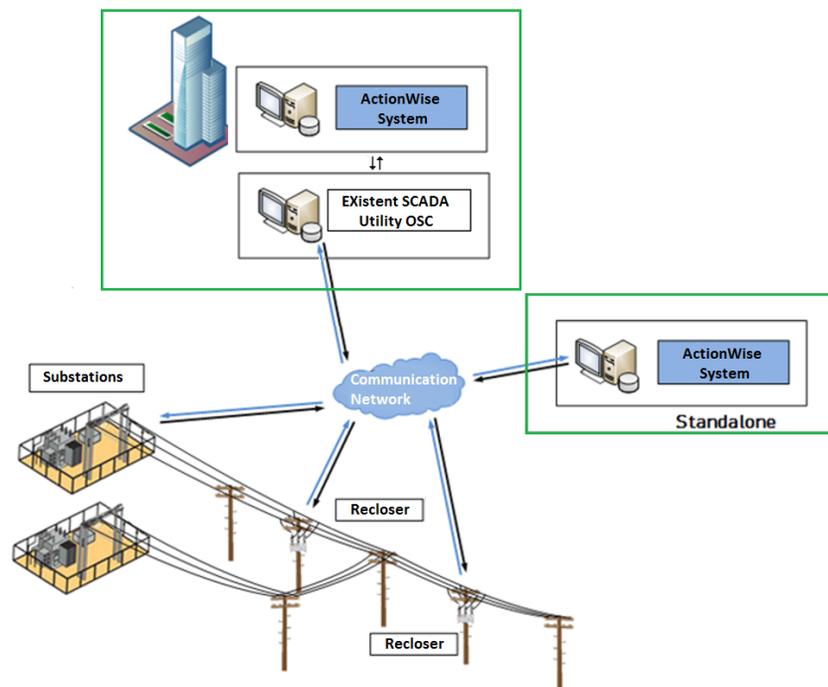


Figure 1 - ActionWise stand-alone or integrated with existing SCADA

## 1.2. Objectives

This manual presents the key features and functionalities of **ActionWise** software detailing its parameterization, testing, and real-time operation.

We do parameterization in two stages: in the first one, we should enter the network data; in the second, one integrates this data into the SCADA area, which is responsible for the interface with the process and the operator.

After parameterizing the system, **ActionWise** can operate in three different ways:

- **Real-time mode:** the software communicates with field equipment and implements the FLISR function, that is, using the data acquired in real-time, it identifies permanent faults in the electric network and proposes / performs actions to isolate the fault and restore the system, improving the reliability indicators and the quality of the service provided to the consumers.
- **Simulation mode:** the software does not communicate with field equipment, and one simulator engine calculates data from historic data of the modeled network. In this case, to verify the behavior of the FLISR function, it is possible to simulate faults through the SCADA HMI and verify how the system will be reconfigured
- **Listening Mode:** This is a special situation in real-time mode where the software acquires real data from the field and informs what it would do for recovery from occurrences, without actually doing so. This mode is used in the commissioning of the system, because the faults are real and the software only informs through the HMI, what would be the maneuvers executed.

## 2. System Configuration

### 2.1 Electrical Network Data Generation

The information required for the modeling of the power grid can be acquired in two different ways, as shown below.

#### 2.1.1 ActionWise toolkit

It allows the insertion of new networks or substations, defined through a set of elements that make up the set. These elements are bars, branches, loads, switches, generators and supplies.

The editor will also allow the "copy" and "paste" action of existing components on the same network or on different networks. We can do this action with multiple components selected in a single action. The multiple edition of similar equipment, such as wire branches arrangements and nominal buses voltages.

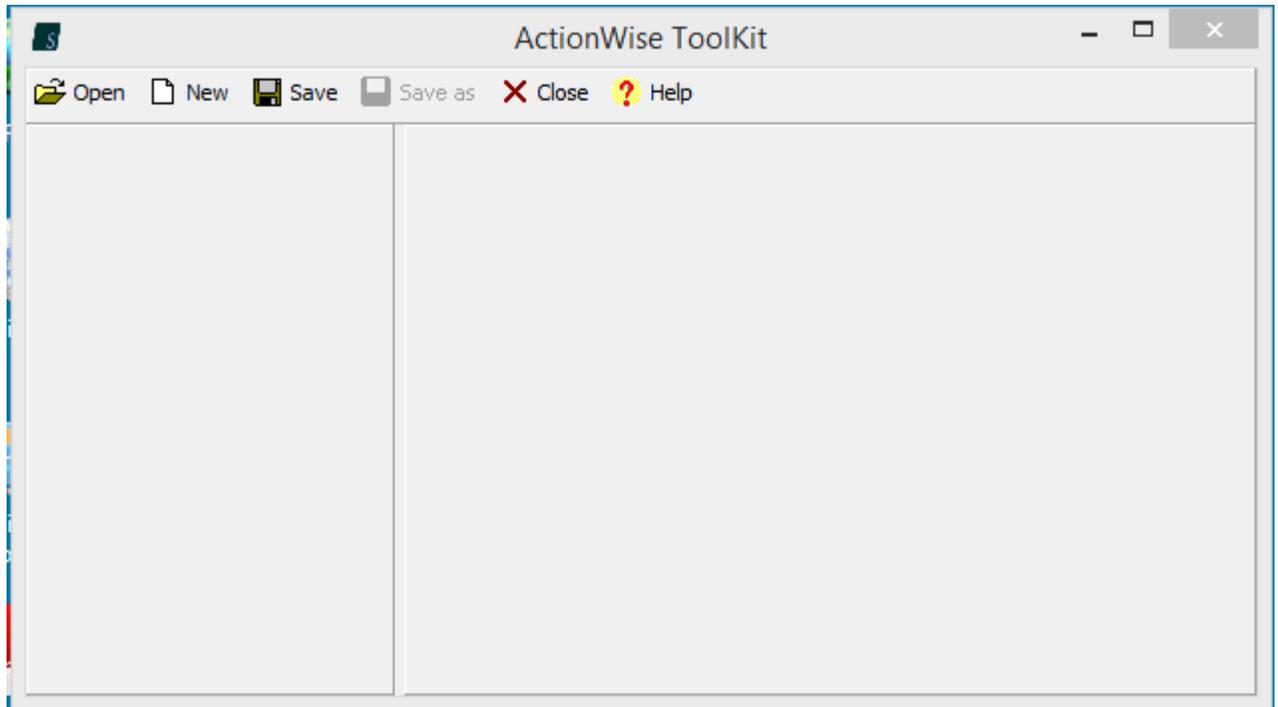


Figure 2 - Editor used to create the electrical grid with all the data needed for its modeling

### 2.1.2 Data Importer / Exporter

As an alternative to entering the data through the ActionWise toolkit, it is possible to import this data directly from the asset management modules and / or client studies. Thus, there are already importers to seek data from certain GIS data formats from electric utilities companies.

It is also possible to import typical demand curves for each class of consumers.

On the imported database, it is possible to perform a critical analysis of the data and identify any errors of data inconsistency such as, nominal power of transformers, branches with distances incompatible with georeferenced data, among others.

For more information on importing the data, the customer should contact Spin.

This document is limited to explaining data entry through the ActionWise Toolkit.

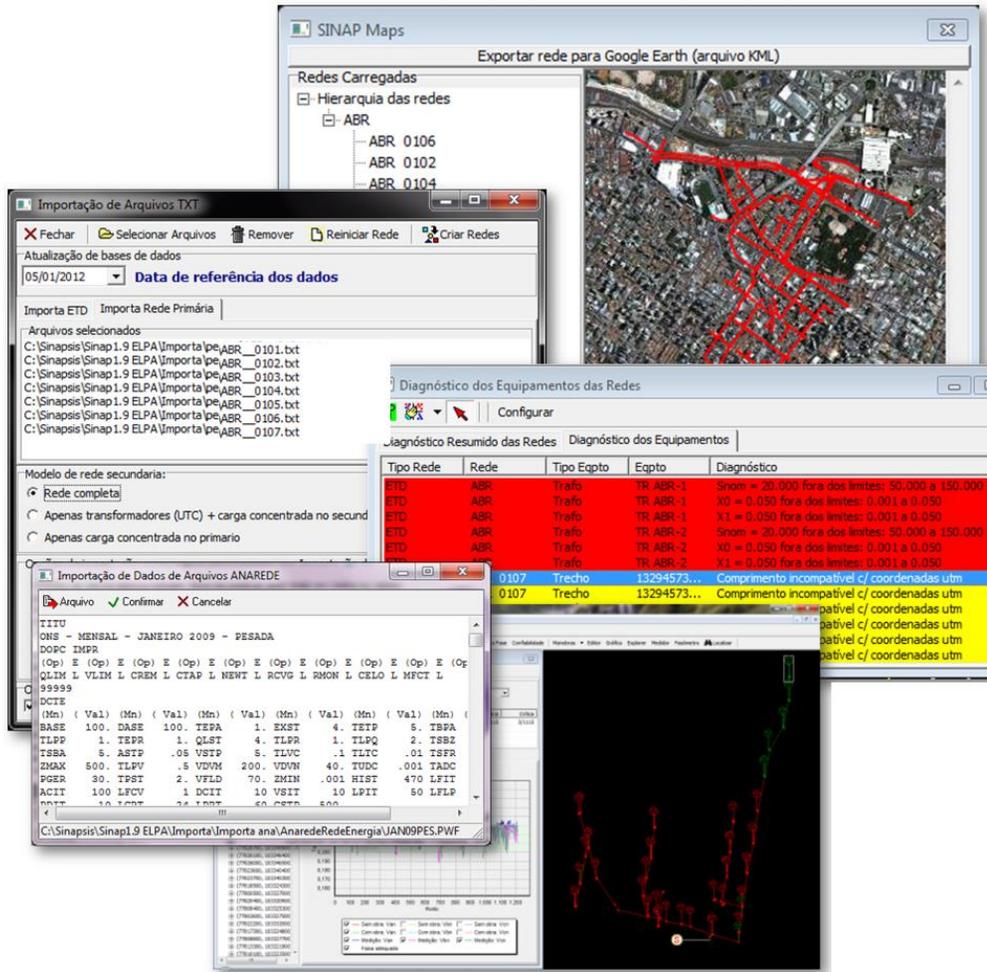


Figure 3 - Importing Third-Party Data

### 2.1.3 Model Topological Database

Whether through the ActionWise Toolkit or the data importer, at the end of the topology network creation, we obtain the set of network data saved in a relational database such as "MS-Access.MDB". This database, with a meaningful name for the project, will be used in real time by the ActionWise / FLISR modules.

The ActionWise software, at project creation time, will use this database for the automatic generation of data structure and process screens.

### 2.2 Historical Database

The ActionWise system stores the history of switching operations, events and other relevant information obtained during its operation. This database is used for guiding operations personnel during real-time network supervision and provides important information for future analysis.

## 2.2.1 Load Profile Files

The system represents the loads of the circuit through Load Curves containing values for each hour of the day. The program initially adopts standardized curves for the days of the week and for the weekends. After initiating the acquisition of field equipment data, the system will continuously update these data tables by means of power flow analysis.

The FLISR algorithm considers the load curves, when proposing the optimal sequence of switching operations, to analyze the feasibility of the reconfiguration plan for the next few hours. In this way, the network will not experience problems of overloading its equipment and branches after the switching operations.

## 2.2.2 Logging of System Operations (Log)

During system operation the ADMS-WISE electrical calculations module generates messages with current activity information ("operation logs"), which are stored in an appropriate table in the system's historical message database.

This information is stored for up to 60 days after its creation. They can be viewed in real time through daily reports or reports with historical searches of these messages. See item 4.4 for more details on this report.

Another information saved in this same database, in another table (Maneuvers), is the set of data involved in each occurrence of spontaneous opening of supervised keys. The date, interrupted load and description of the switch are recorded. In the case of creation of maneuvering plan for circuit recovery, the maneuver plane, the result of the operations, new load after the recovery are also stored. This information is used to generate the maneuver report described in item 4.5.

## 2.3 Using ActionWiseToolkit application

**ActionWise Toolkit** is a support tool used to mount the network topology that will be used by the ActionWise program. This topology will be composed of electrically independent **schemes** or **groupings**.

A **scheme or grouping** is a set of networks interconnected by remotely maneuverable switches (Vis a Vis) that allow its reconfiguration in case of contingency.

A **network** consists of a set of related equipment, defined according to user criteria.

### 2.3.1 Initial Screen

When you run the application, a splash screen is displayed, with the toolbar menus for using the program features. The figure below shows this screen.

The main tasks you can do with this menu are:

- Open Network - To open an existing database
- Create Network - To start creating a new network model
- Save Network - To save changes made to a network

- Close - to quit the application
- Help - to enter a help file with the same content as this description.

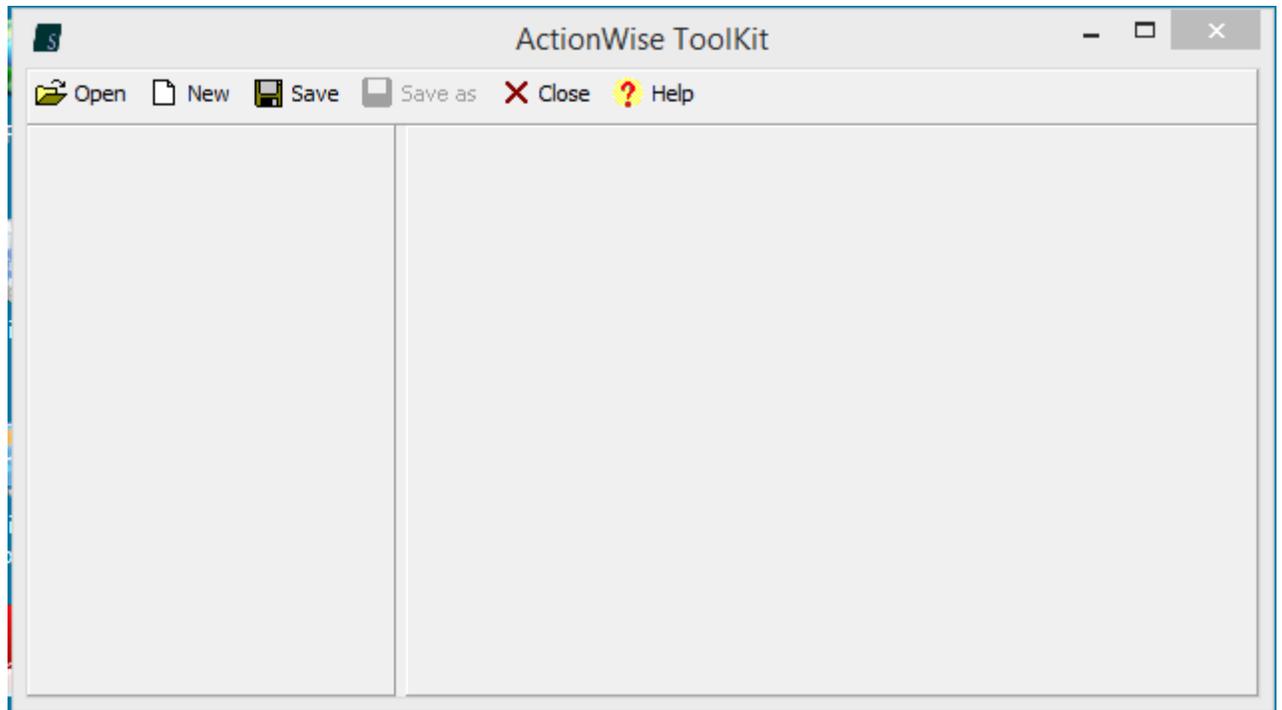


Figure 4 - ActionWise Toolkit Opening Screen

### 2.3.2 Open network

Through the Network menu, you can open networks from an **ActionWise** database.

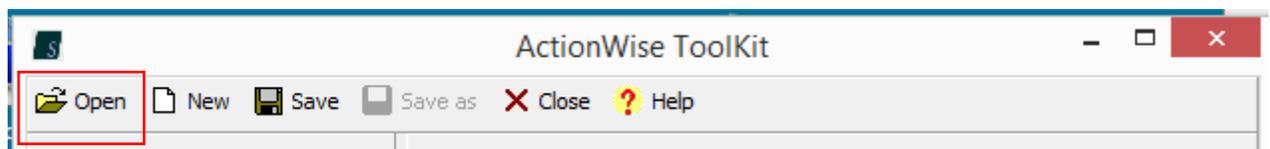


Figure 5 - Open existing network

To open a network:

- Select the database file 
- When you open the network, the **Groups** pane with the schema tree is updated, displaying all networks of all schemas in the selected database
- You can view the networks of all schemes, networks from a single schema, or just a single network.



- Click **Save**

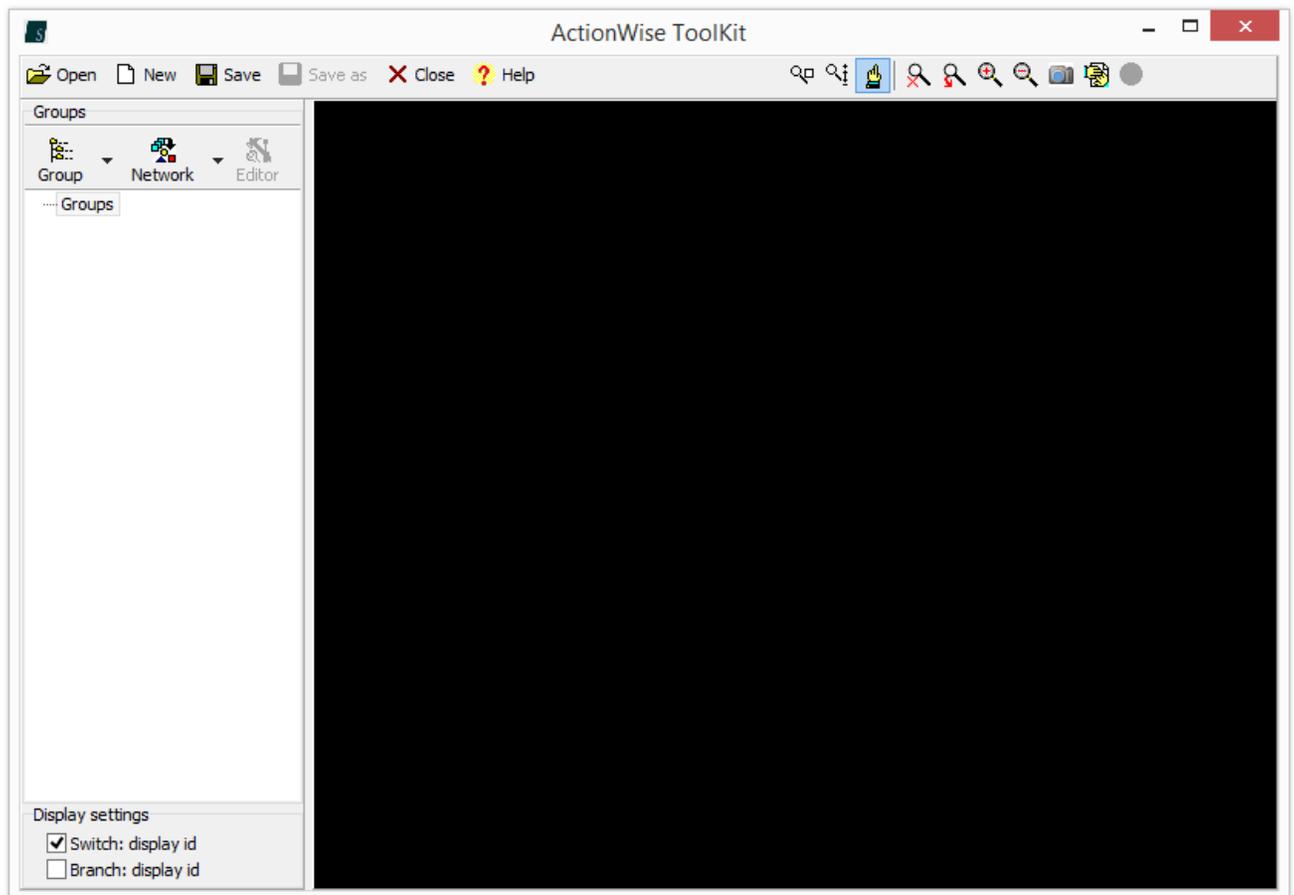


Figure 8 - Empty screen to start new network

To start editing the equipment in a network, you first need to create the **Grouping** and then the network associated with this schema. At this point, the button **Editor** will be enabled. This button displays the **edit dialog** with commands for inserting, changing or removing equipment.

### 2.3.4. To save

The button **Save** updates the current database.

The button **Save As** creates a new database, with the copy of network shown in the chart.

### 2.3.5. Groupings and Networks Panel

When you open a database, the grouping panel displays the following menus:

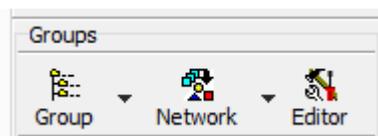


Figure 9 - Groupings and Networks panel

## Groupings

The **Groupings** menu lets you create a new schema, rename a schema, or delete a schema (and all your networks).

## Networks

The Networks menu allows you to create a new network for a Grouping, change the name of a network, or delete a network (and all its equipment).

## Editor

The Editor menu allows you to display the Editor Tool Window, which contains the symbols to create and insert any equipment in the network.

### 2.3.6. Network Associated Command Menu

After opening a network, the following buttons are also available:



Figure 10 – Command Menu

#### Change the right mouse buttons function

By clicking on the icons  (zoom by area),  (pan) and  (zoom by vertical scroll) the application modifies the function of the right mouse. Use this right button when you click on the network and drag.

You can use the **CTRL**, **ALT**, and **SHIFT** keys together with the right mouse button to zoom by area, pan, and zoom by vertical scrolling, respectively.

- Press the right mouse button and select area (zoom by area) 

#### Zoom Functions



In the figure, the four icons in sequence allow, when pressed:

- Zoom out: shows the total area of the network
- Return to previous zoom: shows previous zoom levels until you reach the total area
- Increase level of detail: zoom in
- Decrease level of detail: zoom out

**Save Graphic**  - Write to a bitmap (.bmp) file; clicking on the button will make a Network Print exactly as it is at this time on the screen.

**Wizard**  - View the attributes of the equipment's on the screen itself, in a small temporary window that opens and closes automatically for each piece of equipment pointed by the mouse

cursor. The attributes displayed are the same as those configured by the graphical configuration module.

### 2.3.7. Editor

The button **Editor** displays the **Editor** screen with the symbols that allow you to change, remove, or insert equipment on your networks. To have this button enabled, this database should have at least one network configured.

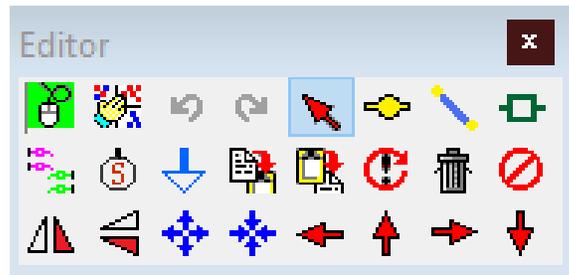


Figure 11 – Editor Buttons

 Active window	 Copy equipment
 Window alignment	 Paste equipment
 Undo	 Delete equipment
 Redo	 Expand
 Equipment Edit	 Contract
 Insert bus	 Flip horizontally
 Insert branch	 Flip vertically
 Insert switch	 Move left
 Change switch state	 Move up
 Insert supply	 Move right
 Insert load	 Move down

Figure 22 – Action get from each editor button

### 2.3.8. Editor Equipment

**Change key status:** The button  allows you to change disconnecting / reclosing devices to their complementary state (if closed, open, if open, close).

- Click the button 
- Click on the Switch you want to change status

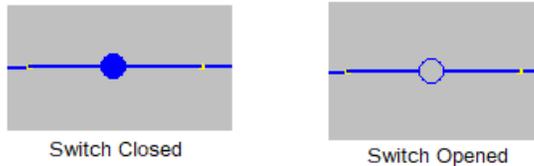


Figure 33 – Switch Position

**Insert bar (bus)**  : A bar is the point or bus that connects equipment in a network. We can draw a bar as a dot (one click) or as a horizontal or vertical bar (by clicking and dragging horizontally or vertically).

The editor considers as **connections** the equipment that require two bars to be drawn. These are the **branches and switches**.

The editor considers equipment of one bar the elements dependent of a single bar to be drawn: **loads and supplies**.

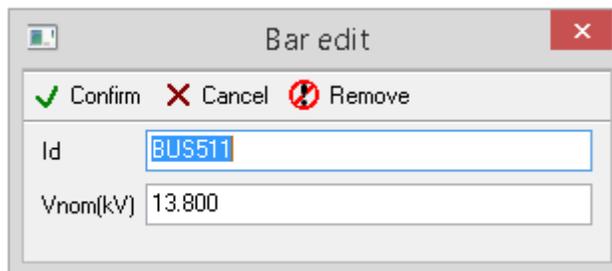


Figure 14 - Attributes of a bar

- **Confirm** – After Editing / inserting / removing the bar, the changes are executed and the screen is closed;
- **Cancel** - To close the dialog without editing / insert / remove bar
- **Remove** - To remove this bar from the grid.
- **Id** - Name of the bar
- **Vnom (kV)** - Nominal bus voltage (phase-to-phase and in kV)

**Insert Branch:**  A Branch is a link that can be drawn by clicking on the source bar and dragging it to the destination bar. In horizontal or vertical target bars, as shown below, you can use the Shift key together to draw the part aligned with the vertical or horizontal.

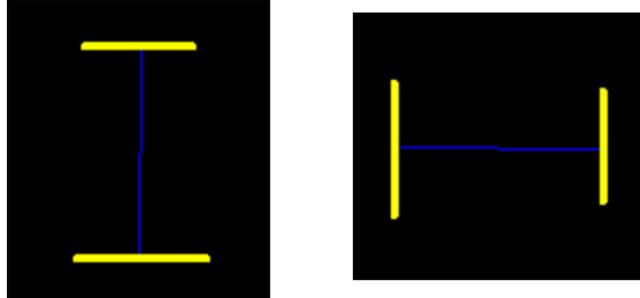


Figure 15 - Vertical and horizontal line

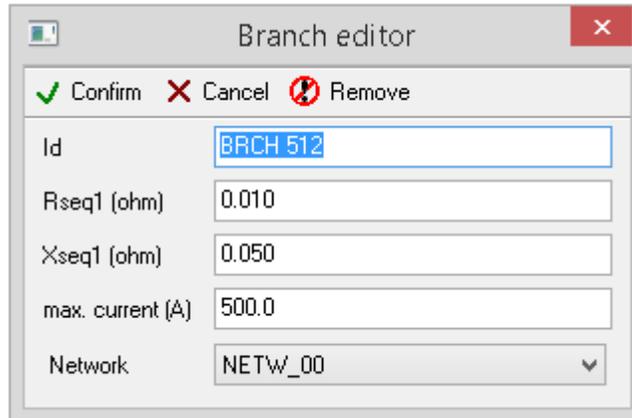
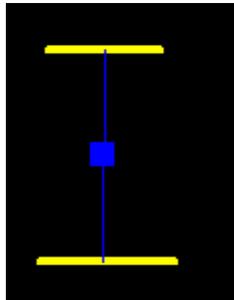


Figure 16 - Attributes of a Branch

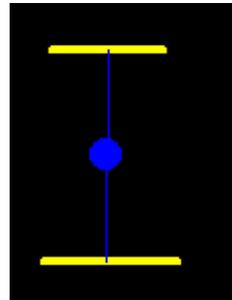
-  **Confirm** After editing / inserting / removing the branch, the changes are executed and the screen is closed
-  **Cancel** Close the screen without editing / insert / remove the excerpt
-  **Remove** To remove this Branch from the grid
- **Id** - Name of the excerpt
- **Rseq1(ohm)** and **Xseq1(ohm)** - Sequence impedances
- **Max Current (A)** - Maximum permissible current for cable type of this branch segment

- **Network** to which the section belongs

**Insert Switch:**  A switch is a link or connection. It is drawn by clicking on the source bar and dragging to the destination bar.



Recloser



Circuit Breaker

Figure 17 -Connectors (vertical and horizontal).

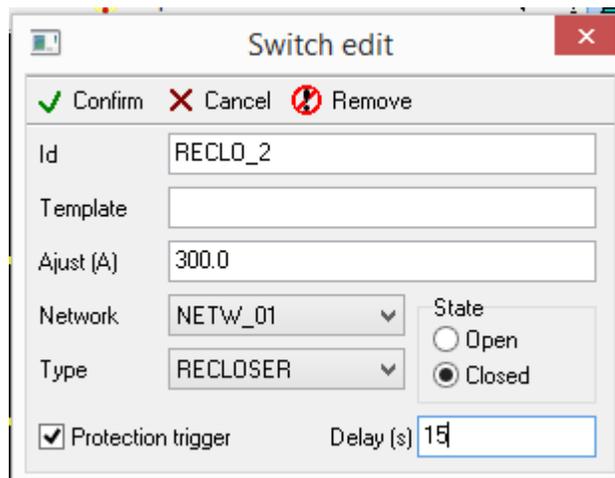


Figure 18 - Switch Attributes

-  After editing / inserting / removing the switch, the changes are executed and the dialog is closed
-  Close the screen without editing / insert / remove the switch
-  To remove this Switch from the grid
- **Id - Switch name**

- **Template** - Name of the template in Action.NET used for this key, if not with the name KEY (some alternative such as a circuit breaker, recloser)
- **Adjust (A)** -Setting is the nominal current ( $I_{nom}$ )
- **Network** to which the Switch belongs
- **Type** of switch: disconnecter or recloser
- **Protection Trigger** - In case of circuit breakers or reclosers where the lockout signal is not available or the use of the multiple reclosing criterion is not possible to trigger recovery maneuvers we can select this option. This cause the use of an algorithm for the detection of openings by protection action, by More details see item 4.6.4
- **Delay (s)** Time in seconds after the protection action

**Insert Supply** : One Supply or Source is an equipment connected to one bar only. It represents an energy source, an energy producer or a generation point feeding the network.

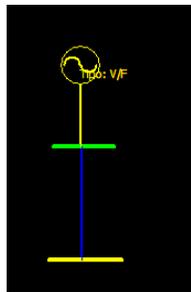


Figura 19 – Attributes of sources

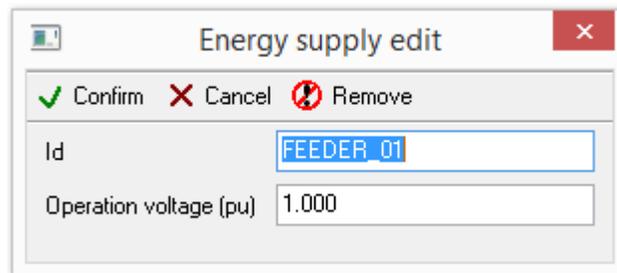


Figure 20 – Edition of sources

- **Confirm**  After editing / inserting / removing the excerpt, the changes are executed and the screen is closed
- **Cancel**  Close the screen without editing / insert / remove the excerpt
- **Remove**  To remove this Source-Supply from the grid

- **Id** - Name of the excerpt
- **Operation voltage (pu)** - Supply operation voltage

**Insert Load:**  A load is a one-bar equipment, which can be drawn by clicking on a bar and dragging in one direction. It is a consumer of energy.

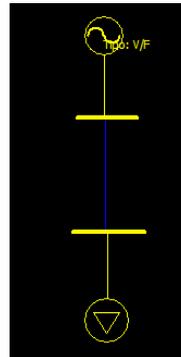


Figura 21 –Load

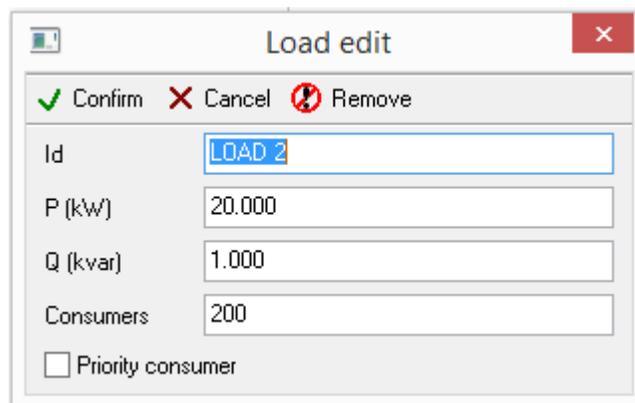


Figure 22 – Attributes of Loads

-  **Confirm** After editing / insertion / removal of Loads, the changes are executed and the screen closes
-  **Cancel** Close the screen without editing / insert / remove the Load
-  **Remove** To remove this Load from the grid

- **Id** - Name of the Load

- **P(kW)** - Active power information of the inserted load
- **Q (kVAR)** - Reactive power information of the inserted load
- **Consumers** - Number of consumers that the load represents
- **Priority Consumer** - Mark this in the case of important priority loads you want create and show in ActionWise one-wire displays. In ActionWise, there is a different symbol for normal load and priority consumer load.

**Copy and paste equipment:** The button  allows the selected equipment (by the button ) to be copied (shortcut key: Ctrl + C). To paste these equipment, you must press the button  (shortcut key: Ctrl + V) and click where they should be pasted. We can paste it on the same screen where copied, in another window with other networks, or in a window without networks. Once pasted, the equipment assume the attributes of the original equipment. When copying equipment's they assume the isolated state, without a network definition, that is, it is necessary to associate a network with the copied equipment.

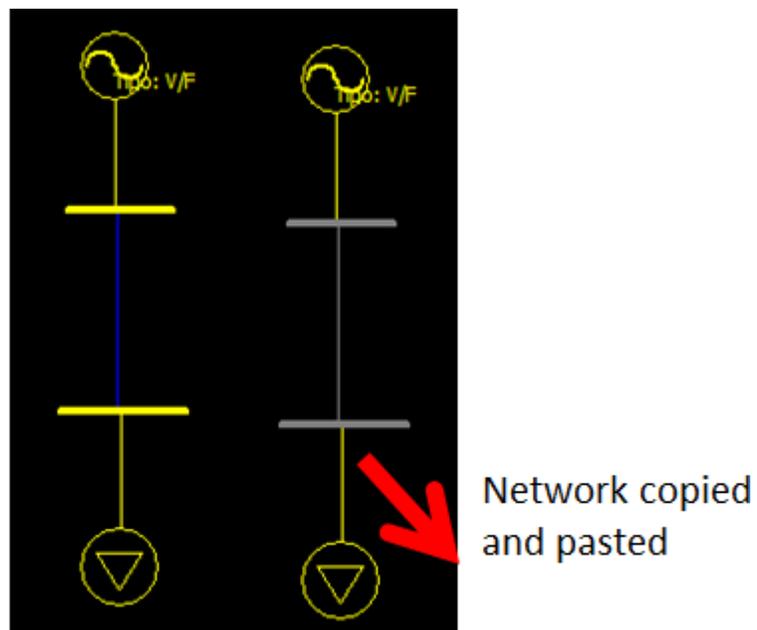


Figure 23 – Copy / Paste equipment

## 2.4 Configuring SCADA from Network Data

In **ActionWise** system the modeling of electrical networks, including topological georeferenced characteristics, interconnections, equipment, electrical parameters of the equipment, load profiles, etc. is all in this database created using **ActionWiseToolkit** application. These are all the data parameters needed to allow the electrical studies calculations.

For the **ActionWise** SCADA module operation, however, will need only some of this data. An import extension procedure available in the **Action.NET** management application will transfer the data needed to ActionWise Scada project.

This import routine found in **Run / Extensions**, named **SinapWise Import**, reads the topological base obtained in ActionWiseToolkit. For this importer, this dataset name is **SinapRemote**, and when reading it, the importer creates a simplified database containing only the objects needed to support the real-time integration between the SCADA module and the ADMS module. The importer also makes the generation of text files with data to create in Action.NET graphical interface displays to be used for ADMS MMI users.

The following figure shows, in the ActionWise-editing module, the MDB file declaration used by the Toolkit (**SinapRemote**) and the DB3 file generated by the **SinapWise Import** extension from the MDB file (SinapLocal). This same extension also generates a CSV file, used to generate the diagrams of the FLISR groupings. The same figure shows a block diagram with the files and the processes that use its.

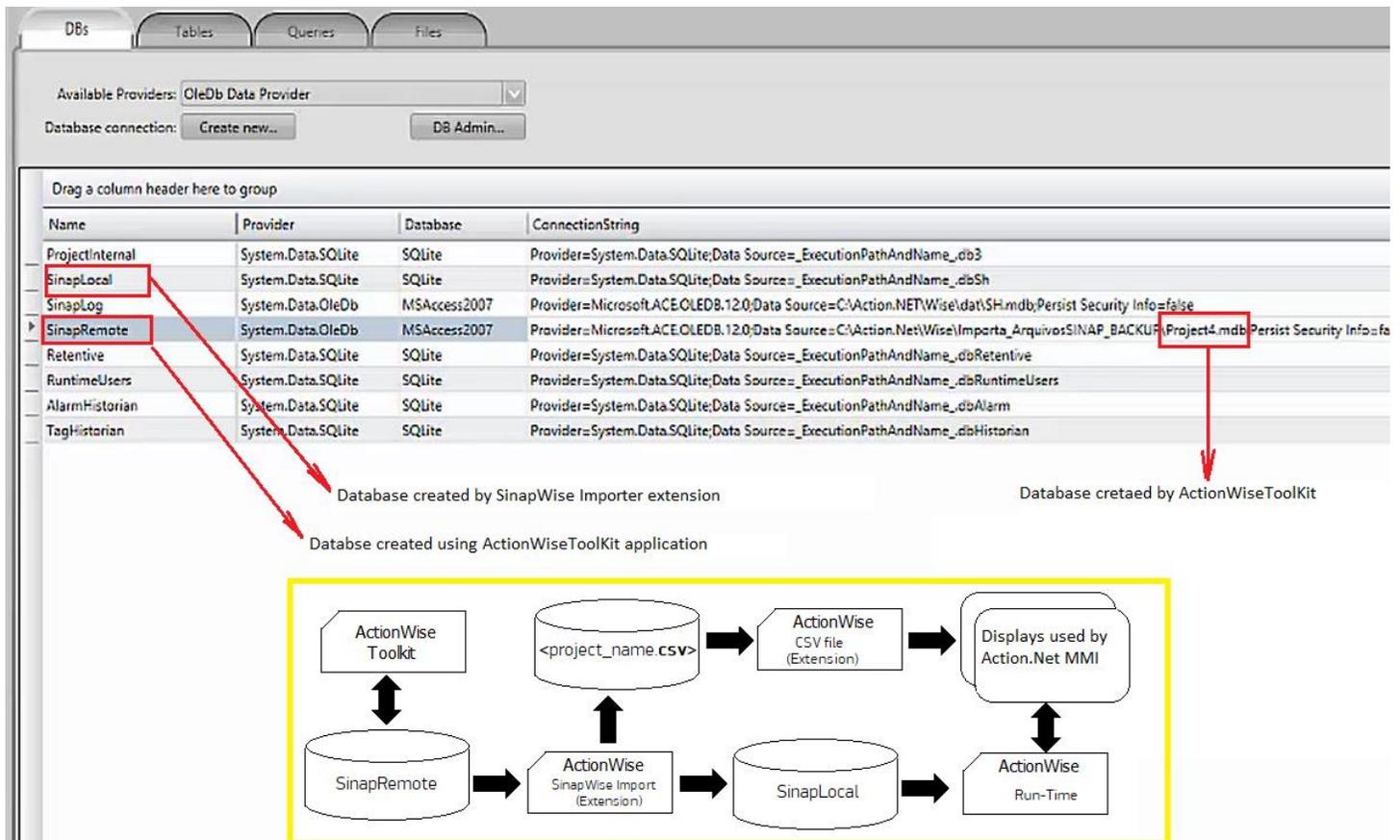


Figure 24 – Database used by ActionWise (Toolkit and Run-time)

## 2.4.1 Data Available to the SCADA Module

The import routine generates a database with extension "dbSh", corresponding to an internal relational database of the SCADA module (**SQLite** Database). This database, as shown in the figure below, will have the same name as the project created by the engineering tool of the SCADA module. In SCADA, this database is interfaced by the name of **SinapLocal** dataset.

The importer creates and populates the tables in this database, with relevant data to the integration, contained in the original topological database, called SinapRemote.

- **Groupings** - Contains the records of each of the logical schemes, or here called Groupings. Each grouping is a set of Switches, Branches, Feeders, Loads, Networks, etc., existing in the same project, electrically isolated from others groupings.
- **Equipment** - contains the records of each of the equipment described in the original database. The equipment objects are the main elements of this model, for example, switches, circuit breakers, bars, networks, supplies, branches of networks, lines, etc.
- **Bar** - contains the records that define each of the bars in the network. The interconnection of equipment, by this model, is always done through bars. In each bar record, there are the positioning coordinates of the bar.
- **Networks** - Each network is the complete set of interconnected equipment objects, which will be considered in the implementation of the system. In each network, record there is the coordinates of the area in which the network is.
- **Equipment type** - contains the codes and names of the equipment that can be used in the network.
- **Switch Type** - contains codes and names of different types of switches, such as, Circuit Breaker, Recloser, Disconnecter,
- **Switches** - contains records for each of the system switches, with the identification, type and initial normal state of the switches.
- **Network Type** - is the list of different types of networks that can be handled by ActionWise. It contains a numeric identifier of the type, the descriptive name and an acronym that indicates if the network is transmission or distribution (SET = Transmission System, SED = Distribution System, etc.).
- **BarNetwork** - is the list that lists each bar with the Network to which it belongs.
- **Bar equipment** – this is a list of records with relations between equipment to the bars: Which bar they connect to and the position on the bar.

## 2.4.2 SCADA Module Design

The SCADA module project must be created from a basic **ActionWise** project “empty” that already has all the templates of the structures, all symbols and other ActionWise objects. This project named **FLISR-Project.tproj** is included in the ActionWise installer. To create an individualized project for a deployment, copy this file and change the name to one that is significant for it.

The project thus created is an Action.NET project with all the structures, files, symbols needed to start a project, ADMS / FLISR.

The following are the required objects already available in ActionWise **FLISR-Project.tproj**:

a) **References to relational databases** associated with SinapRemote and SinapLocal:

- In **Edit> Datasets** in the DB tab - There is a Dataset record for access to the ActionWise Toolkit database (which is in Microsoft Access format - mdb); the name of this DB is "SinapRemote". Use the name of this network file to name the same project with extension ".mdb".
- In **Edit> Datasets** in the DB tab - There is a Datasets record for access to the local database that will be created with "DB3" format (SQLite); The name of this DB is "SinapLocal", it must use the same project name and extension ".dbSh".
- In **Edit> Datasets** on the DB tab - There is a Dataset record for access to the local database where written **Log and historical data** of Maneuvers are stored by the ActionWise calculation module. The name of this dataset is SinapLog and the data has the named "SH\_" followed by the project name and extension ".mdb".

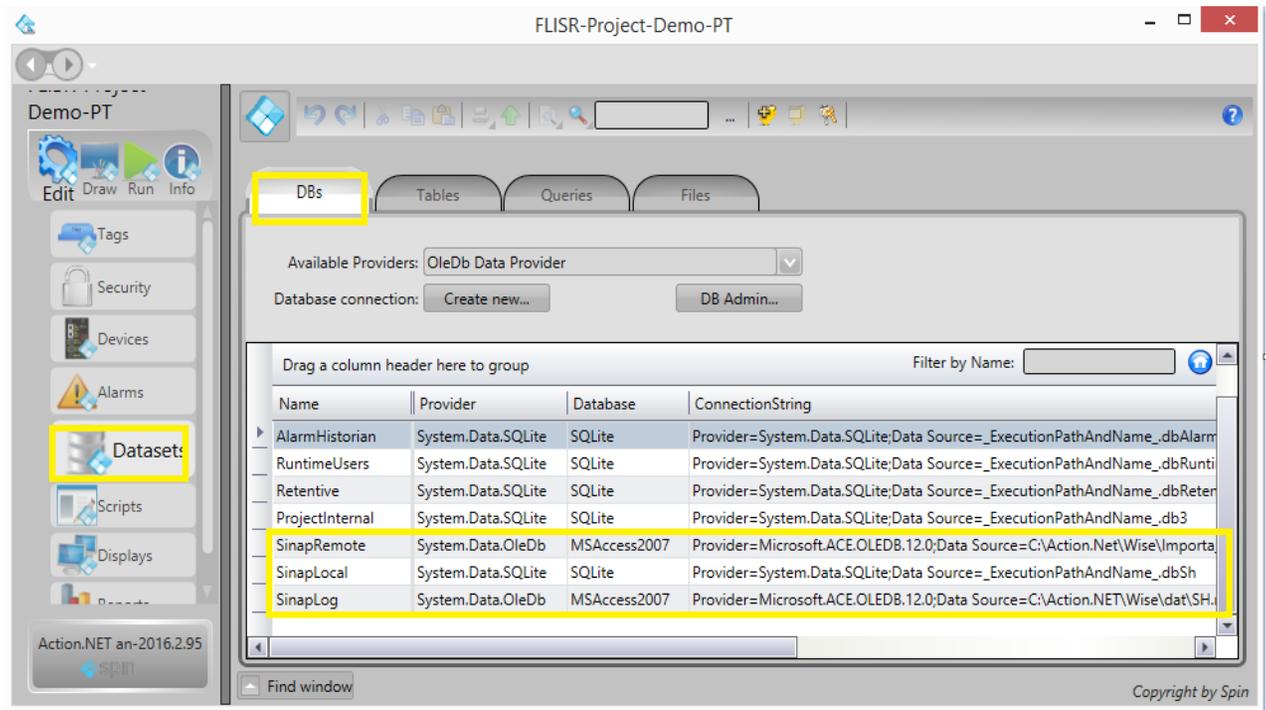


Figure 26 – Declaration of databases used in ActionWise

b) **Communication channels** (integration bus) used in real time for the exchange of data between the SCADA and ADMS modules. This communication was implemented through a proprietary protocol, called SinapWise available in both modes: client and server:

- Client: SCADA requests from the ADMS Module the load status, the colors of the sections (Responsible feeders), estimated values, indicative of maximum loads reached.

- **Server:** Through this communication channel, the SCADA module sends to the ADMS module the field data, i.e. key states and electrical quantity measurements currently in real time. Inversely the module ADMS sends to the SCADA through this channel the plans of maneuvers and estimated measures.

These channels are declared in the SCADA engineering tool in **Edit> Devices**. In this section the channels (client and server), the nodes of these channels (IP addresses: Port) and I / O points exchanged between the two modules are defined. Thus, in the **basic ActionWise project**, these channels are already defined, and it is only necessary to adjust variable attributes, depending on the architecture of the network.

- In **Edit> Devices** on the Channel tab there will be a communication channel with the SinapWise protocol, in Server mode and other in Client mode. (See the Online Manual for details on this configuration - Devices and Interfaces).
  - In **Edit> Devices** in the Nodes tab there will be two communication node for the channels server and client defined in the previous item (See the online manual for details on this configuration - Devices (Devices and Interfaces)).
  - In **Edit> Devices** in the Points tab, all the points that will be read and written through the two modes (client and server) will be automatically generated, but these points will be described later in the detailing of the import - extension routine SinapWise and Deploy. Extension.
- c) **Templates** (templates) of all the equipment that makes up the electrical network modeled by ADMS / FLISR. These templates are already created in Edit> Tags in the Templates tab. Will exist as many templates as the equipment used to model the data of the equipment defined in the ActionWise - ADMS module. Routines for automatic generation of the application database will use these templates, whose names will be the same names of the equipment types, to instantiate corresponding Object Objects. The following item presents these Templates.
- d) **Main screen** for a FLISR project, with buttons for the call of the schemes, and selection of general modes of operation.
- e) **Windows** (dialog screens) for the Maneuver, Diagnostics and Switches command.
- f) **Symbols** (graphic) for building one-way wiring displays to show topology of networks and groupings.

### 2.3.3 Templates for SCADA / ADMS Integration

The SCADA engineering tool uses templates to define the equipment with all relevant data for the integration of the SCADA and ADMS modules. The main features of these templates are:

- The name has the prefix **SW\_** (short for SinapWise) followed by the name of the **type of equipment** that it models, in capital letters.

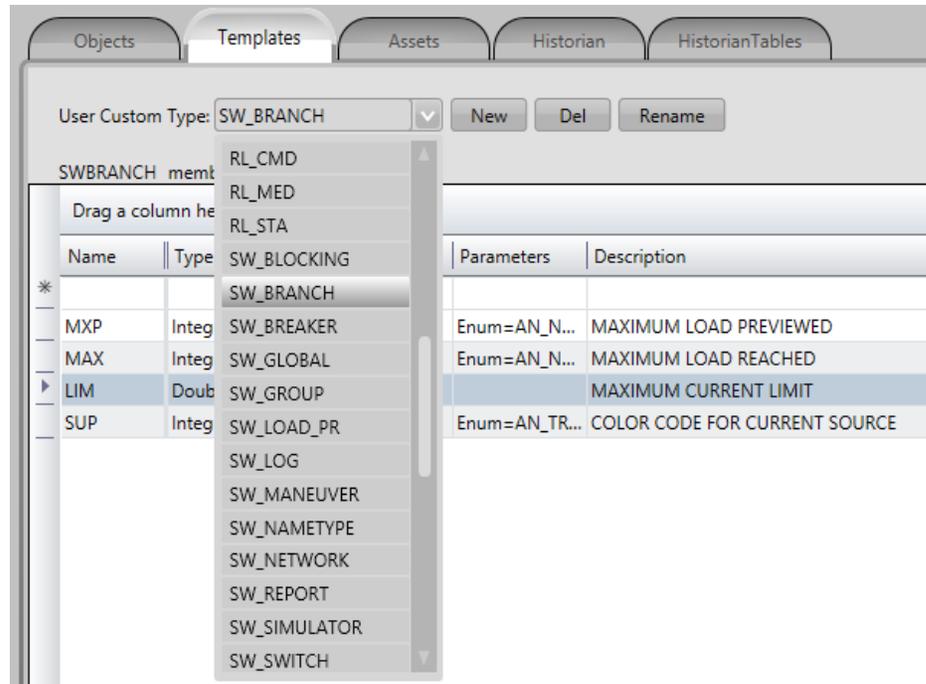


Figure 27 - Equipment models used in ActionWise (SW\_ <Equipment Type>)

- The template can include sub-templates as data items. These data items are pertinent to SCADA x ADMS and SCADA x Field integration.
- The **SW\_BREAKER** Template, for example, is composed of three sub-templates: STA (type status - digital variables of switch), CMD (Type Commands – tags used to send commands to breaker) and MED (Type Measurements –with measurements contents tags).
- The following figure shows the sub-template RL\_STA (Digital Variables) where only members marked with yellow are relevant for the SCADA x ADMS integration. The system will read from the field, in real time, the others members, needed for the supervisory functions of the switch in question.

**Note:** In projects where there are relevant differences between the tags of a type of switch, and of another type, we suggest have more than one template for this definition. The ActionWise allows define as many types as needed, as long as the main tags, shown in yellow in the next figure, are maintained, for the protocol communication between SCADA vs. ADMS. The names of these alternate templates should start with **SW\_** and be included in the "strExtId" field in the Equipment table, without the prefix **SW\_**.

Objects Templates Assets Historian HistorianTables

User Custom Type: RL\_STA [v] [New] [Del] [Rename]

RLSTA members:

Drag a column header here to group

Name	Type	Array	Parameters	Description
* BLA	Integer		Enum=AN_AT...	FLISR BLOCKED BY ACTUATED SIGNALS
BLF	Integer		Enum=AN_AT...	FLISR MANEUVER EXECUTION
BLM	Integer		Enum=AN_AT...	FLISR BLOCKED COMMANDED BY OPERATOR
BLMEC	Integer		Enum=AN_N...	MECHANICAL BLOCK
BLR	Integer		Enum=AN_AT...	FLISR BLOCK FOR ALL NETWORK (LV, R79, R51NB)
CFL	Integer		Enum=AN_N...	COMANDO EXECUTADO POR SELF HEALING
CH43REM	Integer		Enum=AN_L...	GENERAL LOCAL/REMOTE SWITCH
COMM	Integer		Enum=AN_N...	COMMUNICATION FAIL
▶ CTRLOK	Integer		Enum=AN_N...	CONTROL OK FOR FLISR
DELAY	Integer			DELAY FOR WAITING LKO
FEQUIP	Integer		Enum=AN_D...	EQUIPMENT FUNCTION - SWITCH
INVPOS	Integer		Enum=AN_N...	CONTACTS INVALID POSITION
LKO	Integer		Enum=AN_N...	AUTOMATIC RECLOSE LOCKOUT
LV	Integer		Enum=AN_D...	HOT LINE SWITCH
MAX	Integer		Enum=AN_N...	MAXIMUM LOAD REACHED
MAXP	Integer		Enum=AN_N...	MAXIMUM LOAD PREVIEWED
OPSH	Integer		Enum=AN_O...	OPERATED BY FLISR MANEUVER
P27CA	Integer		Enum=AN_N...	AUXILIARY AC SOURCE
POS	Integer		Enum=AN_DI...	POSITION
R51NB	Integer		Enum=AN_BL...	NEUTRAL REALY
R79	Integer		Enum=AN_RE...	RECLOSING MODE SWITCH
RCF	Integer		Enum=AN_N...	MANUAL RECONFIGURATION REQUEST
RESETSH	Integer			RESET COMMAND FOR FLISR
SUP	Integer			COLOR CODE FOR CURRRENT SOURCE
SWTYPE	Integer			SWITCH TYPE: 1=BREAKER; 2=RECLOSER; 4=DISCONNECTOR
TENBAT	Integer		Enum=AN_N...	BATTERY VOLTAGE

Figure 48 – Template RL\_STA with several field tags. Wise algorithms use the yellow tags by iteration Scada x ADMS Wise.

### 3. SCADA and ADMS Modules Databases Integration

As presented in Chapter 2, ActionWise Toolkit is the tool to update the SinapRemote database (MDB). An empty database is already available, with cable types, equipment types, switches types and their default attributes, etc. Using the ActionWise Toolkit, from this empty database, we will generate the groups, and their electrical networks and equipment, as shown in the video.

Later, using the Action.NET engineering tool, you include the data from the SinapRemote database to the new project created from the basic ActionWise project.

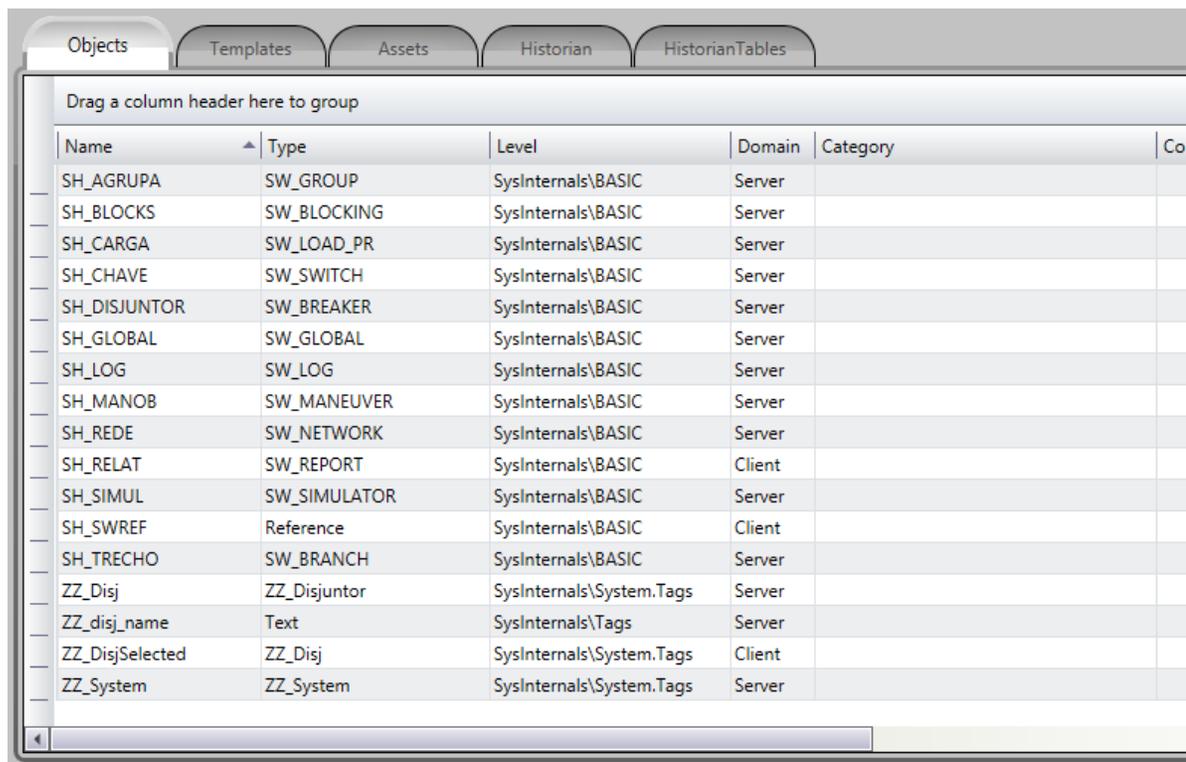
In this project, already exists the entire basic structure of ActionWise required to generate a new application of SCADA / ADMS FLISR.

### 3.1 Basic Design for an Electric Utility Company

The **basic ActionWise project** should contain the basic elements already described, such as datasets, devices, templates, symbol libraries, command windows, required for SCADA / ADMS / FLISR integration projects.

However, as each energy utility has different cultures from the others, with regard to substantive points of the reclosers, rules for naming tags associated with an equipment, etc., it is suggested that for each company be created, from the **basic project** a new basic project, using templates, symbol libraries, etc. suitable for this utility.

- The following figure shows the set of basic templates in the basic ActionWise project.



Name	Type	Level	Domain	Category	Co
SH_AGRUPA	SW_GROUP	SysInternals\BASIC	Server		
SH_BLOCKS	SW_BLOCKING	SysInternals\BASIC	Server		
SH_CARGA	SW_LOAD_PR	SysInternals\BASIC	Server		
SH_CHAVE	SW_SWITCH	SysInternals\BASIC	Server		
SH_DISJUNTOR	SW_BREAKER	SysInternals\BASIC	Server		
SH_GLOBAL	SW_GLOBAL	SysInternals\BASIC	Server		
SH_LOG	SW_LOG	SysInternals\BASIC	Server		
SH_MANOB	SW_MANEUVER	SysInternals\BASIC	Server		
SH_REDE	SW_NETWORK	SysInternals\BASIC	Server		
SH_RELAT	SW_REPORT	SysInternals\BASIC	Client		
SH_SIMUL	SW_SIMULATOR	SysInternals\BASIC	Server		
SH_SWREF	Reference	SysInternals\BASIC	Client		
SH_TRECHO	SW_BRANCH	SysInternals\BASIC	Server		
ZZ_Disj	ZZ_Disjuntor	SysInternals\System.Tags	Server		
ZZ_disj_name	Text	SysInternals\Tags	Server		
ZZ_DisjSelected	ZZ_Disj	SysInternals\System.Tags	Client		
ZZ_System	ZZ_System	SysInternals\System.Tags	Server		

Figure 31 - Basic templates available in Basic FLISR Project

- **Channels and Nodes** will be the server and client channels and the nodes. In Datasets the accesses to the database. In Assets a Level with the name ACTIONWISE. Under Categories a created with the name SINAPWISE. This category will be used to all tags created in the import of the topological database.
- In the **symbols library**, as shown in the figure below, already exist symbols associated with each of the above templates (SW\_) with the same name were created, which allows the direct pasting of objects instantiated with the templates on screens.

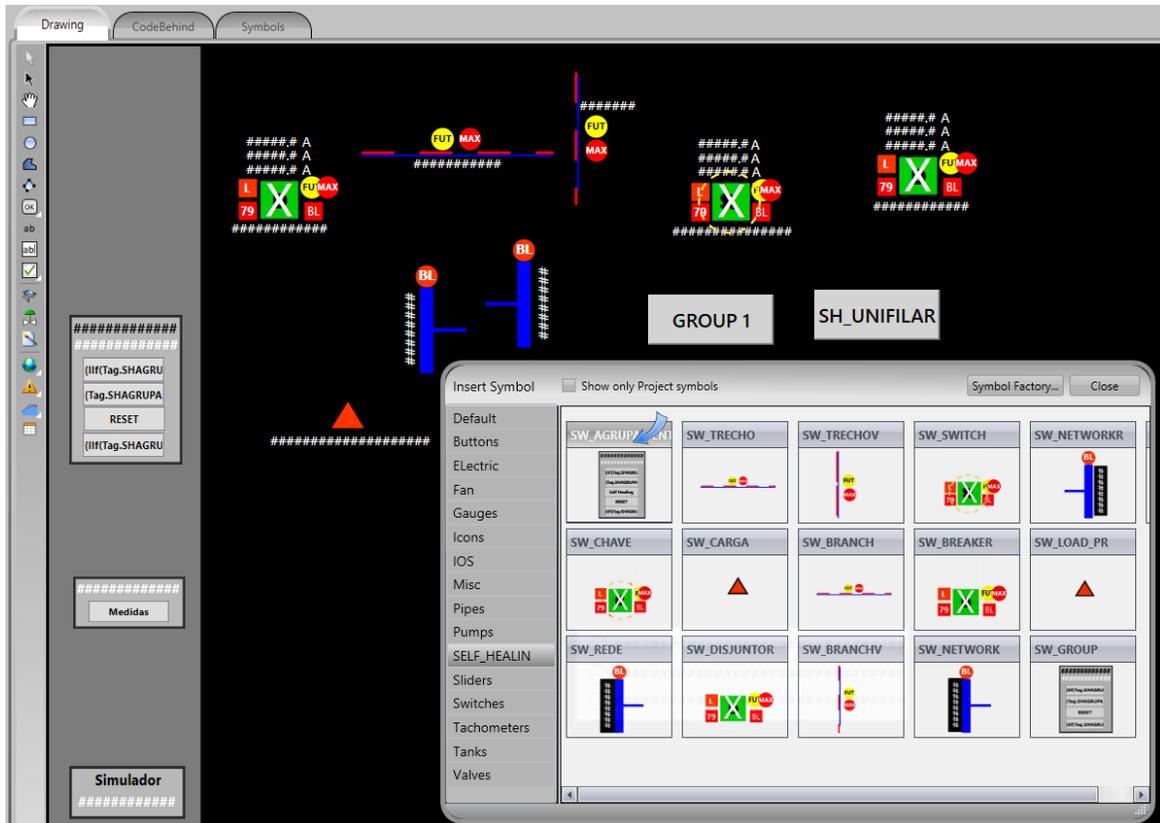


Figure 32 - Symbol library with each network device

- If desired, you can create new or change imported objects, which are only the basics for integrating ActionWise with a network of reclosers, their feeders, branches, and the loads involved.
- In Datasets will be the necessary references the bases of damages, already described.

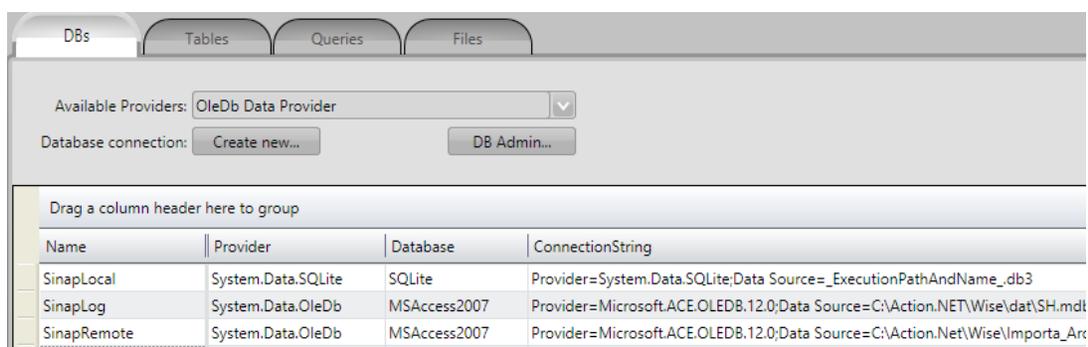


Figure 33 - Datasets

- In Devices, the Channels and Nodes tabs will be the necessary registers for the communication between the SCADA and ADMS modules that form the ActionWise
- In Displays, you can find a basic Home screen, the Switch Command Windows, Self-Healing Maneuvers and Lock Diagnostics.

### 3.2 Creation of a New SCADA / ADMS / FLISR Project

After you change the basic ActionWise project and adapt it to the dealership culture, you just need to create a new project in Action.NET that will automatically include all the basic structures needed to develop a SCADA / ADMS / FLISR application.

The electrical elements themselves are missing, that is to say, the description of an electrical network with various Groupings, electric networks, switches, loads, feeders and the interconnecting parts between these equipment's. This information is in the Model database, created as described in item 2, by the **ActionWiseToolKit** application.

This section describes procedures for object import, tag creation, screen generation, and simplified database assembly used in the SCADA module (dbSh extension).

To import data from the ADMS module, registered in the SinapRemote database, you must use the **SinapWise Import extension**, specifically created for this task.

In Action.NET Extension-type applications run in the engineering tool of the SCADA module to import and export project data, as shown below:

- Go to **Run> Extensions** and the screen of the following figure will be shown;
- On this screen, double-click on SinapWise Import.

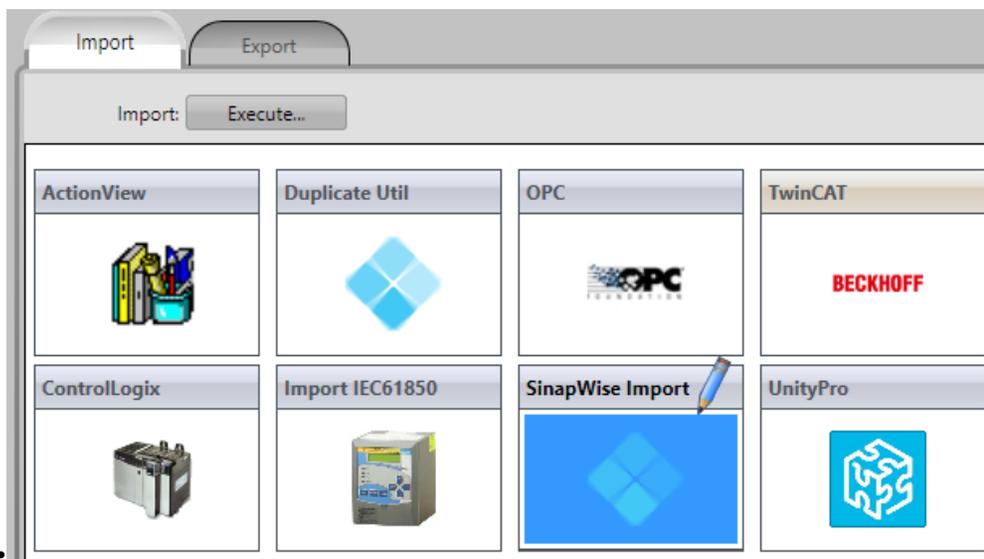


Figure 34 – Extension SinapWise Import

- The execution window of this extension will appear, as shown below.
- In this window, you must choose the Nodes names for the Server and Client channels, which will be necessary for the communication between the SCADA module and the ADMS module. By importing the original Component SinapWise into the project, the nodes will have the names shown below.

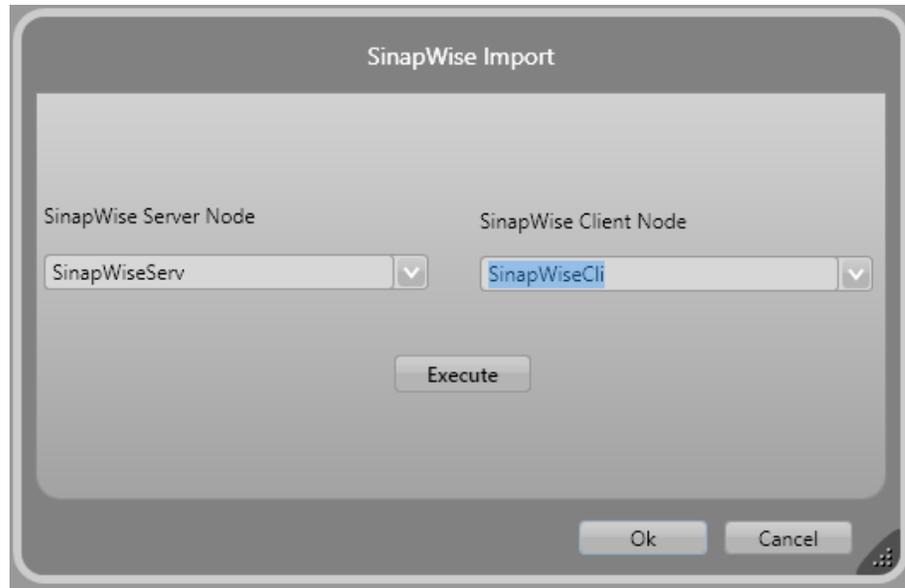


Figure 35 - SinapWise Import Execution Window

- Clicking **Execute** button, the import will start. You can follow the main steps that will be described in the window. After the message "Import Completed" appears, press the Ok button to close the window.
- To ensure that the generation of Tags and Points, made in the import is consolidated correctly, it is recommended to execute a build, going to **Run> Build**;
- In case of an import problem, you can check the contents of a log created in the Projects folder, subdirectory Logs, named ActionNet-dd.mm.aaaa.Log. (Date of the day).

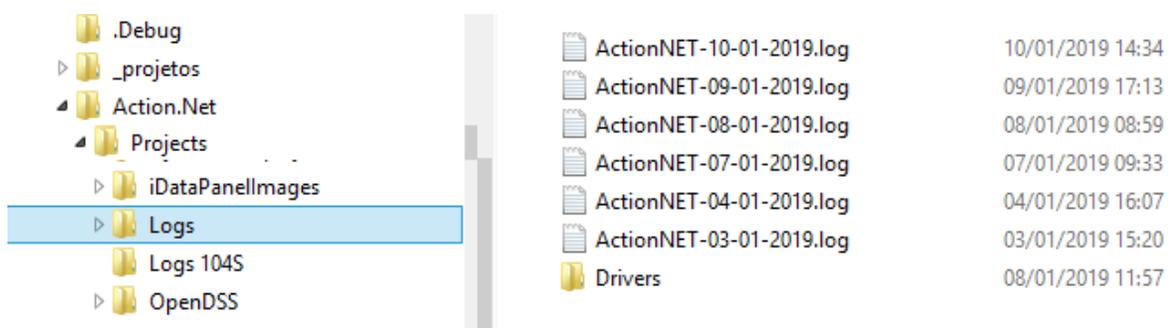


Figure 36 - Logs generated in the SCADA project directory

### 3.3 Procedures Performed on Importing Data from SinapRemote

This section provides a brief explanation of the activities performed automatically during the import of the ADMS data, available in the SinapRemote database, only to better guide the user of

the executed procedures and to substantiate it to make changes or corrections if necessary. It is not an explanation of procedures for the user.

The activities carried out are as follows:

- The first phase is the reading of the SinapRemote database and the simplified recording of data obtained there for the local database with the <project name> and extension "dbSh" (**SinapLocal**).
- The input database is type MS-Access and has its access described by a Dataset with the name **SinapRemote** (available automatically in the DefaultNewProject of ActionWise). If this access is not possible, an error message is displayed and the application terminates.
- The output database is of type "dbSh" (**SQLite**) and will be located in the SCADA Projects directory. Its access will be obtained by the SinapLocal name dataset, in the project dataset table. If it does not exist, it will have its schema and necessary tables automatically created, named project name, followed by ".dbSh", which is copied to the project directory during the Action.NET installation.
- If the local database already exists, it will be completely cleaned, all records deleted, to be populated, again, with the current data from the SinapRemote database. The input base is read out and real-time repositories are created (lists), which are written, to the local output base.
- The tables are read and written in the following order: (in all readings only records that have intStudyId = 0 will be considered)
  - Grouping
  - Equipment Type
  - Type of network
  - Type Switch
  - Networks
  - Bars
  - BarNetwork
  - Equipment
  - BarEquipment
  - Switches
- After this recording, the Equipment repository will be read and for each of its records, a Tag object corresponding to the type of equipment will be instantiated in the Objects table. The instantiated tag will use the equipment **strCode** as its object **tag name**. Duplicate tags **will not be allowed**.

Drag a column header here to group					
...	Name	Type	Level	Domain	Category
*					
21	BRCH_1	SW_BRANCH	ACTIONWISE\GROUP_01	Server	SINAPWISE;
22	BRCH_2	SW_BRANCH	ACTIONWISE\GROUP_01	Server	SINAPWISE;
23	BRCH_3	SW_BRANCH	ACTIONWISE\GROUP_01	Server	SINAPWISE;
24	BRCH_21	SW_BRANCH	ACTIONWISE\GROUP_01	Server	SINAPWISE;
25	BRCH_22	SW_BRANCH	ACTIONWISE\GROUP_01	Server	SINAPWISE;
26	BRCH_23	SW_BRANCH	ACTIONWISE\GROUP_01	Server	SINAPWISE;
27	NETW_01_BK	SW_BREAKER	ACTIONWISE\GROUP_01	Server	SINAPWISE;
28	RECL0_2	SW_SWITCH	ACTIONWISE\GROUP_01	Server	SINAPWISE;
29	NETW_02_BK	SW_BREAKER	ACTIONWISE\GROUP_01	Server	SINAPWISE;
30	NETW_03	SW_NETWORK	ACTIONWISE\GROUP_02	Server	SINAPWISE;
31	NETW_04	SW_NETWORK	ACTIONWISE\GROUP_02	Server	SINAPWISE;
32	BRCH_31	SW_BRANCH	ACTIONWISE\GROUP_02	Server	SINAPWISE;
33	BRCH_32	SW_BRANCH	ACTIONWISE\GROUP_02	Server	SINAPWISE;
34	BRCH_33	SW_BRANCH	ACTIONWISE\GROUP_02	Server	SINAPWISE;
35	BRCH_41	SW_BRANCH	ACTIONWISE\GROUP_02	Server	SINAPWISE;
36	BRCH_42	SW_BRANCH	ACTIONWISE\GROUP_02	Server	SINAPWISE;
37	BRCH_43	SW_BRANCH	ACTIONWISE\GROUP_02	Server	SINAPWISE;
38	NETW_03_BK	SW_BREAKER	ACTIONWISE\GROUP_02	Server	SINAPWISE;
39	RECL0_4	SW_SWITCH	ACTIONWISE\GROUP_02	Server	SINAPWISE;
40	NETW_04_BK	SW_BREAKER	ACTIONWISE\GROUP_02	Server	SINAPWISE;
41	RECL0_8	SW_SWITCH	ACTIONWISE\GROUP_02	Server	SINAPWISE;
42	RECL0_7	SW_SWITCH	ACTIONWISE\GROUP_01	Server	SINAPWISE;
43	BRCH_5	SW_BRANCH	ACTIONWISE\GROUP_02	Server	SINAPWISE;
44	RECL0_3	SW_SWITCH	ACTIONWISE\GROUP_02	Server	SINAPWISE;
45	BRCH_8	SW_BRANCH	ACTIONWISE\GROUP_01	Server	SINAPWISE;

Figure 37 - Each equipment is created with its respective template-type

- If there is no template with the same equipment type name, the corresponding tag will not be created. By the way, this is the way to make **Bars** not be created, since in the current version they are not used in the SCADA module of ActionWise.
- If it is necessary to define more than one template for the same type of equipment, for example for the type of SWITCH equipment, you can define more than one template and in the record of the Equipment table, in the **SinapRemote** database, in the **strExtId** column, you must indicate the template to be used for each piece of equipment. The template should have the name preceded by **SW\_**. In the **strExtId** field, only the template name is used. Ex: The equipment SWITCH normally uses the template SW\_SWITCH. In the case of breakers also classified as equipment SWITCH, we will force use the template SW\_BREAKER by filling in **strExtid** column with word BREAKER. The **strExtId** field is accessible through the "Template" field in the edit window in the ActionNET ToolKit,

- The importer will create, In POINTS table, an entry for each member of objects instantiated in Objects, with member names defined in the protocol. In the Address field will contain the **intEquipId** of the record in **Equipment**, with a subfield to define for the ADMS module which measurement or corresponding state. These fields coincide with existing member names in equipment type templates, such as POS, SUP, CMF, BLF, LKO, CMA, A\_A, A\_B, etc. The figure below shows some points created in points:

Drag a column header here to group					
TagName	Node	Address	DataType	AccessType	Modifi
* RA_574617.MED.A_A	SinapWiseServ	SW:77:A:A	Native	ReadWrite	
RA_574617.MED.A_B	SinapWiseServ	SW:77:A:B	Native	ReadWrite	
RA_574617.MED.A_C	SinapWiseServ	SW:77:A:C	Native	ReadWrite	
RA_574617.MED.V_A	SinapWiseServ	SW:77:V:A	Native	ReadWrite	
RA_574617.MED.V_B	SinapWiseServ	SW:77:V:B	Native	ReadWrite	
RA_574617.STA.POS	SinapWiseServ	SW:77:POS	Native	ReadWrite	
RA_574617.STA.BLF	SinapWiseServ	SW:77:BLF	Native	ReadWrite	
RA_574617.STA.LKO	SinapWiseServ	SW:77:LKO	Native	ReadWrite	

Figure 39 - Automatically created I/O points by SinapWise Import

- The communication module that will read this table is named SinapWiseServer. For details on configuration, field types, etc. see the Protocol Manual, which can be read in Edit> Devices> Channel, choose the SinapWiseServer module in the "Installed Protocols" box and press the Help button. Points for the client module, the SinapWiseClient, will be created too.
- Virtual tags, created to receive ADMS data, such as the Supply that is connected to each segment, whether or not a load is energized, will be allocated on the Client Node. Tags created to receive actual acquisition data, such as the position of a recloser or the measurement of a current will be placed on the Node Server to be sent to the ADMS.
- Actual data acquisition points can be the same or existing tags used for supervision. In any case, they should be configured for the protocols used in their acquisition, independently of those described here. If the tags are other, see in the protocol manual how to define them also in the ADMS module to ensure the matching of the data between the two sets of tags.
- You cannot duplicate tags in ActionWise. Therefore, when you run an import for the second time, nothing will be imported. In the execution log it can be verified that for each tag there will be the warning "Tag already exists => BVA\_21C5 - Not created!"
- Generation of .CSV files with specification of graphic screens corresponding to the equipment and topology described in the database of the Topological Model. (See next item for the use of these files).

### 3.4 Creating Graphical Screens in the ActionWise SCADA Module

In order for ActionWise operators to be able to monitor, in real time, the supervision and monitoring of the distribution networks, the system must create, graphical displays showing the states and measurements in the equipment's and the topology of the networks. These screens allow simplified schematic (non-georeferenced) visualization, the network implemented in the SCADA module and ADMS

In ActionWise DefaultNewProject, as shown in figure 32, there are already *Symbols* to represent each of the objects handled in the SCADA / ADMS integration, with the same name of each of the templates corresponding to the objects they represent.

In ActionWise, an extension named **Import CSV file** is available among the Import tools. Among the features of this extension is the import of ASCII text files with the specification of graphical screen creation with predefined symbols in .CSV format.

#### 3.4.1 .CSV file generation with definition of graphic screens

The Import extension **SinapWise Import**, in addition to creating the SQLite database with the substantive data for the execution of Action.NET run-time, generates files extension "CSV", from the definition of the network topology in the base SinapRemote. These files contain the electrical schemas generated by the Toolkit, with the video coordinates of each associated equipment (Top, left), as shown in the following figure.

It is important for the user to be aware of certain predefined criteria in generating the files:

- The .CSV file will have the name formed by the project name followed by the word "\_Display", and a number just to differentiate it from other files, and with the .CSV extension.
- For each **Grouping** (scheme), information will be generated to create / change a graphic display with the name formed by the prefix "UNSH\_" followed by the name of the grouping. The screen generation will use black background and the dimensions 1680 x 830. The second line in the figure below shows data for creating the screen.

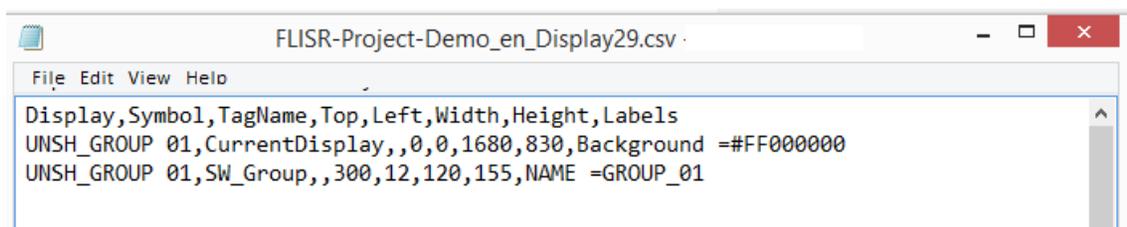


Figure 40 - Example of .CSV file for generating screens

- For each screen a symbol with the name SW\_GROUP is included in the center left of the screen. This is the grouping monitoring and control panel, shown in the figure below.

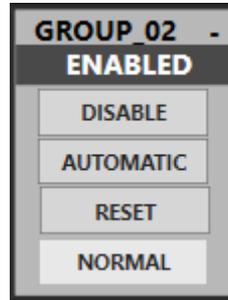


Figure 41 - Grouping monitoring panel

- Screen size, symbol allocation, and layout are proportional to the dimensions and allocations of the corresponding symbols on the original graphics screen created in the ActionWiseToolkit application. In this way, we recommend the creation of the original network take into account that the screen in the SCADA will have these proportions of approximately double the height double. In the case of Schemas that require a higher height, this condition is in mind during the generation of the screen, which will be higher. Maybe we will need manually divide this screen after its import.
- Continuing to generate the other symbols on the screen, a symbol representing a NETWORK will be allocated to each Supply in the original network. There are two of these symbols one SW\_NETWORK, used if the Supply is found in the left half of the graphical screen and another with the name SW\_NETWORKR, if found in the right half of the graph.

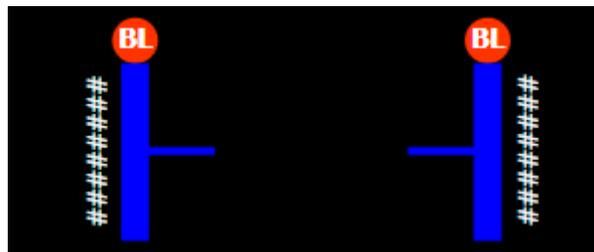
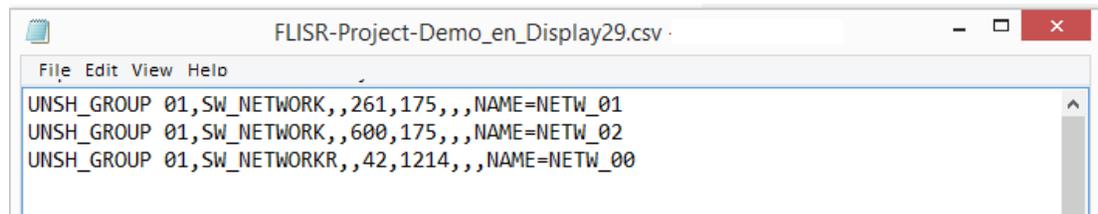


Figure 42 - Symbols for SW\_NETWORK and SW\_NETWORKR (right) networks sources

- For each Recloser object (SWITCH), a SW\_SWITCH symbol will be allocated, which already contains the representation of the switch status, current measurements, switch name, lock status, etc. and the call to display a status and command screen when double-clicking on the recloser symbol. In the case of circuit breakers (from feeders), a different symbol with the name SW\_BREAKER is used. The positioning of the symbols is proportional to the originals in the network.

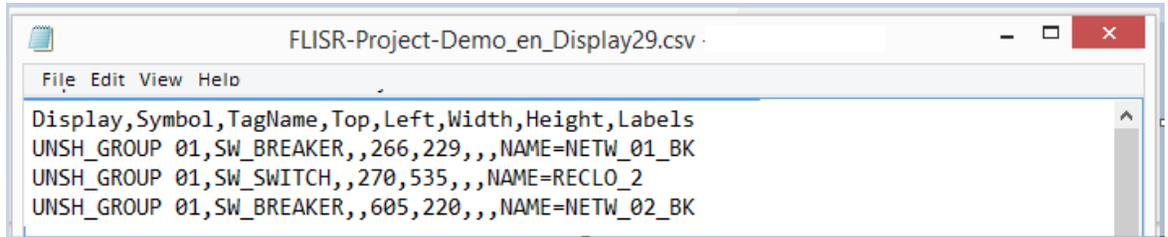


Figure 43 - Symbols for Switches and Switches

- For each section of BRANCH of the schematic registered in the database, a record will be generated for the creation of a Branch Symbol on the ActionWise screen. There are two symbols for Branches: one for horizontal branches SW\_BRANCH and one for vertical stretches SW\_BRANCHV.

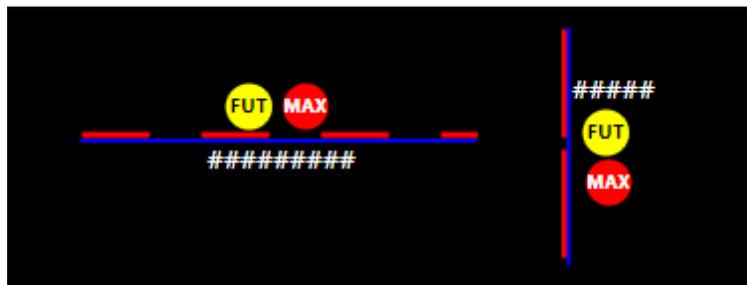
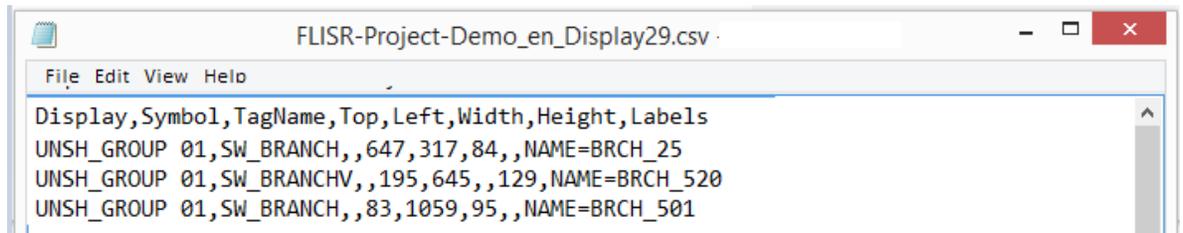


Figure 44 - Symbols for horizontal and vertical segments

- For each **priority load** found in the original network, a SW\_LOAD\_PR symbol will be inserted on the screen to display its fed state or not.
- Only screens and symbols corresponding to objects that were created in the Action.NET project will be generated during the current SinapWise import run. Therefore, when you want to add new equipment, you must do it in the network, by ActionWiseToolKit and after repeating the imports, the corresponding new tags and the new symbols for the screens are generated.

### 3.4.2 Importing the Screen Generation .CSV File

For the import of graphical displays, one must again use an extension developed to read the file with extension CSV: **CSV File**.

Within the engineering environment:

- Go to **Run> Extensions** and the screen of the figure below will be shown;
- On this screen double-click on **CSV File** extension;

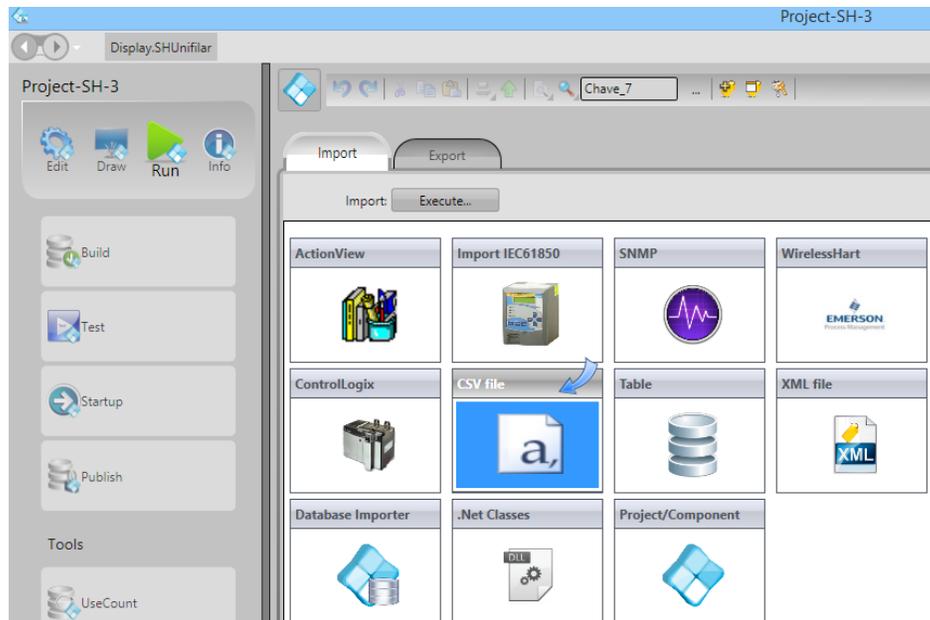


Figure 45 - CSV Importer (used to generate the screens)

- The execution window of this extension will be opened, as shown below.
- In this window, you should choose the CSV file you just created, with the name of the project and the termination **\_Displaynn.csv**. The importer will create this file in the **c:\Action.Net\Projects\** directory, the same directory as the other project files.

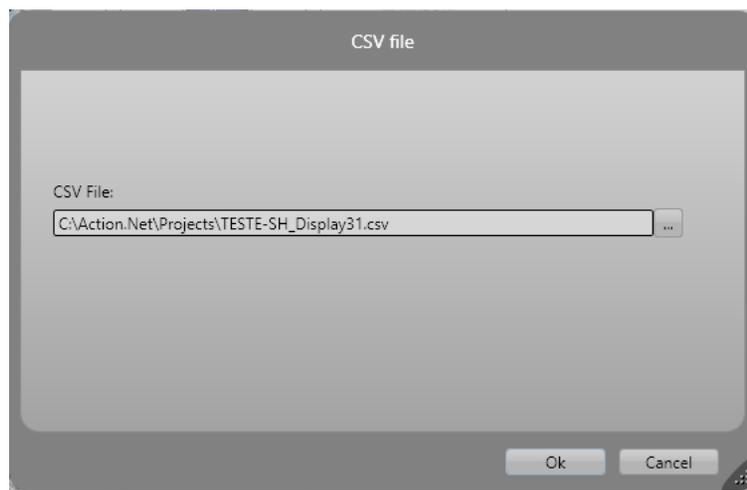


Figure 46 - Importing the CSV file to generate ActionWise screens

- Click Ok, and the import will start, showing in this window information about the progress of the task.
- At the end of execution, the window will close automatically.
- After the import go to **Edit> Displays** and choose the screens that were created to verify the result. Note that all feedstocks, switches, breakers, and priority loads have been included on the screen.
- The layout of the objects will be quite close to what the objects have on the original network graphic defined in ActionWiseToolKit. Maybe the objects will not be correctly adjusted with each other because of the difference in proportion that the symbols in the SCADA and those in ActionWise Toolkit have. The SCADA symbols have more visual information and different dimensions, in relation to the simpler symbols used in the design of the network.
- Open ActionWiseToolKit to check the correct positioning of the symbols on the screen and use the graphical screen to orientate yourself when making the moves to adjust positioning of objects on the SCADA screens. To check which equipment each symbol refers to, click on it to see a window with the name of that specific equipment.

## 4. Real-time interface (HMI)

### 4.1 Functional Screens

Functional screens are ready-made screens in the default project of the ActionWise SCADA module for presentation of events, alarms and historical research.

- System events;
- Current alarms;
- Operation Log;
- Historical data measures screen;
- Real-time and historical trend screens;
- Listing of database parameters

### 4.3 Logic Blocks One-wire Diagrams

As shown in Figure 32, in ActionWise you can easily create one-wire logical blocks diagrams to facilitate operator understanding. We can create one or more screens according to the convenience to facilitate the operation.

In principle, we recommend create one screen for each **Grouping** of networks (with multiple feeders - sources).

Several simple **Groupings** can be included in the same display.

### 4.3.1 Networks (Feeders)

They consist of a power supply (substation feeder) and a set of equipment type: reclosers, branches and interconnected loads. The symbol shown below is for the interconnection point.

Clicking on the Network symbol (TAM\_21F2) shows the network activation / deactivation window, shown in the middle figure.



Figure 47 - Logical Block - Details

By clicking on the switch you can change the mode **Enabled to Disabled** (or blocked), that is, none of the reclosers of this network can be controlled by the FLISR algorithm. In the off mode, a red circle with the symbol "BL" appears beside the bar. The WISE engine module will receive this information, and will consider this network disable for FLISR: no longer will attempt to recover this network in case of occurrences.

**Note:** This blocking of the FLISR function in the logic block, will also be triggered automatically when at least one of the network reclosers (feeder) has:

- Indication of **Hot line works**;
- Automatic Recloser blocked;
- Neutral Relay blocked;

For security, this set (feeder) will not participate in the FLISR algorithm if an occurrence comes.

This automatic locking function is configurable and, at the discretion of the utility's operation, tags may be included or excluded in this consideration. Tag names are set to StartValue of a tag of type SW\_GLOBAL: TAGSFORBLRCALC.

User Custom Type: SW_GLOBAL				
SWGLOBAL members:				
Drag a column header here to group				
Name	Type	Array	Parameters	Description
*				
VIEW	Integer			To change tabs inside CMD_RELIG
TEXT	Text			Parameter to pass text
TAGSFORBLRCALC	Text			Tags for blocking networks
TAGSFORBLACALC	Text			Tags for blocking reclosers
STATE	Integer		Enum=AN_AT...	FLISR General sate: 0=ACTIVE; 1 = NOT ACTIVE
REDE_REF	Reference		SW_NETWORK;	Reference for Network type
NAMETYPE	SW_NA...	20		Conversion for Leaf names to Types -Protocol SinapWise
MWISE	Integer		Enum=AN_N...	Self Healing-Wise logics working
LISTEN	Integer		Enum=AN_IN...	Listtenning mode NOT ACTIVE = 0; ACTIVE = 1
GTW_SIMUL	Integer			Gateway Simulador
COMM_WISECLI	Integer		Enum=AN_N...	Communication SCADAxWise - SinapWiseCli

Figure 48 - Logical Block – Details

### 4.3.2 Reclosers

The Reclosers Switches are represented by the symbol shown below. They include the state, code or name of the key, the three phases of currents measures, and some indications of modes of operation.



Figure 49 - Reclosers - Information

The following figure shows the reclosing blocking indications (BR symbol), self-healing block (BL) and local control mode. The blocking function for self-healing is detailed forward in this manual.



Figure 50 - Reclosers - Blocking Types

The ADMS module continuously monitors the current load on each recloser. The MAX and FUT symbols are respectively shown if the current load exceeds the acceptable maximum (85% of nominal), or if the load is expected to reach this maximum in the future (3 hours later) if the current trend is maintained.

A double-click on the recloser picture shows the control window of the recloser shown and described later in this manual.

### 4.3.3 Branches

The Branches symbolize the interconnection cabling between the reclosers, having as parameters the maximum permissible current. The branches have colors and assume the color of the network symbol that feeds it now. When are not energized, its color is light green.



Figure 51 - Branches

The ADMS module continuously monitors the current load at each stretch. The MAX and FUT symbols are respectively shown if the current load exceeds the acceptable maximum (85% of nominal), or if the load is expected to reach this maximum in the future (3:00 p.m.), if the current trend is maintained.

### 4.3.4 Loads

The one-wire screens do not show all the common Loads existent in all sections. **Priority loads**, which you want be shown on the displays, must be created in ActionWise and represented by a triangle with the name. When energized they appear in Red color. If are not energized are shown only with external lines in green color.

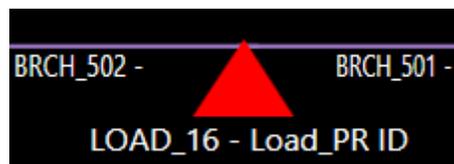


Figure 52 – Priority Load existing in branch

To specify the load as a priority, you must mark this option in the load creation window in **ActionWiseToolKit**. (See item 2.2.8).

### 4.3.5 FLISR General Selection Key

On the home main display or on each one-wire diagram display, as it is considered most convenient, there is a selector switch for switching operation mode between FLISR ON for all groupings or schematics or OFF for all Groupings. The following figures show these two modes. (When off shows flashing red).

On the scheme main screen or on each scheme screen, as it is considered most convenient, there is a selector switch for switching operation mode between FLISR ON for all groupings or schematics or OFF for all groupings. The following figures show these two modes (When off shows flashing red).



Figure 53 - FLISR function states for all Groupings

When the switch goes to the OFF position, state active of all Networks of all Groupings are set to OFF, and this information is sent to the ADMS module, which will no longer attempt to retrieve occurrences.

#### 4.3.6. FLISR Selection Key for Grouping

In the display of each Grouping, a panel can be shown to select modes of operation of each grouping. Individually, in this panel, you can control the ON / OFF status of the FLISR function of each schematic by clicking on the on / off button, as shown in the first two pictures below to the left.

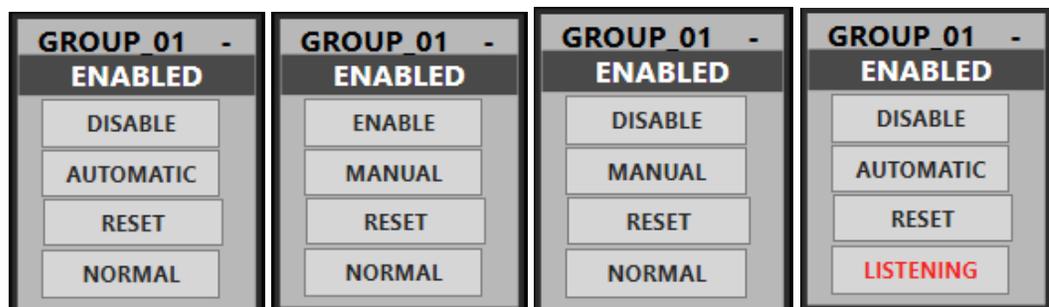


Figure 54 - Panel of each grouping or Grouping

In the last figures on the right, you can see the AUTOMATIC / MANUAL button, which when clicked changes the execution mode of maneuvers recommended by the FLISR function:

- **Manual:** in case of maneuvers propositions, a window will appear and the operator should decide to execute it or not.

- **Automatic:** in case of maneuvers propositions, the maneuvers window will appear, and starts executing automatically. Upon completion of the maneuver, the window closes automatically.

The **RESET** button can be used after the occurrence of maneuvers to normalize the indications of the occurrence of the maneuver on the screen.

Command executions made by the manhole leave a blinking circle around the reclosers maneuvered. To remove these circles, press the Reset button.

When in **SIMULATOR** mode this button also restores the normal states of all cluster keys, as well as the indication of communication failure and the flashing lines next to the sections not fed by their normal supply sources, if there is one.

The **NORMAL** button is only used in real-time production mode (not simulated). This button allows change to **LISTENING** mode, preventing the actual sending of commands to the field. When the system is in LISTENING mode, the word is red.

#### 4.4. FLISR operation log screen

This screen shows the "log" of operations created by the ADMS electrical calculations module.

The information is the date of occurrence, the sector of the application that generated the log, and the description of the occurrence.

When button **FLISR Messages** is clicked, in real time, the log display is shown with the most recent 500 log records.

If desired, you can choose the historical presentation by clicking the "**History**" option and entering the date and time range for the search.

The project periodically deletes log messages generated more than 60 days from the current date.

Runtime		Summaries		Logs		Charts	
<input type="radio"/> Runtime	<input checked="" type="radio"/> History	Initial Date	01/30/2019 12:00	Final Date	01/30/2019 11:59	Search	
Date	Source	Event					
1/30/2019 2:24:31 PM	Topology	Demand variation In the network: 0,070 MW					
1/30/2019 2:24:31 PM	Topology	Variation In the number of consumers In the network: 10					
1/30/2019 2:24:31 PM	Topology	RECLO_60 was closed					
1/30/2019 2:24:31 PM	Topology	Demand variation In the network: 0,080 MW					
1/30/2019 2:24:31 PM	Topology	Variation In the number of consumers In the network: 12					
1/30/2019 2:24:31 PM	Topology	RECLO_90 was closed					

Figure 55 - Historical log with start and end date / time

FLISR Project Demo

Action.NET © SPIN User: SUPER

Back Home Alarms Events Horn Operations Tags Events Measures Trends Reports Filter Export CSV

Runtime History

Date	Source	Event
1/30/2019 2:24:31 PM	Topology	NETW_02_BK was closed
1/30/2019 2:24:31 PM	Analysis	RECLO_90: Closed switch indication
1/30/2019 2:24:31 PM	Topology	Demand variation in the network: 0,016 MW
1/30/2019 2:24:31 PM	Analysis	RECLO_71: Closed switch indication
1/30/2019 2:24:31 PM	Topology	Variation in the number of consumers in the network: 200
1/30/2019 2:24:31 PM	Analysis	NETW_05_BK: Closed switch indication
1/30/2019 2:24:31 PM	Analysis	RECLO_500: Closed switch indication
1/30/2019 2:24:31 PM	Analysis	NETW_00_BK: Closed switch indication
1/30/2019 2:24:31 PM	Analysis	RECLO_3: Closed switch indication
1/30/2019 2:24:31 PM	Analysis	RECLO_10: Closed switch indication
1/30/2019 2:24:31 PM	Analysis	RECLO_41: Closed switch indication
1/30/2019 2:24:31 PM	Topology	RECLO_2 was closed
1/30/2019 2:24:31 PM	Topology	Demand variation in the network: 0,014 MW
1/30/2019 2:24:31 PM	Topology	Variation in the number of consumers in the network: 100
1/30/2019 2:24:31 PM	Analysis	RECLO_9: Closed switch indication
1/30/2019 2:24:31 PM	Analysis	RECLO_7: Closed switch indication
1/30/2019 2:24:31 PM	Analysis	RECLO_8: Closed switch indication
1/30/2019 2:24:31 PM	Analysis	NETW_04_BK: Closed switch indication
1/30/2019 2:24:31 PM	Analysis	RECLO_4: Closed switch indication
1/30/2019 2:24:31 PM	Topology	NETW_01_BK was closed
1/30/2019 2:24:31 PM	Analysis	NETW_03_BK: Closed switch indication
1/30/2019 2:24:31 PM	Analysis	NETW_02_BK: Closed switch indication
1/30/2019 2:24:31 PM	Analysis	RECLO_2: Closed switch indication
1/30/2019 2:24:31 PM	Analysis	NETW_01_BK: Closed switch indication
1/30/2019 2:23:10 PM	Topology	Reconfiguration not started: there are no switches with lockout signal activated
1/30/2019 2:23:10 PM	Topology	Reconfiguration not started: there are switches executing the reclosing cycle
1/30/2019 2:23:10 PM	Topology	Reconfiguration not started: there are no switches with lockout signal activated

Figure 56 - Log with last 500 events

## 4.5 FLISR Event Reporting Screen

In this screen, you can visualize each occurrence of key opening and intervention of the FLISR algorithm. On the left side of the screen, a list shows the last occurrences of FLISR maneuvers. By clicking on one of these lines, the detailed data of the occurrence appears in the main part of the screen.

Still in the list, if the operator has learned the cause of the opening of the switch / breaker, he must choose one of the existing phrases in the "Reason" box, and this reason will be permanently stored in the list.

In the central part of the screen, in red, are the data of the opening moment, the number of consumers supplied and the load supplied, before and after the occurrence. In green are the data on the maneuver of recovery of the load, that is to say, how much of consumers and load were recovered with the maneuver. The blank right block shows the final situation after the recovery maneuver with consumers served and load supplied, before the complete manual recovery of the sector.

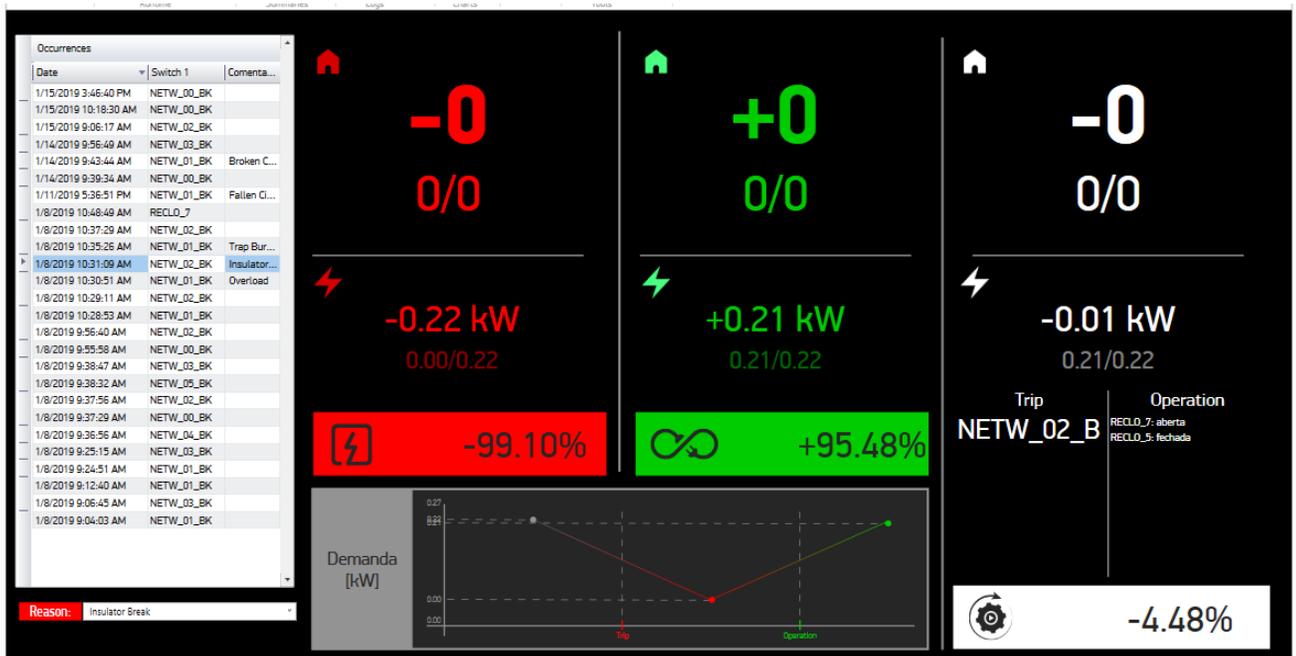


Figure 57 - Report of occurrence

## 4.6 Operating Windows

### 4.6.1 Breaker Command Window

In order to allow the operator to control the opening and closing of breakers and reclosers, as well as locking operations, a control window is available, shown in the figure below. At the top is the code and description of the breaker.

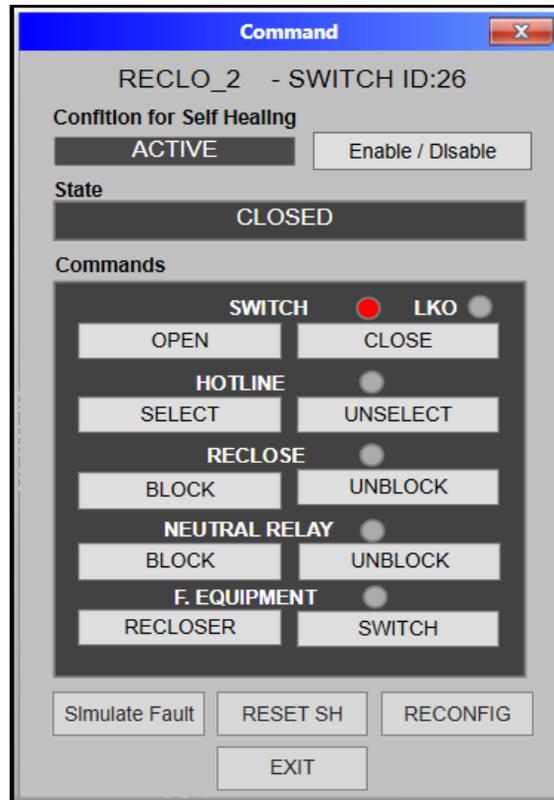


Figure 58 - Breaker Command Window (Recloser)

**Condition for Self-Healing** - The text ACTIVE appears in this box if the recloser does not present any blockage that prevents it from participating in recovery maneuvers. If the recloser is in Locked (or Disabled) mode for Self-healing, this window will be as shown below, with the indication of red emerald state.

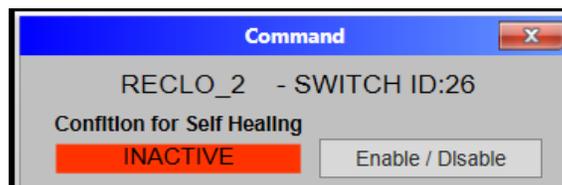


Figure 59 - Locked Recloser

Clicking on this frame, with information about locking for FLISR, will show the Blocking Diagnosis Window shown in the next item).

The **Disable** button on the right side of the blocking state allows the operator to **manually deactivate / activate** this recloser for self-healing operations. If this **manual lock** is set, and there are some possibility of recovering from occurrences, the recovery algorithm will not consider this recloser in the maneuver proposition. This prevents the Breaker from automatic operations.

**State** - Displays texts OPEN or CLOSED, depending on the current state of the key.

**Commands** selection - In this box are a set of several command buttons for the recloser.

- **Open / close** the switch.

- **Select / De-select** the blocking operations per **Hot Line** works.
- Block / Unblock or **auto reclose**
- Block / Unblock **neutral relay**
- Change equipment function between **recloser / disconnecter**

At the bottom of the command window are buttons for SIMULATE FAULT (See in Simulator Mode description below), RESET SH, RECONFIG, and EXIT.

**RESET SH** - Resets this breaker/recloser, i.e. if in simulated mode, returns its state to normal mode and, in any mode, resets the OPSH flags (yellow circle after operation) or CFL (red circle if operation canceled) .

**RECONFIG** - Allows the operator to trigger the network reconfiguration algorithm by opening the breaker, without other reasons such as the reclosing Lockout operation. It is also possible to use this button to request a reconfiguration, even if the circuit breaker is not open. The reconfiguration algorithm will look for a new topology that is the most optimized among possible different ones from the current one. If algorithm find this topology, it will propose a maneuver to get in there.

#### 4.6.2 Blocking Diagnostics Window

In the Recloser Command Window, by clicking on the **Condition for FLISR** (Self-Healing) frame, the self-healing diagnostics window is displayed, showing the states of several field points of the recloser, which are used to generate the signal **Automatic** locking for self-healing of the recloser. Frames with black background and white text are in normal condition, allowing the use of this switch in recovery maneuvers. The tag names that appear with **yellow background** and black text are reasons for blocking. Will appear with the background in **red** the points that cause disabling FLISR over the whole network, when actuated.

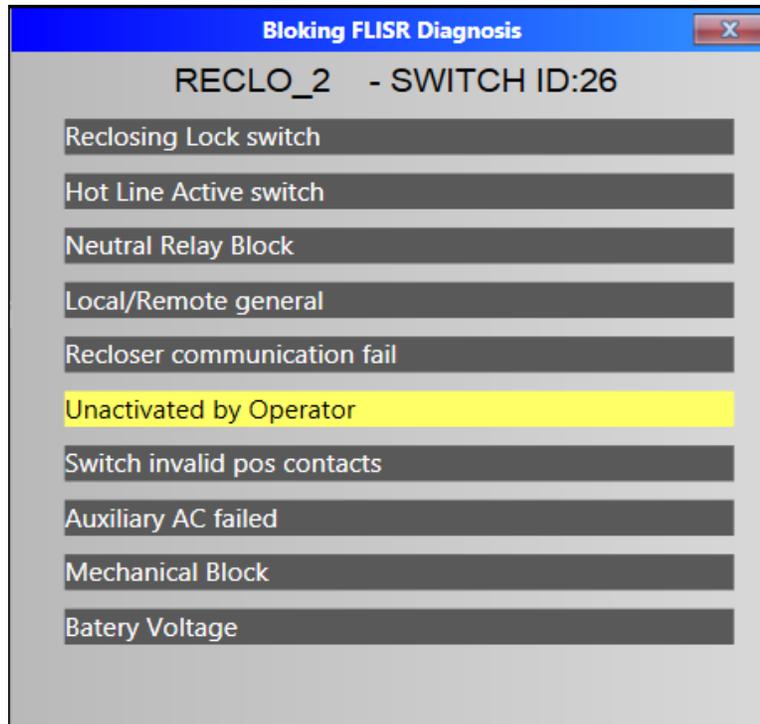


Figure 60 – Blocking FLISR Diagnostics Window

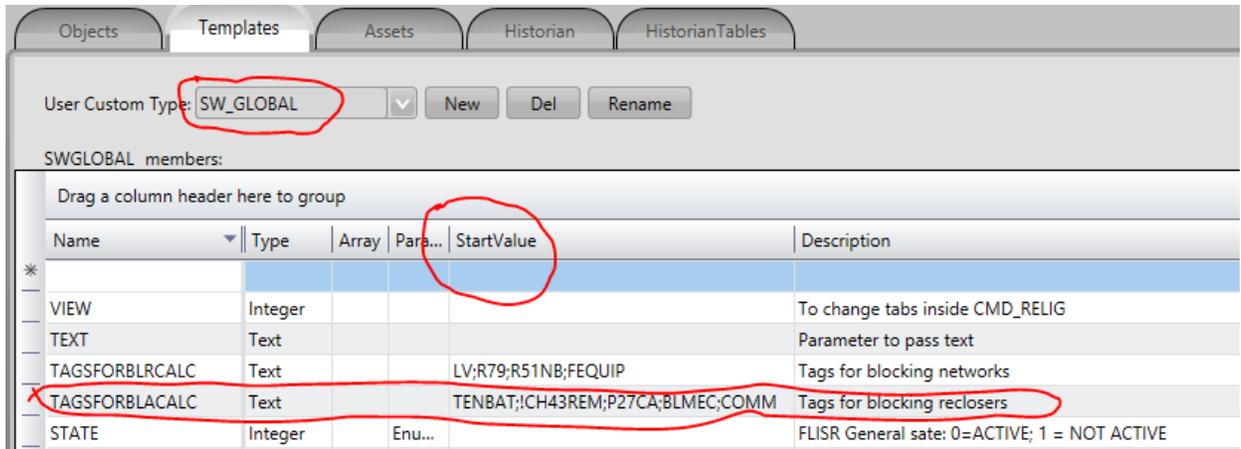
The configuration of the tags and names that appear in this window, is possible in **Edit / Scripts Class** tab, choose the **SH\_Diagnose** class item.

**Self-Healing Blocking (BL)** - Locking a Recloser to participate in a recovery maneuver can occur for three reasons:

- The **Manual** Lock made by the operator by clicking on the Disable button of the command window, described in item 4.6.1;
- **Automatic** Blocking, when any of the tags listed in the above window is active in the field, and also appears in yellow;
- States that cause **Network blocking** (the first three in the above window), as already described in item 4.3.1;

In Manual and Automatic cases, the blocking is done only on the recloser, so that it is considered as **non-existent (and closed)** and is not considered as maneuverable in the choice of maneuver alternatives to obtain a recovery of loads in case of an occurrence of circuit opening.

The configuration of the tags chosen for this deactivation is possible in the installation of the system according to the preference of the operation of the concessionaire. Tag names are set to StartValue of a tag of type **SW\_GLOBAL: TAGSFORBLACALC**. This is a string consisting of the names of the tags (leafs in the templates) separated by ";". Tags whose status "0" (or off) should turn off Self-Healing, should be preceded by "!" In the definition of this text.



Name	Type	Array	Para...	StartValue	Description
VIEW	Integer				To change tabs inside CMD_RELIG
TEXT	Text				Parameter to pass text
TAGSFORBLRCALC	Text			LV;R79;R51NB;FEQUIP	Tags for blocking networks
TAGSFORBLACALC	Text			TENBAT;!CH43REM;P27CA;BLMEC;COMM	Tags for blocking reclosers
STATE	Integer		Enu...		FLISR General sate: 0=ACTIVE; 1 = NOT ACTIVE

Figure 61 - StartValue Column

### 4.6.3 FLISR (Self-Healing) Maneuver Window

The Maneuvering Window aims to present a recommended maneuvering plan in case of spontaneous openings of reclosing switches.

After opening a switch, the automatic reclosing cycle starts. The ADMS module will analyze the occurrence and, if it detects any of the conditions bellow listed, it may generate, if possible, a plan of maneuver. The conditions for triggering the reconfiguration are:

- If, after opening, automatic reclosing fails and the recloser enters the **Recloser Locked Out** condition;
- More than two openings and closures of the recloser occur within a predefined time interval and the **recloser remains open**;
- The operator manually presses the **RECONFIG** button, shown above in the command window;
- ActionWise detects that a breaker opening or recloser occurred due to **protective action** and has expired the maximum time considered for a reclosing. (See details in the next item).

When the algorithm decides to perform a maneuver, the ActionWise Server communication module receives from ADMS this plan, that is, a set of remote commands requests to change topology, and places them in this maneuver window, automatically presenting it in the screen.

If the **Automatic** execution option is selected the commands are sent to the reclosers.

The command window has the following fields:

**Grouping identification** - Its initials and identification number.

**Maneuver Number** - A sequential number, which starts at 1 every new activation of the ADMS module.

**Date** - The date and time ithe maneuver started

**Steps** - The number of commands that make up the maneuver.

**Commands** - Each command, indicating if the switch is OPEN or CLOSE and the target switch code.

**Execute** - This is a button to trigger the execution of the maneuver in case the system is in manual execution mode.

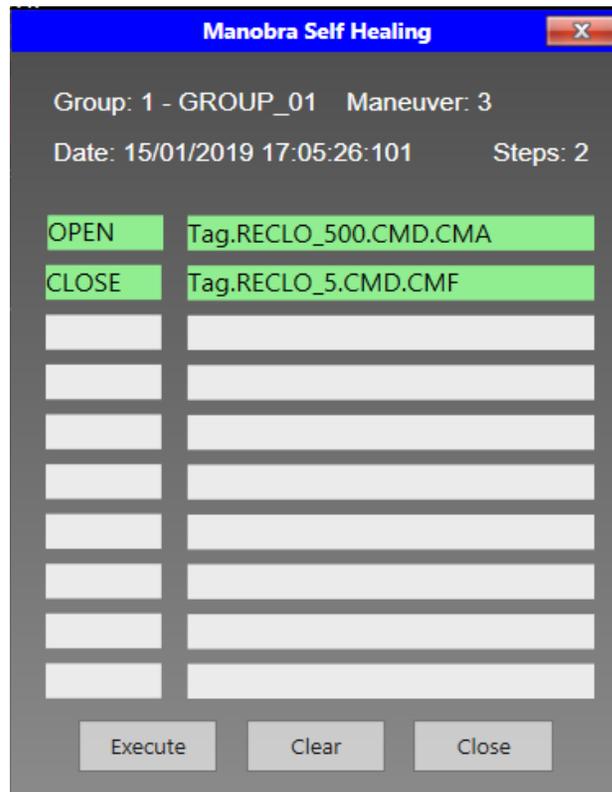


Figure 62 - Suggested FLISR Maneuver Window

When the execution of the maneuver starts, the background of the line corresponding to the sent command turns light **yellow** at the time the command is sent. It stays in this color until the correct signalization of the new state of the switch arrives, turning to the **green** color. If the signaling does not arrive in acceptable time, 10 s, the description of this command is **red**.

After successful execution of the commands, each of the controlled reclosers will show a flashing dashed circle in **yellow** to draw the attention of the operator of the executed operations. This circle becomes visible when the **OPSH** (Self-Healing Operation) tag goes into the actuated state. This action is done by internal code of the self-healing window, when detecting the successful signaling of the execution of the command.

If a command does not succeed, i.e. the code sent a command to recloser and in a timely manner (10s), the correct signaling of the new switch, state is not received, the **CFL** (command failure) tag is turned "one". This causes a red circle to flash around the recloser symbol. The system cancels the maneuver and this step and the followings will no longer be executed.

**Clear** - This button changes the state of the OPSH and CFL tags to zero by making the circles disappear.

**Close** - Button to close the window.

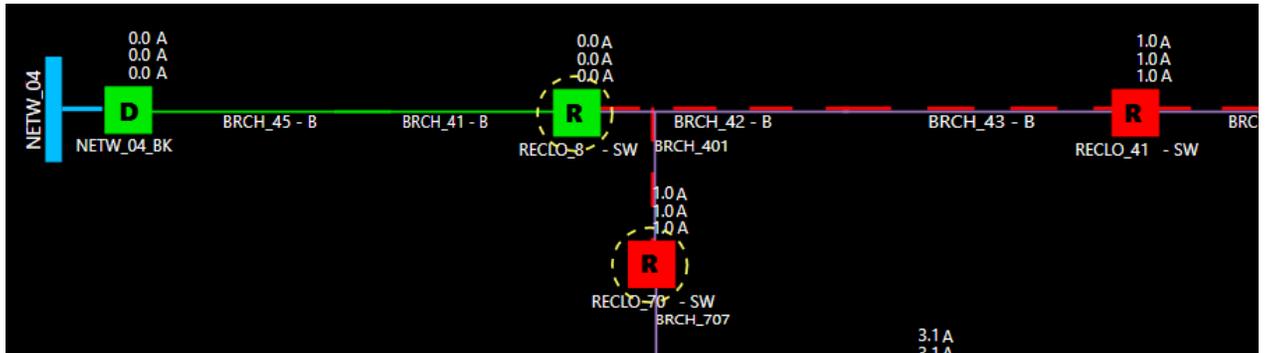


Figure 63 - Shows reclosers that were operated by the FLISR (Self-Healing)

#### 4.6.4 Maneuver started by protection detection tripping

For cases of circuit breakers or reclosers where the lockout signal is not available or if the multiple reclosing criteria cannot be used for triggering recovery maneuvers, is available an algorithm for the detection of openings by protection.

To use this option, you must choose this feature when you register the circuit breaker or recloser in the **ActionWise Toolkit** application. The figure shows the window used for this registration. You should mark the item "Protection Trigger" and set Delay as the time in seconds that the algorithm must wait to make sure that there will be no reclosing.

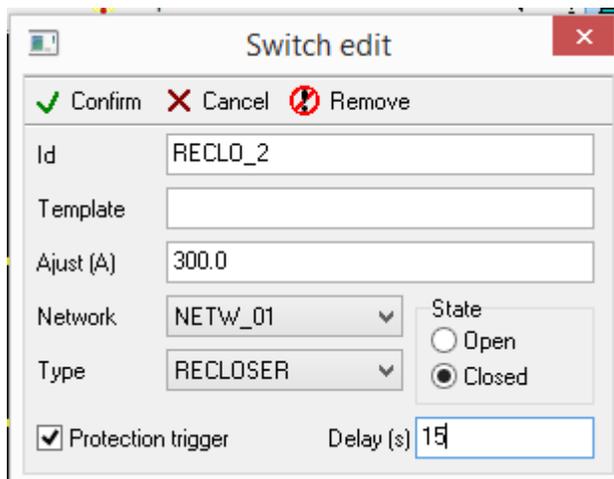


Figure 64 - Delay Protection Trigger Option

These protection signals must also be registered in ActionWise and should be considered as the cause of the opening. These signals must be available on the communication channel between ActionWise and the Reclosers (or and center that provides the real time recloser data).

The algorithm will only generate a trigger signal for the recovery (TRP signal), in the case of:

- The opening of the circuit breaker or recloser has occurred;
- At least one of the protections registered on the circuit-breaker or recloser occurs;
- Reclosing time expires without a reclosing;
- The currents in the circuit breaker or recloser remain at or near zero.

The PRT-Occurrence of Protection signals and the TRP-Trip signal for protection will be sent to the ADMS module, to prepare anew maneuver plan, if possible.

#### 4.6.5 Maneuver Failure

When the SCADA module receives a maneuver from the ADMS module, it places each of the commands in the FLISR Maneuver window and shows this in the one-wire diagram display properly. If it is in **manual** execution mode, it waits for the click of the execute button in this window. If it is in **automatic** run mode, it immediately starts executing commands.

Commands are sent **sequentially** and the window waits for each confirmation flag before sending the next command. If one of these signals is not received in a predefined time, typically 10s, the SCADA module sends event messages to the operator, indicating the No Operation Success, and marks the existing CFL tag for each recloser. This information will be sent to the ADMS module by the CFL actuated field of the Key equipment in question.

The maneuver in question will be canceled. The remaining steps will not be performed.

## 5. ActionWise as Simulator

**ActionWise** has an integrated simulation module, allowing simulation exercises to occur in the set of equipment, networks and reclosers controlled by the system. These simulation features are implemented in part by the ADMS module and partly by the SCADA module.

After completing the ActionWise configuration, before testing it in the field, in the real-time environment, you can run it in the lab environment in simulator mode.

In this case, as the video available on Youtube shows, the ADMS module simulates the normal state of the system, making small variations on the measurements that represent its loading. In this simulated environment, the user may cause various fault situations and verify how the system will react.

The simulator features are presented below and the video above allows a clear understanding of their operation.

### 5.1 Simulator Operation

In the current version, the simulation facilities are implemented in the same real-time supervision project.

A key on the home screen allows the operator to enter or exit the simulation mode. A panel with the smaller key is also displayed on all one-wire diagrams displays.

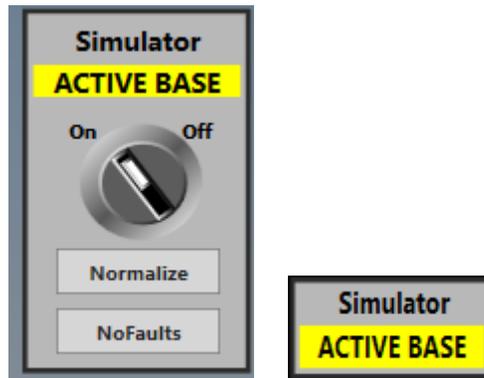


Figure 65 - Simulator module on / off

Clicking on this key to switch to **Simulator** mode following actions will take place:

- a) A tag indicative of the mode is changed, this change being sent by the SCADA module to the ADMS module.
- b) In the simulation control panel key it is shown in yellow the indication of the simulation mode
- c) All communication channels with the field are disabled.
- d) In simulated mode entry, all keys and reclosers have their states changed to the normal states, originally defined. The **Normalize** button, if pressed also performs this reset, in addition to returning to the simulated base state.
- e) From this moment, the ADMS module sends to the SCADA current and voltage measurements, from reference values obtained from pre-registered curves to a situation of normal operation of the system (base case).
- f) Only to give an idea of changes in measurements are they altered by the addition or decrease of insignificant values.
- g) These measurements, as well as the state of the keys, continue to be sent by the SCADA to the ADMS, as if they were the actual field values.
- h) The ADMS on receiving them can redo the currents calculation and resend the new values to the SCADA, setting the normal state of the system. The values sent are based on the current topology of the network, and the reference values (base case) are no longer sent.

## 5.2 Simulation Interventions

Some intervention operations are planned to simulate occurrences and to observe the system response in these situations.

### 5.3 Switch Opening

This operation will simulate a fault occurrence in the downstream portion of a key.

- Select the switch on the screen and left click on it.
- The switch command window appears. Then click the **Simulate Fault** button at the bottom of the window (this button is not shown in Normal Supervision mode - Simulator Disabled).
- With this action, the Simulator causes the opening of the switch by changing the state of the signaling tag of the same (**POS**) to open. It also changes the LKO tag, which signals the **Locked Out** state of the switch to acted after 5s.
- These changes are sent from SCADA to ADMS. The ADMS detects these changes by sending the measured zero values and indicating the energies of the segments affected by the opening.
- The ADMS will analyze if there has been an occurrence with recloser lockout and will decide whether to create a list of appropriate maneuvers, aiming at reestablishing the largest possible number of loads.

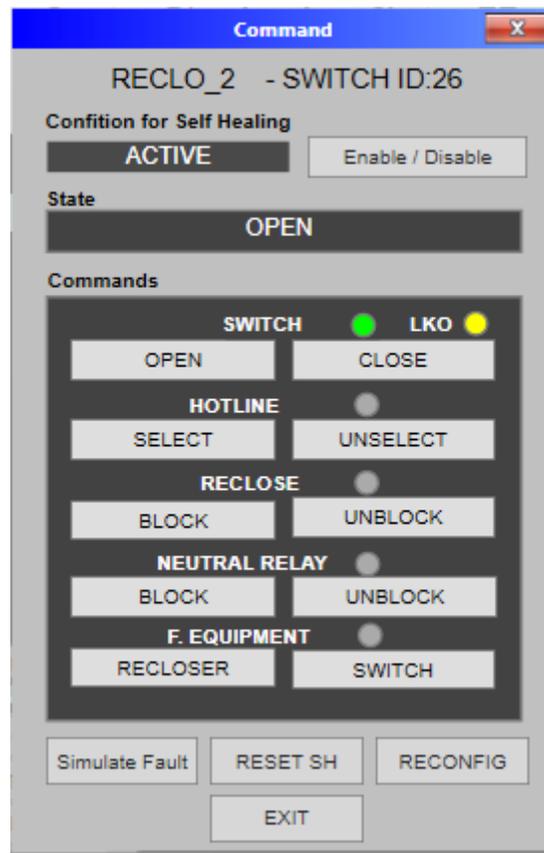


Figure 66 - Window with Simulate Fault and Lockout actuated

## 5.4 Switch Blocking for the FLISR Function

For each switch there is a tag (BLF) calculated as an "OR" of all signaling faults or locks that are considered to be impeding the switch to be handled automatically by the FLISR (self-healing) functionality. You can simulate this situation.

- Select a switch on the screen and click the left button on it.
- The switch command window appears. Click on the Status box to show the condition window for lockout. The state of one of these tags is changed to the value ACTUATED.
- If at least one condition of the conditioning tags is ACTUATED, the Blocking status will change from ACTIVE to Blocked.
- This block state change will be sent to the ADMS module, which will no longer consider the key in the FLISR (self-healing) maneuvers.
- On the main screen next to the switch symbol, a **blocked status** indication for FLISR (self-healing) will appear.
- The Disable button allows the operator manually block the Breaker manual operation. In addition to the field causes, this mode also causes the recloser to lock for self-healing.

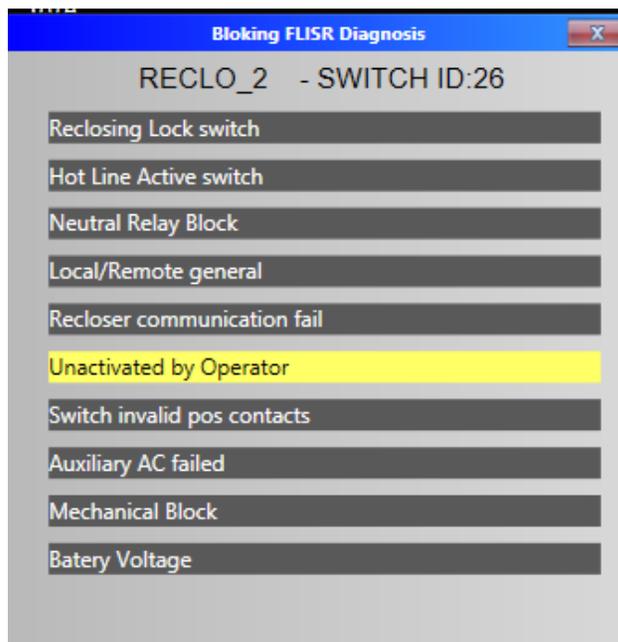


Figure 67 - Window with blocking conditions

## 5.5 Change of Maximum Current

For simulation of increase / decrease maximum current capacity in a circuit **breaker** or **branch**, you can change these parameters on the one-wire diagrams displays.

- a) On the screen, choose the switch, and over the field of Current, or branch whose measurement parameter, you want to do a change and click with the left mouse button on it.
- b) A window will be displayed showing the current value of the parameter Maximum Current for this equipment. This value can be changed by typing a new one. Click the Ok button to close.
- c) The new value will be placed in the maximum parameter tag and this change will be received in the ADMS module, which will recalculate the state estimate, and if necessary provoke preventive maneuvers to establish a permissible situation.
- d) This feature does not exist in Normal Mode (simulator disabled).

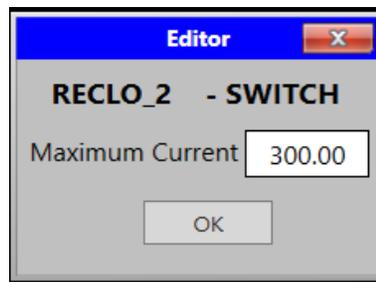


Figure 68 - Change window of the maximum current of a section

## 5.6 Return to Base Case (Normalization)

After any simulations and state changes brought about by interventions for the simulation, the system enter in another Mode: Simulator Active Changed. The operator can return to the base case, that is, return to the normal situation of measurements, parameters and equipment status:

- a) On the main home display, press the **Normalize** button below the Normal / Simulate mode selection key;
- b) The mode will return to Simulator Active Base. This is, the simulator will reset the positions of the keys to the normal situation, and the ADMS module will send the set of reference measures again (base case).
- c) This feature does not exist in Normal Mode (simulator disabled).

## 6. ActionWise Simulation - Network Simulator

In the case of using ActionWise integrated with the customer's SCADA, before the field tests, it is possible to test the software and project configuration in its real-time mode, integrated with a communication gateway that in turn communicates with the **ActionWise Simulator standalone**

application. This application reads the same network database used by actual project and simulate currents, loads, sources and switches states using load flow calculation.

## 6.1 The Simulation Environment

The purpose of this test is, without using the customer's SCADA, executes tests with ActionWise / ADMS / FLISR in a real-time environment through a Gateway that will work seamlessly simulating customer software.

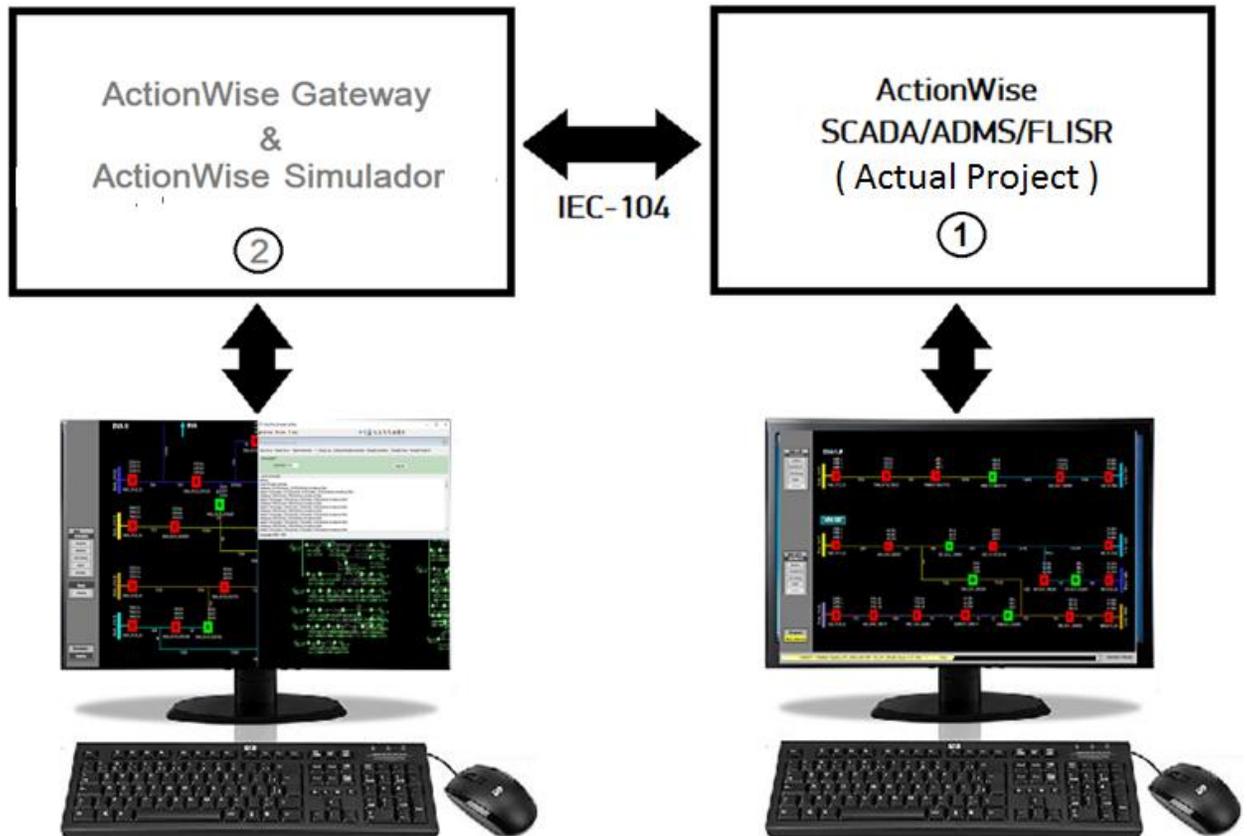


Figure 69 - ActionWise Simulation Environment

As shown in the figure above, in this test there are two environments:

- **ActionWise / SCADA /ADMS / FLISR running in real time.** This software shall receive information read from the field by the customer's SCADA using an electrical communication protocol, which may be IEC-60870-5-104 or DNP3.0. Whenever there is a change in the system topology or in its load, the client's SCADA, as a server in the selected protocol, sends the information to the ActionWise / ADMS / FLISR. This system executes the topology processor, the state estimator and the FLISR function and, in response, sends maneuvers to isolate the problem and recovery the system. In the simulation environment, using the ActionWise Network Simulator integrated with ActionWise / Gateway,

- **ActionWise / Gateway** and **ActionWise Simulator** creates a simulated field with data very close to the real one. In this environment there are two applications running:
  - a) **ActionWise Simulator**: It builds a real time database from actual project network database ActionWise, and is a complete tool for simulating loads, sources, currents and network failures, based in real time load flow calculations and topological processing. This software communicates with the ActionWise Gateway that translates all faults in IEC-60870-5-104 messages and sends them to the environment ActionWise SCADA/ADMS/FLISR. The ActionWise Network Simulator has a toolbar that allows user simulate different faults manually or automatically, as shown below.

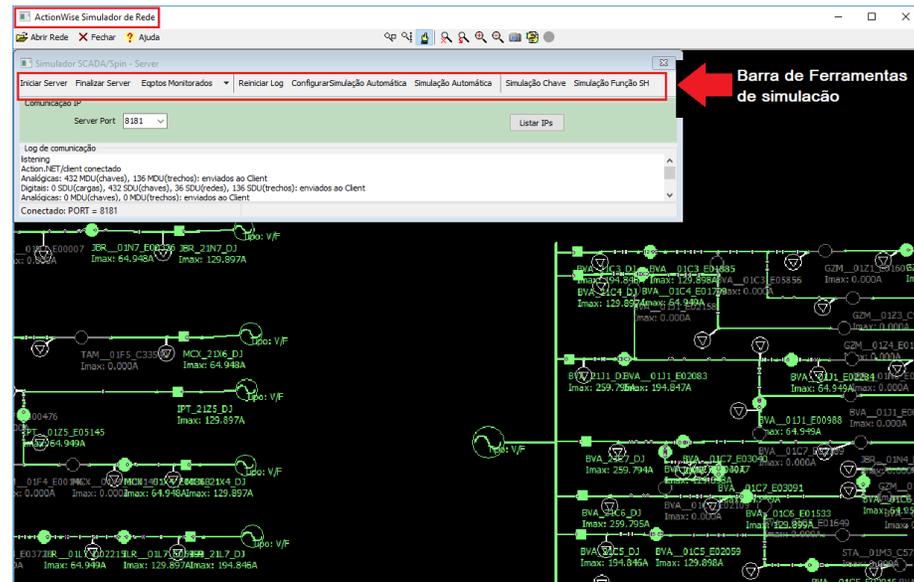


Figure 70 -ActionWise Toolkit Simulating the Network

- b) **ActionWise Gateway**: This module is the ActionWise/SCADA ADMS / FLISR itself using the same Action.NET project, but configured as a gateway. In this mode, their displays reflect the state of the process, but the FLISR algorithm is not executed. The program communicates with the ActionWise Network Simulator and translates all analog and digital variable changes into IEC-60870-5-104 messages and, as an IEC-104 server, sends these messages to ActionWise FLISR real project, which in response sends commands to make the maneuvers defined.

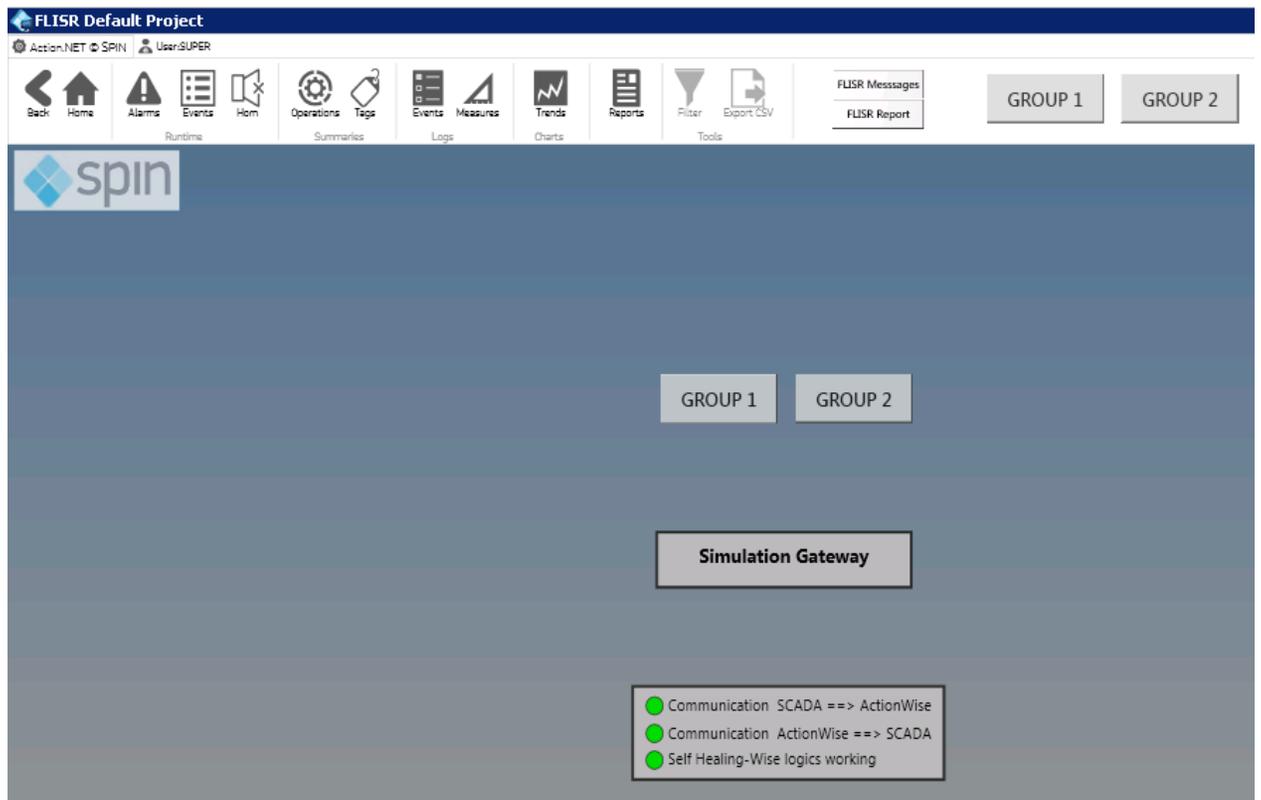


Figure 71 -ActionWise Simulation Gateway

As we will see below, it is possible to do several system crash tests and analyze how it will behave. More importantly, if the **ActionWise / SCADA / ADMS / FLISR** real-time environment works correctly in this test and does not work when integrated with the client's SCADA, we will, in a easily way, found out possible mistakes in all integration.

Another great advantage of this test environment is that you can automate failure events and thereby stress the system by verifying their actual performance.

### 6.1.1. Creating the Action.NET Gateway project

The Gateway project is created from the ActionWise project, by performing the following actions:

- 1) Copy and paste the ActionWise project, to create another project, changing its name, for example by adding the suffix \_GTW (Gateway) to the original project name.
- 2) Access the new project and edit the template SH\_GLOBAL, changing the attribute StartValue from the GTW\_SIMUL member to value 1, indicating this is a gateway.
- 3) Go to Edit/Device, tab Channel: in the SnapWiseServer channel, change the attribute InitialState to RESERVED. In the IEC8705104M, change to RESERVED.
- 4) In the original project, copy and paste in an EXCEL file all the points from the IEC104M node (channel that in the original project receives data in real time from the Operation

Center). If this communication is made by more than one channel, repeat this for each one of them.

- 5) In the original project, in the IEC104 Masters, change the addresses and IP ports so that they connect to the addresses where the Gateway Simulator will be executed.
- 6) In the new project new\_GTW, remove all the IEC104S channel points that were from the original project, because they will not be used in the simulation.
- 7) In the EXCEL spreadsheet, change for all the points the name of the node from IEC104M to IEC104S. In the same file, also change the points with Access Type **Write** to Access Type **ReadWrite104**. In addition, the points with Access Type **Read** to **ReadWrite**.
- 8) The points obtained in the EXCEL file will create in the Gateway the channel that will provide the real time data (simulated) to the original ActionWISE object. Paste all the points from the EXCEL sheet in the table POINTS in the gateway project.
- 9) The SnapWiseCli from the GTW will receive the simulated data in the Wise simulator application. That way, besides the already existing points, we need to include the main points for the reopeners and circuit breakers, types: RCF, POS, LKO, CFL and BLF, as well as the command tags to open and close the circuit breakers and reopeners.

TagName	Node	Address	DataType	AccessType
*				
SH_SIMUL.POS	SinapWiseCli	SW:0:POS:	Native	Read
SH_MANOB[0].MANID	SinapWiseCli	AN:0:MAN:	Native	Write
REClo_90.STA.RCF	SinapWiseCli	SW:177:RCF	Native	Read
REClo_90.STA.POS	SinapWiseCli	SW:177:POS	Native	Read
REClo_90.STA.LKO	SinapWiseCli	SW:177:LKO	Native	Read
REClo_90.STA.CFL	SinapWiseCli	SW:177:CFL	Native	Read
REClo_90.STA.BLF	SinapWiseCli	SW:177:BLF	Native	Read

Figure 72 – Including all the main points of the reopeners and circuit breakers.

## 6.2. Running ActionWise / Simulator / Gateway

In the environment of ActionWise Gateway and ActionWise Simulator Network, you must first activate the simulator, as shown in the figure below.



Figure 73 -ActionWise Toolkit - opening the network to be simulated

- When the program opens, click **open network**, and the program will go to the directory of the last network opened;
- A double-click on the network that you want to open in the ActionWise network simulator, and the network graph will appear inside de main screen.
- With the existing network open, the program displays it on the screen and provides a toolbar for simulations, shown in figure bellow.
- To start communication with ActionWise Gateway, click on **Start Server**, as shown in the figure below, the simulator starts to work. The schematics are in green and on the electrical networks small white rectangles move slowly towards the flow of the current, showing that the simulator is active. The toolbar with several magnifying glasses, highlighted in blue in the figure below, has its functions explained in item 2.2.6 of this manual.

Beyond the simulation running, this program starts to serve analog and digital data over an integration bus that should be read by the **ActionWise / Gateway** program. You can see on gateway displays the currents and switches positions with the same measures and states as in

## Simulator.

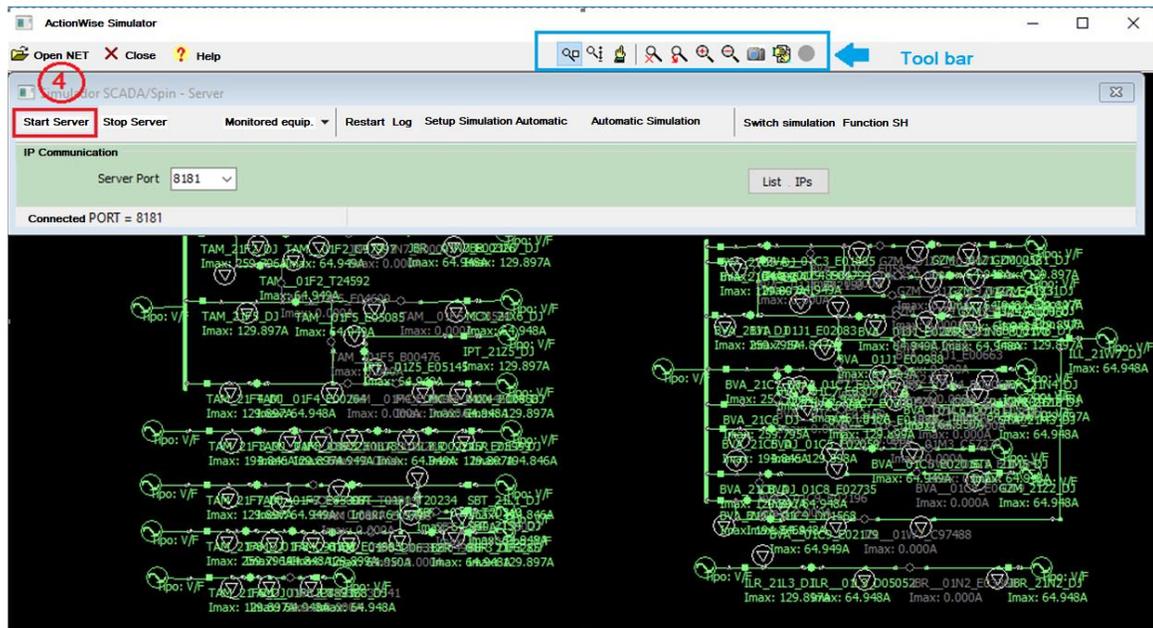


Figure 74 -ActionWise Toolkit - simulation start

- In the sequence, the ActionWise / Gateway program is triggered. When it starts, it prompts the user to login (default: super / s) and then opens the window shown in figure 64 and starts to process the information on the bus of integration. The gateway also starts to serve data changing in real time on IEC-60870-5-104 protocol for ActionWise / ADMS / FLISR.

The screen below shows the two programs running in the ActionWise / Gateway environment and the ActionWise toolkit.

As can be seen, immediately after the ActionWise / Gateway is activated, it connects to the Network Simulator, assuming the states of all keys and reclosers, as well as the value of the monitored analog variables.

Any failure that is triggered in the Network Simulator will be displayed on the ActionWise / Gateway screens and sent to ActionWise / ADMSS / FLISR via the IEC-60870-5-104 channel. When the FLISR function sets maneuvers to isolate the problem and re-establish the system, these commands will be sent to the Gateway that will send them to the Network Simulator. The schemes presented in this program will reflect the real-time situation of the simulated network.

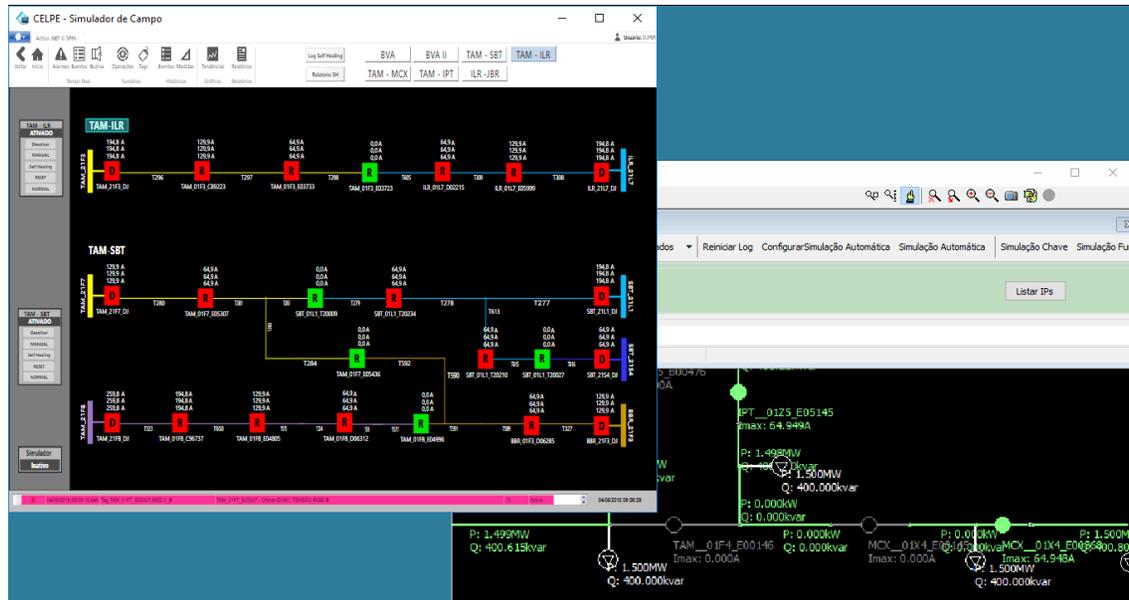


Figure 75 - ActionWise Gateway Environment and ActionWise Toolkit during Simulation

### 6.3 Executing ActionWise/ADMS/FLISR

This program, when activated, in this environment, automatically connects with the ActionWise / Gateway that is in the same address that should be the customer's SCADA. The information about operation of this simulator is described in the chapters above.

### 6.4. Toolbar of the Simulator

Next figure presents the overview of the ToolBar to control the ActionWise Simulator. Mainly there are three blocks of commands.

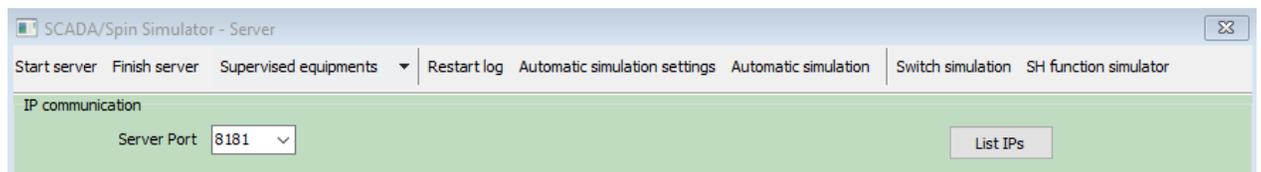


Figure 76 – Simulator tool bar

#### 6.4.1 Server Control

Commands to control server connection and activity:

- **Start server:** the simulator starts sending data to the integration bus.
- **Finish Server:** the simulator is closed
- **Restart log:** Restarts the event log messages presented to the user.

## 6.42. Automatic Simulation

These commands deal automatic simulation switches and options settings and start/stop automatic simulation.

Automatic simulation is a mode in which we can pre-program a set of switches to simulate occurrences sequentially and repeatedly.

### Automatic Simulation Settings

This command allows the user to choose which switches will participate of a simulating task. When clicking this button appears a dialog shown on next figure.

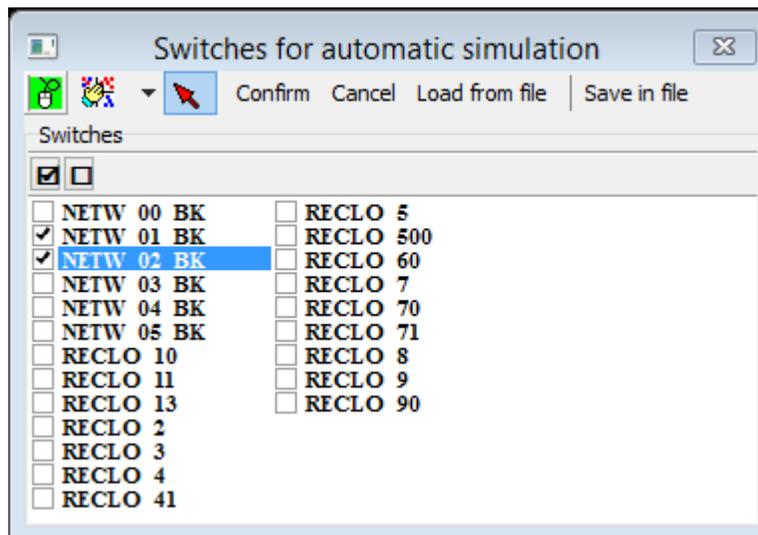


Figure 77 – Select switches for automatic simulation

Select switches clicking in box correspondent. Switch marked will participate of simulation.

### Automatic simulation

This option simulates permanent faults on all switches that were previously selected, in previous item.

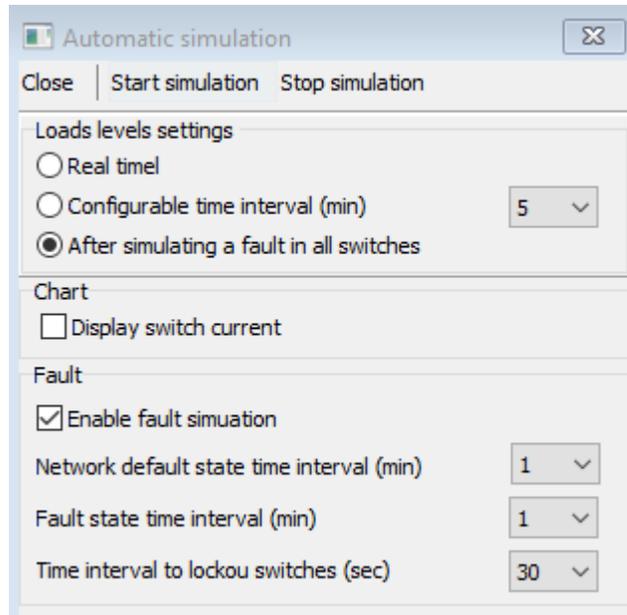


Figure 78 – Parameters for automatic simulation

The loads are represented by curves with loading values for each hour of the day. In the automatic simulation mode, the user can vary the hour of the day and the network loading using the following features:

- Load Level settings
  - Real time: the hour of the day is chosen according to the computer clock.
  - Configurable time interval: the hour of the day is changed after the chosen time interval is reached.
  - After simulating a fault in all switches
- The user can also choose the time interval used in the simulations:
  - **Network default state time interval** (min): the period in which the network is maintained in its default state. After this period, the chosen switch is opened.
  - **Fault state time interval (min)**: the period in which the network information about the permanent fault is maintained, before the next permanent fault simulation cycle.
  - **Time interval to lockout switches** (sec): the period between opening the switch and the activation of the lockout signal
- Close – to close this dialog saving options.
- Start/Stop Simulation – To activate the simulation or stop if running now.

### 6.4.3 Manual Operated Simulation

#### Switch simulation

This function allows the user to change statuses of the switches. After clicking in a switch on the screen to select it, the following window dialog appears to the user. This dialog presents to the user the status of each meaningful point (tag) of the switch for simulation and operation.

There is also available buttons to send commands to change status of these points.

In **Selected Switch** list box on top, we can change from one switch to another to get a new selection without closing the dialog.

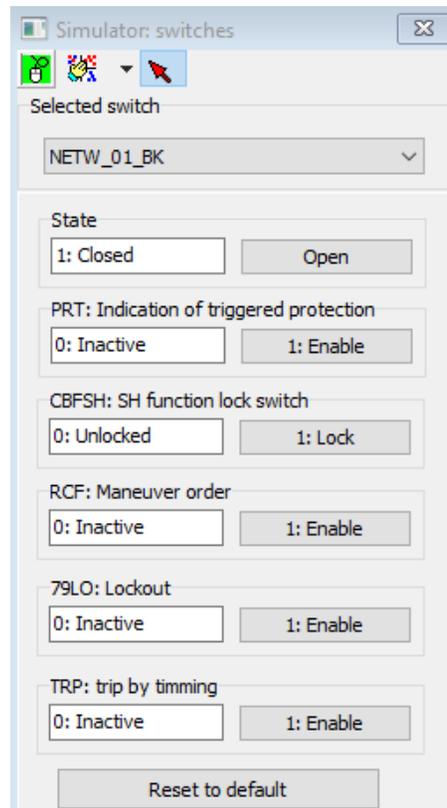


Figure 79 - Window to manually change switch states

- **State** - Status of the switch. Clicking the button change state of switch, as indicated in button;
- **PRT**: Indication of triggered protection (this signal is normally set by ActionWise)
- **CBFSH**: SH function lock switch (this signal is normally set by ActionWise)
- **RCF**: Manual Reconfiguration maneuver order (this signal is normally set by ActionWise)
- **79LO**: Lockout – Simulation of signal come from breaker, ending reclosing retries.
- **TRP**: trip by timing. (this signal is normally set by ActionWise, when triggering by protection)

### SH Function simulator

It allows the user to enable or disable the FLISR function in each network feeder. When click shows a dialog window with a list of all networks. We can disable/enable (signal BF) one or all

networks. If disabled, the whole network will not have permission to participate in automatic operations.

The toolbar include the following commands:

**Confirm** –To confirm and save any changes done in networks

**Cancel** – To close this dialog without saving any changes

**Enable all networks** - To enable all networks to FLISR functions.

**Disable all networks** - To disable all networks to FLISR functions.

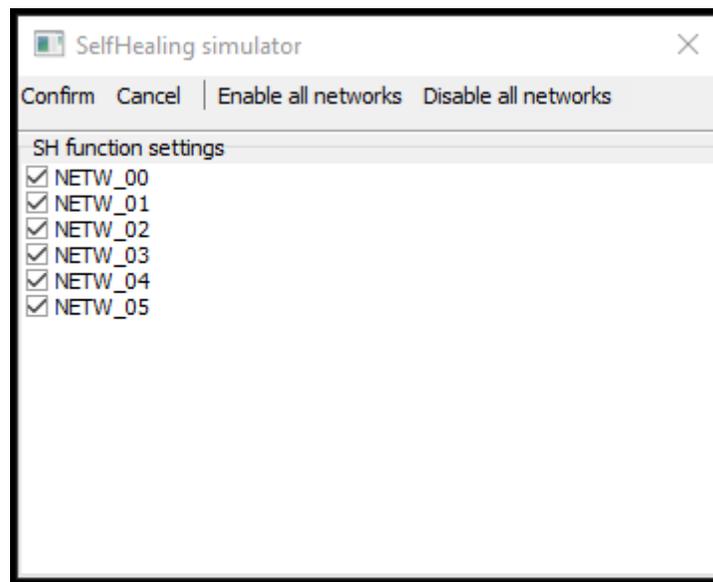


Figure 80 - Window to enable / disable existing networks

## 7. Other Features

In next items are some other features included in ActionWise system, regardless of the simulator.

### 7.1 Simple Commands

If the SCADA module receives a command, with Maneuver ID = zero, it will send directly to the recloser without displaying the execution window. In these commands will also be actuated the OPSH tag (operated by FLISH -self-healing), which is used to show a blinking yellow circle on the recloser symbols that were commanded by self-healing.

This facility will allow the implementation of change of the recloser protection adjustment.

### 7.2 Data Types in the protocol

Below the table with protocol field types.

#### Data Types in Equipment

Field description	Code	Acronym
"Switch Position",	0	"POS"
"Blocked for FLISR"	1	"BLF"
"Reclosing Lockout	2	"LKO"
"Source Id"	3	"SUP"
"Maneuver Command Fail"	4	"CFL"
"Current - Amp",	5	"A"
"Voltage - Volt",	6	"V"
"Active Power",	7	"KVA"
"Current - Amp- Secondary"	8	"AS"
"Voltage- Volt-Secondary",	9	"VS"
"Nominal current limit"	10	"LIM"
"Maximum Current reached"	11	"MAX"
"Maximum Current is predicted"	12	"MXP"
"Reconfiguration Network"	13	"RCF"
"Wise occurrence"	14	"OCW"
"Protection actuated"	15	"PRT"
"TRIP by protection"	16	"TRP"
"Command Close ",	20	"CMF"
"Command Open",	21	"CMA"
"Info for Maneuver",	30	"MAN"