Electrical Template

The Electrical Template: a ready power system solution with symbols, equipment models, operator dialogs and demo diagrams.

Create a solution from the template
What comes ready
Demo diagrams
Symbols
Equipment data model
Equipment dialogs

Create a solution from the template

1. Open **Solution Center** and click **New Solution**.
2. Select the **Windows** platform (.NET 4.8). The Electrical Template is a Windows solution and is not listed for Multiplatform.
3. Select **Electrical Template**.
4. Enter the solution name and location, and click **Create**.

The new solution is a copy of the template. The template file in the installation folder never changes.

> The Electrical Template is listed when the license includes the Electrical Pack, the Electrical Pack is installed, and the Windows platform is selected. See Electrical Pack.

What comes ready

Layouts and navigation

Layout	Regions
`Startup`	Header at the top, Menu on the left, BottomAlarm at the bottom, MainPage in the content region
`WithoutAlarm`	The same without the alarm bar

- **Header**: page path, solution name, the **SIMULATION** check box, theme selector, date and time, the logged user with log on and log off, close.
- **Menu**: MainPage, SingleLine, Graphics (trend), Alarm/History, Info, Solar PV, Mining, Data Center.
- **BottomAlarm**: active alarm list with acknowledge, and the alarm sound on and off button.

Solution content

- **Equipment models**: UserTypes for every equipment type, enumerations for states and voltage levels, and the voltage and switching color tables. See Equipment data model.
- **Symbols**: 144 symbols. See Symbols.
- **Scripts**: the `Power` class (the power flow engine of the solution), `SymbolScript` and `MovePopup` for the dialogs, and one simulator task per demo diagram.
- **Alarms**: groups Critical (acknowledge with comment), Warning, Events and AuditTrail (operator actions, safety tickets included), one alarm area per sample equipment so the equipment dialog can filter its alarms, and the alarm items the demos use.
- **Historian**: tables `Table1` and `Table2`.
- **Security**: users and permission groups. Review the command permissions of the groups your operators use before going to production.

First run

1. Start the runtime from **Runtime Startup**.
2. The client opens on MainPage, the SE Northfield 230/69 kV substation. It energizes a few seconds after the display opens.
3. Log on from the header to operate equipment.

To start your own project, remove the demo displays, simulator tasks and tags you do not need (see Demo diagrams), then follow Single Line Diagrams and Power Flow.

Demo diagrams

Each demo is a single line diagram driven by its own simulator task, with buttons for fault scenarios. Scenario 9 restores the normal state.

MainPage: SE Northfield 230/69 kV substation

- **Network**: two 230 kV lines and two 12 kV generating units through step up transformers feed the 230 kV bus. A 230/69 kV, 100 MVA transformer feeds 69 kV bus A with three feeders. A three winding 230/69/12 kV transformer feeds 69 kV bus B with two feeders and a 30 MVAr capacitor bank, and the 12 kV auxiliary services. The bus tie is normally open.
- **Side panel**: generation, grid import, total load, frequency, voltage color legend, a 10 minute power trend, scenario buttons, last event, transformer table and customers out of service.
- **Simulator**: `SE_Northfield_Simulator`. Power flow, tap changer regulation, transformer thermal model, automatic bus transfer when a 69 kV source is lost, and an accelerated day.
- **Scenarios**: 1 feeder ground fault, 2 loss of a 230 kV line, 3 generating unit trip, 4 transformer differential trip with bus transfer, 5 heat wave, 9 restore.

SingleLine: a simple example

A 230 kV line, a 230/69 kV transformer and one 69 kV feeder. It is the diagram built step by step in Single Line Diagrams and Power Flow. It has no simulator: turn on **SIMULATION** to operate it.

Solar PV: 120 MWp plant

- **Network**: a 138 kV line and a collector substation with a three winding 138/34.5/34.5 kV, 150 MVA transformer, two 34.5 kV collector buses with six 20 MWp inverter circuits, a capacitor bank, a shunt reactor and the auxiliary services transformer.
- **Simulator**: `Demo_UFV_Simulator`. Irradiance over an accelerated day, inverter output, and automatic reactive control (capacitor by day, reactor at night).
- **Scenarios**: 1 cloud passage, 2 inverter trip, 3 curtailment, 4 ground fault on a collector circuit, 5 loss of the 138 kV line, 9 restore.

Mining: 138/13.8 kV main substation

- **Network**: a 138 kV utility line, two 138/13.8 kV, 40 MVA transformers, two 13.8 kV buses with a normally open tie, a ball mill on a variable frequency drive, a crusher, a 5th harmonic filter, a 380 V motor control center, conveyor and slurry pumps, a capacitor bank and a 2.5 MVA emergency generator.
- **Simulator**: `Demo_MIN_Simulator`. Loads, tap changers, bus transfer on transformer loss, load shedding and generator start on utility loss, harmonics.
- **Scenarios**: 1 mill start, 2 crusher short circuit, 3 transformer differential trip, 4 loss of utility, 5 harmonic filter out, 9 restore.

Data Center: Tier III 2N

- **Network**: two independent sides, A and B. Each has a 13.8 kV utility feed, a 13.8/0.38 kV transformer, a 380 V main switchboard with a generator on an automatic transfer switch, a chiller, a UPS with battery and a critical bus with three PDUs. Dual cord racks take one cord from each side.
- **Simulator**: `Demo_DC_Simulator`. Transfer to the generator and back, UPS on battery, battery autonomy, redundancy and PUE.
- **Scenarios**: 1 utility A failure, 2 blackout on both sides, 3 generator B start failure, 4 UPS A failure, 5 generator A monthly test, 9 restore.

Operation pages

Display	Menu	Content
TrendNew	Graphics	Trend with a tag tree for pen selection, zoom, pan, real time and pause, annotations and saved pen groups
AlarmsNew	Alarm/History	Online alarms, history and audit trail, with filters and acknowledge
About	Info	Solution name, path, last change, build, computer name and IP address
Architecture	(not in the menu)	A network architecture page built with the Architecture symbols

Removing a demo

Delete its display, its simulator task and its tags. The Solar, Mining and Data Center simulators also stop when their `SimEnabled` tag is 0.

Symbols

The electrical symbols are in the Symbols panel of the Draw editor, under **In Solution**. An electrical symbol is a regular symbol with three extra traits:

- **Terminals**: connection points (`Node1`, `Node2` and so on) that the Connector tool attaches to. Each connection is saved with the display and becomes an edge of the network the power flow engine follows.
- **The electrical flag** (`IsElectrical`): only symbols with this flag take part in the power flow. Every equipment symbol below has it set.
- **An equipment link**: a symbol property named after the equipment type (`Breaker`, `Generator`, `Transformer` and so on) that points to a tag of the UserType with the same name. When a symbol is first placed, the link points to a placeholder tag such as `@Tag.UI.DummyBreaker`; replace it with your equipment tag.

Equipment symbols

Equipment	Symbols	UserType	Terminals

Generator or source	Generator, Generator_H, Generator_V, Generator1, Generator_Transmission (a line from the grid), Generator_Continue, Generator_LineColor	Generator	1
Circuit breaker	Breaker, Breaker_H, Breaker_V	Breaker	2
Disconnector	Switch, Switch_V	Switch	2
Fuse	Fuse, Fuse_Type2	Fuse	2
Transformer, two windings	Transformer, Transformer_H, Transformer_V, Transformer_Type2	Transformer	2
Transformer, three windings	Transformer3, Transformer3_H, Transformer3_V, Transformer3_Type2	Transformer3	3
Reactor	Reactor, Reactor_H, Reactor_V	Reactor	2
Rectifier or UPS	Rectifier, Rectifier_H, Rectifier_V	Rectifier	2
Capacitor bank	Capacitor	Capacitor	2
Busbar	Bar, Bar_Node	(none)	Any number; all terminals are the same node
Directional busbar	BusBar	BusBar	98: odd nodes 1 to 95 on top are inputs, even nodes 2 to 96 at the bottom are outputs, nodes 97 and 98 at the ends tie bus couplers
Load, ground, continuation	Standard_Load, Standard_Load2, Standard_Ground, Standard_Continue_Out	(none)	1
Line crossing without connection	Jumper	(none)	2

_H and _V are horizontal and vertical drawings of the same equipment. `Type2` variants use an alternative drawing.

Properties set on equipment symbols

Property	Meaning
`BaseVoltage`, `BaseVoltage1` to `BaseVoltage3`	Source voltage (generator) or winding voltage (transformer). Choose a level from `Tag.Enum.Voltage`, for example `@Enum.Voltage.kV_230`.
`TypeConnector1` to `TypeConnector3`	Winding connection: Delta, Wye or WyeGrounded
`DlgDetails.Name`	Dialog opened when the operator clicks the symbol (default `DlgDetails`)
`DlgDetails.ItemsPerColumn`	Rows per column before the dialog adds a column
`DlgDetails.AlarmObj`, `DlgDetails.AlarmPerItem`	Show the equipment alarm list in the dialog, and an alarm indicator on each point
`BtnInterlock.*`	Expressions that lock command buttons. See Equipment dialogs.
`DlgInterlock.Open`, `DlgInterlock.Close`	Permissive logic drawn in the interlock diagram

Other symbols

- **Measurement blocks**: MeasureBox1 and MeasureBox4 show one or four measurements next to an equipment.
- **Dialog building blocks**: command blocks, point rows, dialog headers per equipment type, close, ticket, resize and the interlock diagram parts. The dialogs use them; they are not meant for a diagram.
- **Architecture symbols**: servers, databases, PLCs, HMIs, firewalls, network switches, wireless, mobile devices, cloud, wind turbines, solar, hydro, ships and satellites, for a network architecture page. They are not part of the power flow.
- **Images**: icons used by the header, menu and alarm pages.

Equipment data model

Each piece of equipment is one tag whose type is the equipment UserType, for example `Tag.Substation1/CB1` of type `Breaker`. Its points are members of the tag, for example `Tag.Substation1/CB1.CmdBreakerPosition`. Map the field points to your devices; the power flow engine and the dialogs work on the same members.

Naming convention

Prefix	Meaning	Mapped to the field	Example

`Cmd`	Command. A command with an enabled `SLT` twin is sent as select before operate.	Yes (output)	`CmdBreakerPosition`, `CmdBreakerPositionSLT`
`In`	Position input of a switching device	Yes	`InBreakerPosition1`, `InBreakerPosition2`
`Dg`	Digital state or status	Yes	`DgState`, `DgEnableOpening`
`Pt`	Protection function pickup or trip	Yes	`PtPhZone1DistanceTrip`
`An`	Analog measurement	Yes	`AnActivePower`, `AnCurrentPhaseA`
`T`	Internal: colors and voltages written by the power flow engine, dialog settings copied from the symbol, interlock results	No	`TCurrentColor1`, `TVoltage1`

Equipment UserTypes

UserType	Members	Content
Generator	18	Start, stop and emergency stop commands, running state, manual or automatic, start failure, low fuel, abnormal voltage, base voltage
Breaker	186	Commands, position inputs, digital states, protection points (distance zones, overcurrent, over and undervoltage, lockouts 86, 86L and 86BF, reclosing 79, synchronism 25, teleprotection 85), and three phase measurements
Switch (disconnector)	33	Open and close command, position inputs, releases, local or remote, internal fault, command failure, pole and position discrepancy
Fuse	14	Fuse state and command
Transformer	204	Tap changer (raise, lower, manual and automatic), forced air stages, paralleling, 86T reset, Buchholz, pressure relief, temperatures, oil level, differential and overcurrent protection, measurements per winding
Transformer3	216	As Transformer, with a third winding
Reactor	124	Buchholz, pressure relief, temperatures, protection trips, currents, 86T reset
Rectifier	21	State, fault, AC and DC abnormal, ground leakage, battery discharging
Capacitor	13	Energization and color on both terminals
Bar	3	Bus lockout 86B state and reset
BusBar	9	Directional busbar: base voltage and color
Standard	6	Voltage, state and color of a load or ground

What decides which points the dialog shows

- **Disable**: a disabled member is ignored by the equipment dialog. Every member is enabled in the template; disable the points your project does not use. The runtime also drops writes to a disabled member, so never disable a point that a device, a script or the simulation writes.
- **Display Text**: the label the operator reads. A `Dg`, `An` or `Cmd` member appears in the dialog when it is enabled and has a Display Text other than empty or `-`. Rows are sorted by Display Text. Translate the dialog by translating the Display Texts.



The power flow engine reads the Disable flag of three members as wiring settings:

- `Breaker.InBreakerPosition2` and `Switch.InSwitchPosition2`: enabled means a double point position (state = 2 × Position2 + Position1); disabled means a single address (state = Position1).
- `Fuse.DgFusePosition`: when disabled, the engine does not read the fuse state and the fuse keeps its initial state.

Key members

Equipment	Members	Role
Generator	`CmdState`, `DgState`, `CmdEmergencyStop`	Start and stop command, running state (1 energizes the generator terminal at its base voltage), emergency stop
Breaker	`CmdBreakerPosition`, `InBreakerPosition1`, `InBreakerPosition2`	Open and close command; position read by the engine (0 intermediate, 1 closed, 2 open, 3 inconsistent). Closed passes power, open blocks it.
Breaker, Switch	`DgEnableOpening`, `DgEnableClosing`, `DgLocalRemoteRelay`	Open and close releases from the IED and the relay local or remote state, used by the default interlocks

Switch	`CmdSwitchPosition`, `InSwitchPosition1`, `InSwitchPosition2`	As the breaker
Fuse	`DgFusePosition`, `CmdFusePosition`	Fuse state and command
Transformer	`CmdTapValueUp`, `CmdTapValueDown`, `CmdTapAuto`, `CmdTapMan`, `AnTap`, `AnVoltage`	Tap changer raise, lower and mode, tap position and regulated voltage
All	`TTicket`	The safety ticket of the equipment

Extending a UserType

- Add your own points following the prefixes. A new `Dg`, `An` or `Cmd` member with a Display Text appears in the dialog automatically.
- Keep the `T` members and the members the engine reads (`DgState`, the position inputs, `DgFusePosition`) with their names.

Equipment dialogs

The template has one equipment dialog, the **DlgDetails** display. Every time it opens, it builds itself from the points of the equipment that was clicked, so a breaker, a transformer and a generator get different dialogs from the same display, and new equipment is ready to operate as soon as its symbol is placed and bound.

How the dialog opens

When the operator clicks an equipment symbol, the symbol writes the equipment tag name to `Tag.UI.TagNameDlg`, copies its dialog settings and interlock expressions into the equipment tag, and opens the display named in `DlgDetails.Name`. One equipment dialog is open at a time per client; drag it by its title bar.

Layout

Area	Content
Title bar	Equipment type and name, safety ticket button, close
Equipment block	A live copy of the equipment symbol with its main commands, each with a lock icon
Digitals	The `Dg` points with their state text. Transformers group them in sections set by `DlgDetails.DigitaisSections`; a point joins a section when its Display Text starts with `#Section#`.
Analog	The `An` points with value and unit
Commands	One block per `Cmd` point; button captions come from the point enumeration
Special blocks	Three position protection transfer on breakers; tap changer mode and raise and lower with voltage and tap position on transformers
Alarms (optional)	The equipment alarm list and acknowledge, when `DlgDetails.AlarmObj` is Visible

Commands and select before operate

1. The operator clicks a command button. A red lock means the command is blocked and nothing happens.
2. The confirmation dialog asks to confirm the command.
3. **Yes** sends it. When the command point has an enabled `SLT` twin, the select is written first and then the operate; disable the `SLT` point for direct commands.

The state shown comes back from the field through the `Dg` and `In` points. Without field devices, turn on **SIMULATION** in the header so the engine echoes commands to states.

Interlocks

Each command button has a boolean expression, set on the symbol, that locks the button while it is true. The dialog shows a red lock (blocked) or a green lock (free).

Symbol property	Locks	Default
`BtnInterlock.Open`	Open	`me.DgEnableOpening = Enum.Blocked OR me.TTicket.Color = Enum.Red`
`BtnInterlock.Close`	Close	`me.DgEnableClosing = Enum.Blocked OR me.TTicket.Color = Enum.Red`

`BtnInterlock.Cmd`	Other commands	Breaker: `me.DgLocalRemoteRelay = Enum.Local`
`BtnInterlock.TAPUp`, `BtnInterlock.TAPDown`	Raise, lower	Tap changer in automatic, or at its limit
`BtnInterlock.ForceVent`	Forced ventilation	`me.DgForcedAirRemote = Enum.Local`

In an expression, `me.` is the equipment bound to the symbol, `Enum.<State>` is a state constant, and `AND`, `OR`, `!` and parentheses combine conditions. A point of another equipment is written `@Tag.<path>.<point>`.

> Releases read Blocked until the IED reports them. With no field data, Open and Close show a red lock: this is the expected safe state.

Clicking the lock next to Open or Close opens **DlgInterlock**, a ladder drawing of the permissive logic in `DlgInterlock.Open` or `DlgInterlock.Close`: AND conditions in series, OR conditions as parallel branches, inverted conditions on a red node, and each node with its point label and current state.

Safety tickets

The ticket button in the title bar opens **DlgTicket** for the equipment. A ticket has a title, a note and a color: **White** is a caution ticket, **Red** is a lockout ticket that locks Open and Close in the default interlocks. The last edit time and user are filled in automatically, and adding or removing a ticket writes an Audit Trail entry.

Using your own dialog

Set `DlgDetails.Name` on a symbol to another display, and in that display read the equipment from `Tag.UI.TagNameDlg` as DlgDetails does.

In this section...