PowerFactory and StationWare Verified Relay Models

PowerFactory Users' Group Meeting

Sydney 05/09/13

Brad Henderson

DIgSILENT Pacific

"It is a capital mistake to theorize before one has data. Insensibly, one begins to twist the facts to suit theories, instead of theories to suit facts." – Sherlock Holmes

Introduction

1. Protection relay modelling and simulation – why?
2. Storage of relay settings information within StationWare
3. PowerFactory relay models
4. How we get the data into PowerFactory
5. The DIgSILENT Pacific Verified Relay Model
6. Challenges with settings export

Why do we need protection modelling and simulation?

- Reason 1: Arc flash hazard assessment
- The calculation of the incident energy is critically dependent on the duration of the arc flash event
- The arc duration is limited by the operation of protection
- How can you determine the arc flash duration if you don't have an accurate model of the protection system?
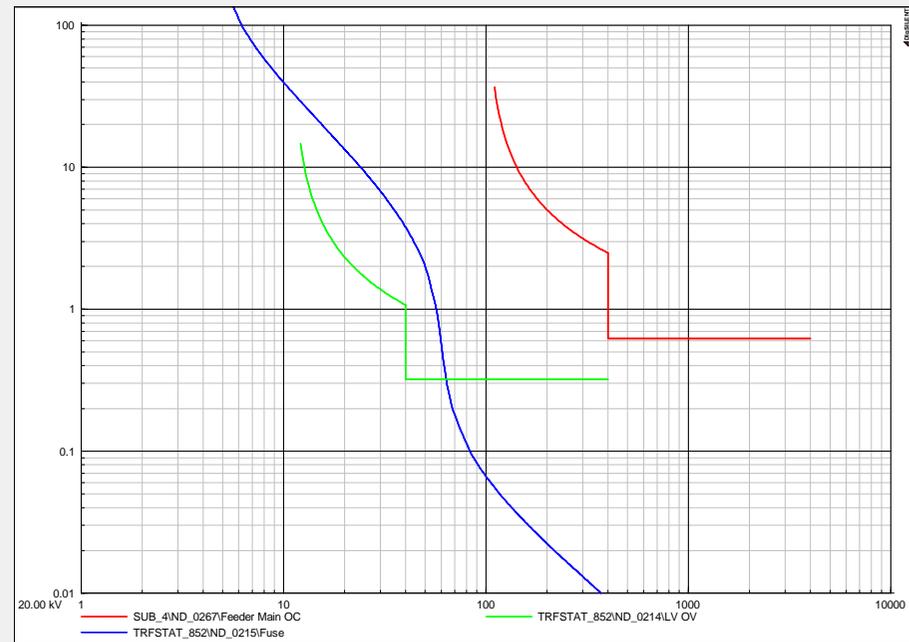
Why do we need protection modelling and simulation?

- Reason 2: Efficient protection coordination

- PowerFactory has tools such as the TOC plot that allow you to quickly identify mal-coordination

- More complex sequential coordination issues can be assessed with DPL scripts

Why do we need protection modelling and simulation?

- Reason 3: Compliance
- Many utilities outsource their protection design
- But…the regulatory burden of a fit for purpose protection system is their responsibility
- Many utility engineers are too time constrained to do detailed settings checks
- Automated settings import and analysis tools can allow for more comprehensive checks


Z1 overreach
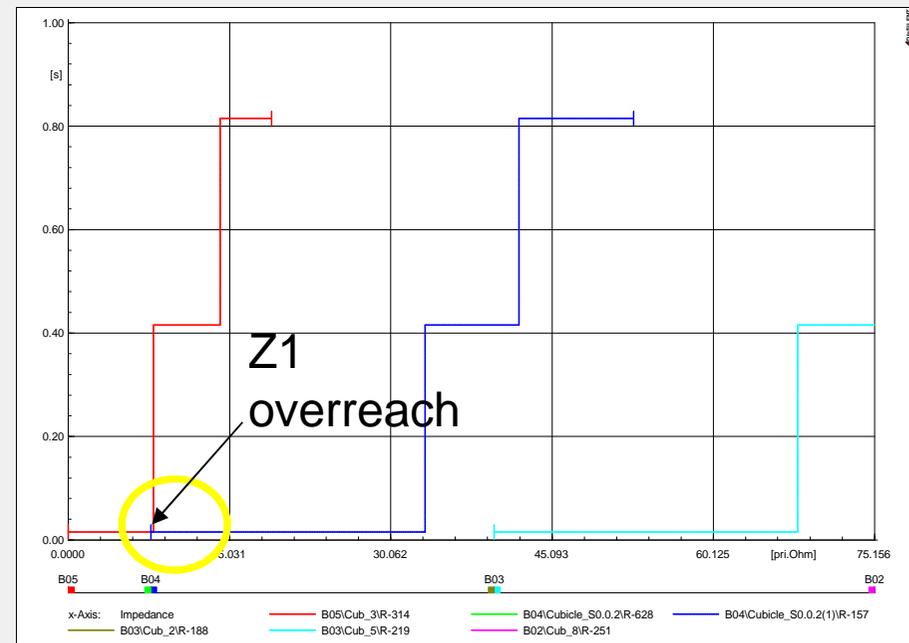
Data management

- Protection settings management is now highly data centric due to complex numerical relays.
- Consequently, the data management systems used are critical to back-up efficient processes.
- DIgSILENT supports efficient settings management and analysis through:
  - Settings storage in StationWare
  - Detailed network and relay models in PowerFactory
  - The ability to link settings in SW and PF through DPL scripts

Settings storage in StationWare

- StationWare was designed to be a *vendor independent* repository for relay settings management
- Settings management in StationWare centres around a settings process lifecycle
- This defines the work-flow process for a relay setting from conception -> design -> review and application in the field
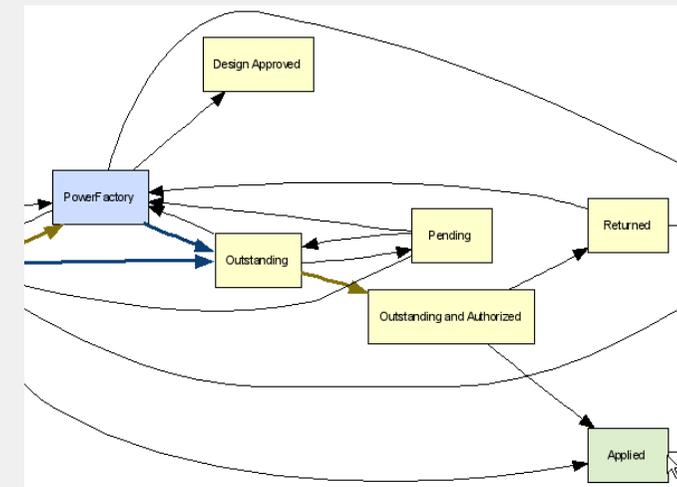
StationWare settings format

- StationWare defines relay settings in XML which is inherently platform independent.
- A 'setting' in StationWare supports integer, string, floating point and enumeration data types
- The idea is to capture every setting and store it in this vendor independent format

```xml
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes"?>
<PSMS>
  <Data>
    <Objects>
      <Object Id="19369" Name="Test for Scott" Description="" Type="PSMS.DeviceModel.REL521_generic.Device_Relay">
        <Attributes>
          <Attribute DeclaringType="PSMS.Custom.Attributes.Voltage" AttrDefName="VoltLev">0</Attribute>
          <Attribute DeclaringType="PSMS.Custom.Attributes.Voltage" AttrDefName="GlobalNote" />
          <Attribute DeclaringType="PSMS.Custom.Attributes.Overall" AttrDefName="OverallStatus">Settings ok</Attribute>
          <Attribute DeclaringType="PSMS.Custom.Attributes.Attr" AttrDefName="Location" />
          <Attribute DeclaringType="PSMS.Custom.Attributes.Attr" AttrDefName="LocationName" />
          <Attribute DeclaringType="PSMS.Custom.Attributes.Attr" AttrDefName="StreetName" />
          <Attribute DeclaringType="PSMS.Custom.Attributes.Attr" AttrDefName="DFISPickID" />
          <Attribute DeclaringType="PSMS.Custom.Attributes.Attr" AttrDefName="DFISEquipType" />
          <Attribute DeclaringType="PSMS.Custom.Attributes.test" AttrDefName="test" />
          <Attribute DeclaringType="PSMS.Custom.Attributes.test" AttrDefName="PSMS.Custom.Attributes.test_real">0</Attribute>
```

Getting settings into StationWare

- For simple electromechanical relays, settings can be manually entered through the web-interface
- For numerical relays, settings should be imported from the Vendor specific settings file
- This creates some challenges:
  - Every vendor format is different
  - Usually not all settings information such as valid ranges and enumerations is stored within the settings file.
  - Sometimes the settings file is in binary format (not plain text)

StationWare import filter

StationWare

- DIgSILENT licenses StationWare import filters that take the vendor settings files and translate them into the StationWare XML format

- Sometimes the import filter can be developed to support a family of devices that share a common format

Import filter

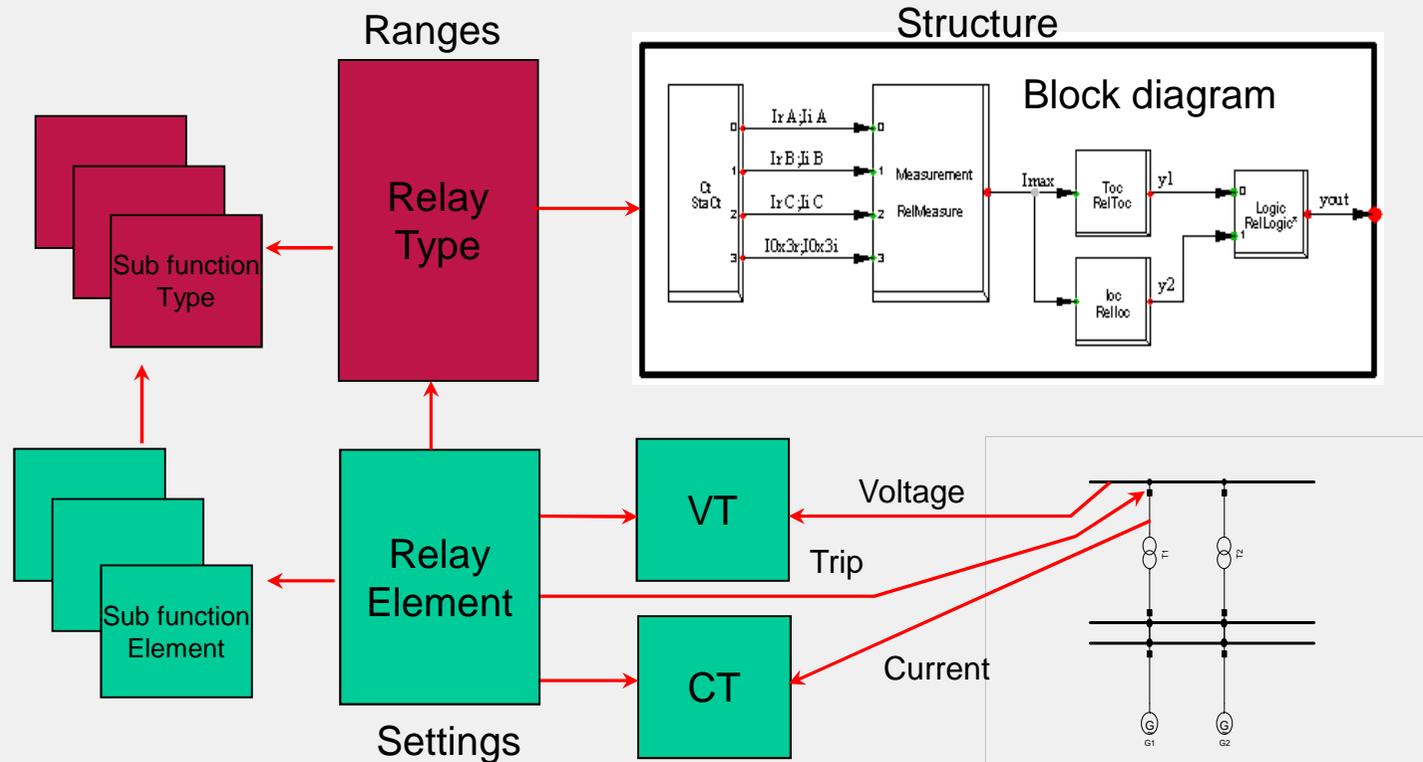- In other cases, an import filter might only support a single relay variant

PowerFactory relay model focus

- The PowerFactory relay model has a completely different focus to the StationWare representation
- It is not a repository for settings, rather a model to simulate and determine the performance of the relay
- The objective is to accurately model the mathematical and logic functions of the relay
- When combined with the built-in analysis tools this allows for rapid checks of protection coordination

The structure of a PowerFactory relay model

Ranges

Structure

Block diagram

Relay Type

Sub function Type

Sub function Element

Relay Element

VT

CT

Voltage

Trip

Current

Settings

Example 1 of PF model detail: quadrilateral characteristics

Example 2 of PF model complexity – polarising blocks

- The PF polarising block supports all the major types of voltage polarisation
- Earth fault compensation factor
- Mutual compensation factor

PowerFactory relay model complexity

- The PowerFactory model is a complete mathematical representation of the relay
- By necessity, a complex relay has a complex model
- Configuration and entry of settings can be time consuming and error prone
- This can be a barrier to entry for some users
- **Solution:** DIgSILENT Pacific provides the verified model service to take the burden of relay model preparation away from our clients

Settings exchange between StationWare and PowerFactory

- Due to reasons of complexity and efficiency, manual translation of settings between different formats should be avoided.

- **Solution:** StationWare – PowerFactory import through DPL

- The DPL exchange script takes the StationWare settings and translates these into equivalent PowerFactory settings

- Effectively the script is a translation table

- In some settings, logic is required to determine how the setting should be implemented in PowerFactory

StationWare

XML

DPL
script
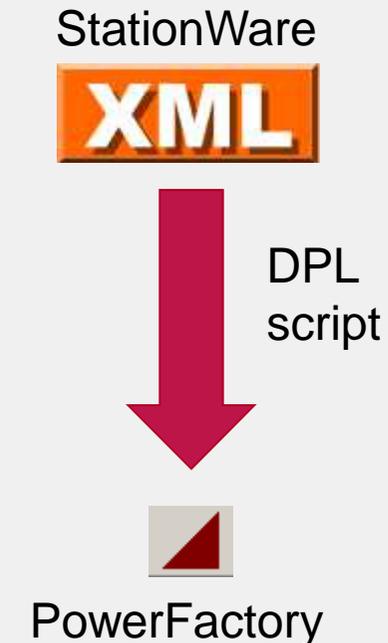
PowerFactory

Example of settings exchange complexity: E21P

- The SEL 311L relay has a setting E21P
- It is an 'enable' setting for the Mho phase distance elements
- Type 'enumeration' with possible values N, 1-4, 1C-4C
- In the PowerFactory relay model of the 311L, each Mho block is modelled separately with an out of service flag
- The DPL import script has to examine E21P and then enable/disable the appropriate number of Mho blocks based on this.

DPL logic – E21P

| Decision	Branch	Action
E21P = N	Yes	Set all Mho blocks OOS
E21P = 1	Yes	Set Mho 1 in service, all other Mho blocks out of service
E21P = 2	Yes	Set Mho 1&2 in service, all other Mho blocks out of service
E21P = 3	Yes	Set Mho 1, 2 and 3 in service, all other Mho blocks out of service
E21P = 4	Yes	Set all Mho blocks in service
E21P = *C	Yes	Report error to user that communication based settings are not supported
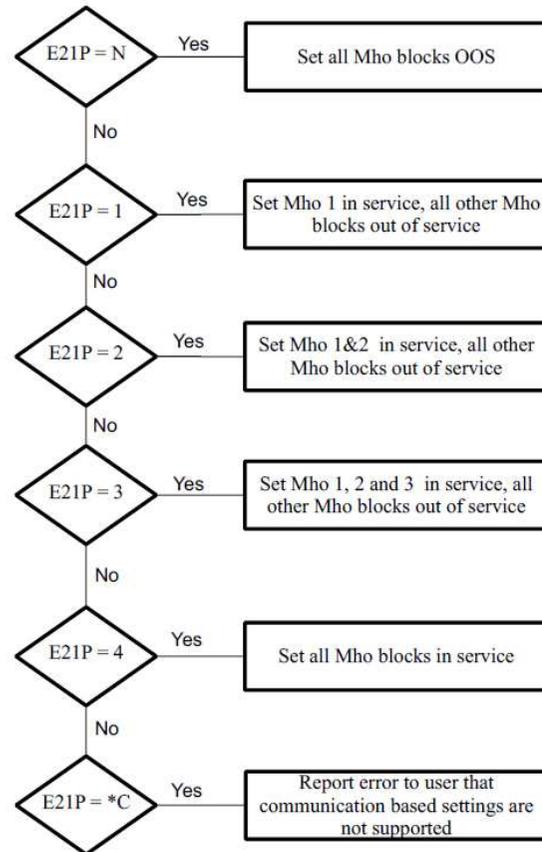
The complete picture

- Developing a detailed relay model in PowerFactory, maintaining a StationWare representation for the relay and also developing DPL scripts for import and export as explained is a non-trivial task
- The barriers to entry for DIgSILENT clients are generally too high
- A high level of user competence is required
  - PowerFactory relay modelling
  - StationWare modelling and handling of settings
  - Complex DPL scripting
- DIgSILENT Pacific identified this as a service that could be provided and managed

The DIgSILENT Pacific Verified Model
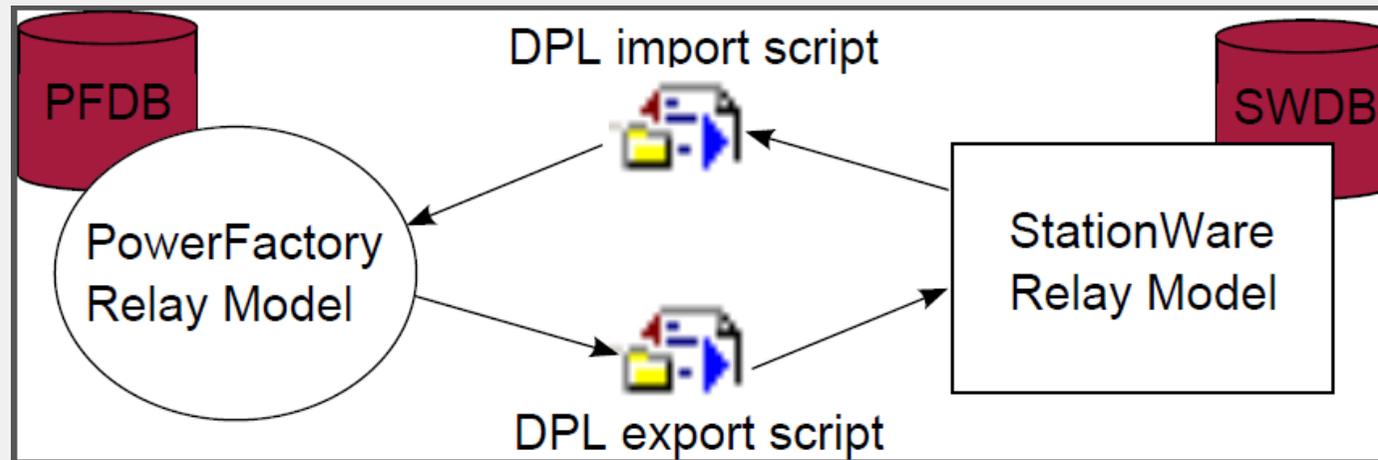
Development of a Verified Model

- A VM is specific to a relay model / part number / firmware variant
- Development of the model involves the following components:
  - Updating/creating/checking the PowerFactory relay model
  - Creation of the DPL import and export scripts for settings exchange with StationWare (the most time consuming part)
  - Creating the matched StationWare relay definition
  - Documentation describing the model, how the settings map from SW to PF and test results.
- The StationWare import filter is licensed separately but crucial to enable settings import direct from vendor settings files.

Testing the PowerFactory relay model

- Test scripts have been developed to automatically test each sub-function of the relay model

- Tests for trip and restrain are completed on an example system

- A report indicates PASS/FAIL for each sub-component

Table 4.3: Polygonal earth fault tests (Phase A - Ground short circuit at 84.92% reach)

| Block	Trip time s	Expected s	Pass/Fail
Z1	9999.999	>1.000	PASS
Z1B-exten	9999.999	>1.000	PASS
Z2	9999.999	>1.000	PASS
Z3	9999.999	>1.000	PASS
Z4	9999.999	>1.000	PASS
Z5	9999.999	>1.000	PASS

Challenges with the DPL export process

- The PowerFactory model contains only a subset of the StationWare settings, typically only those settings that are relevant for the main protection functionality
- Consequently, the PowerFactory model has an incomplete set of relay settings
- This creates difficulties with settings export to StationWare
- How to recreate the missing settings…

Settings export process

Import process

StationWare Applied settings

PowerFactory settings

StationWare PowerFactory settings

Export process

PowerFactory settings
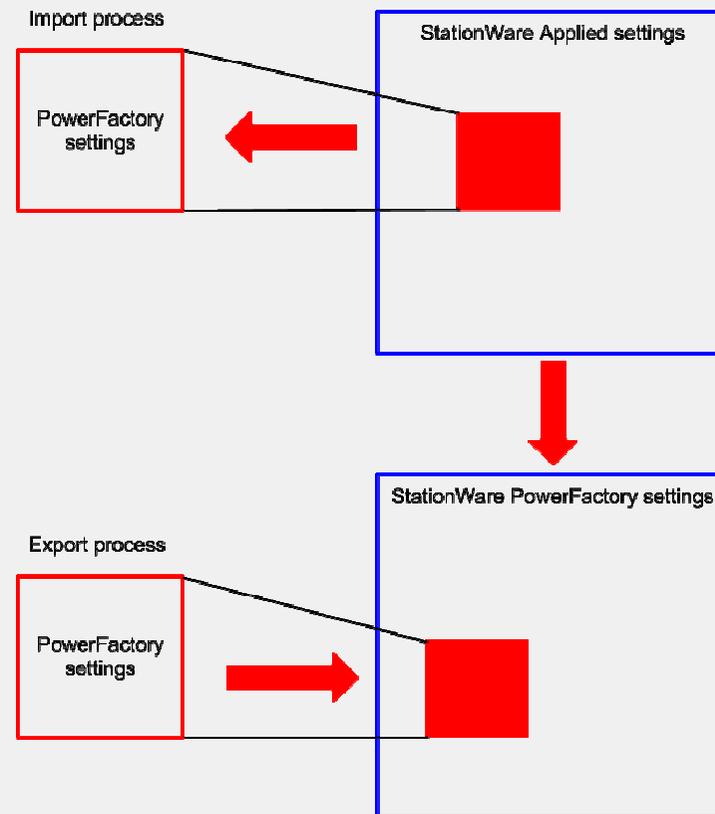
Settings export problems

- The settings export process works well in many situations.

- However, when complex dependencies exist in the relay settings it is currently not possible to accurately reconstruct the missing PowerFactory settings in StationWare

Siemen 7SA632 (starting and load area dependency)

- There is no enable/disable flag for the load area element in the vendor software
- It is always ON if the starting element is set to 'Impedance Z' and OFF otherwise
- Consequently, the load area settings should only be exported if the starting element is set to 'Impedance Z'

7SA632 export problem

- A user imports a 7SA632 settings file set to 'Imp Z' and the load area is enabled.

- In PowerFactory, they decide to change the starting method to over-current.

- They export the settings back to SW. However, the export DPL understands the load area dependency so doesn't export the settings for this element

- StationWare doesn't model dependencies, so it sees the load area settings as missing from the export and combines the original 'Applied settings for this block with the exported PowerFactory settings.

- Result – StationWare now has an incorrect set of settings.

Conclusion

- DIgSILENT sees considerable benefit to modelling protection relay devices accurately in PowerFactory.

- DIgSILENT Pacific has created a concept of verified models to deliver accurately matched StationWare and PowerFactory models to clients

- The verified model also contains export and import scripts to enable settings transfer between the two software platforms.

- Some challenges still exist with settings export, but these will be addressed in future versions of PowerFactory and StationWare.