GE VERNOVA

ADMS

16.20.0

Modeling Overview and Converter User Guide

Copyright

Change History

Date	Change Description
May 2, 2023	ADMS 3.16: • Updated Device ID Namespace and Considerations. • Editorial and formatting updates.
Oct 20, 2023	ADMS 16.1.0: • The product version numbering has changed. The prefix '3' is no longer included in the version numbering for ADMS.
Nov 1, 2023	ADMS 16.2.0: • Added Phase Augmentation Options.
Apr 1, 2024	ADMS 16.7.0: • Updated document format and styling.

Contents

1. Overview

ADMS is GE's solution for Distribution Management Systems (DMS).

ADMS is a suite of modular applications that complement each other, to form one of the most reliable and scalable DMS platforms in the industry. ADMS meets the real-time network management needs of both large and small electric distribution utilities, providing a full range of high-quality network operation functions designed for seamless integration with other systems, such as Supervisory Control and Data Acquisition systems (SCADA).

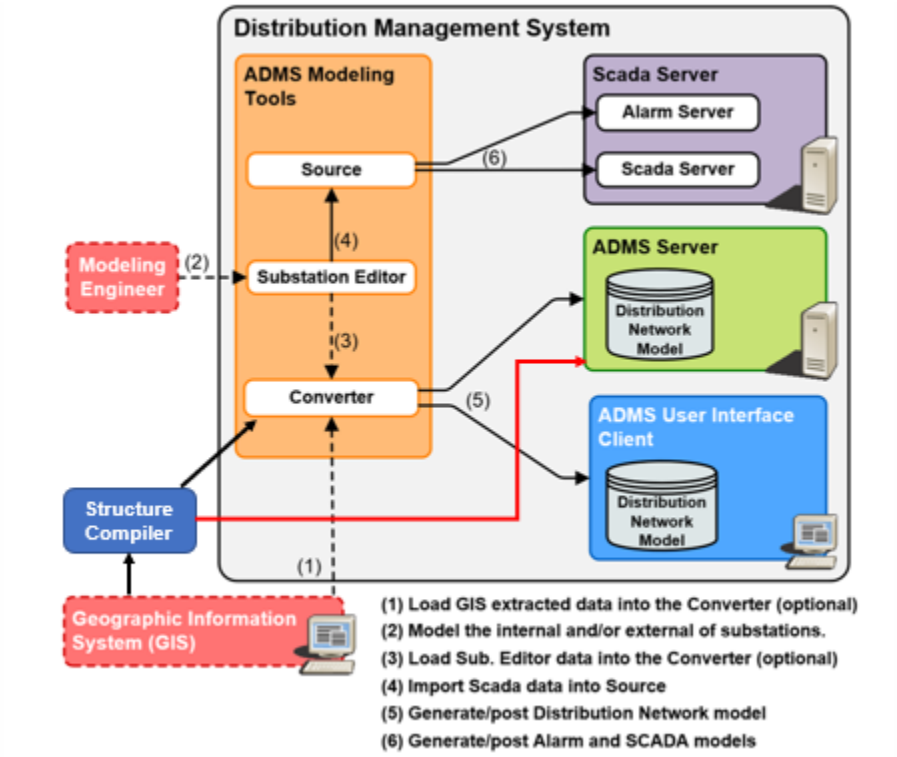


Figure 1. Distribution Management System Overview

1.1. Modeling Philosophy

The network operations model is the core model of the ADMS Distribution Management System (DMS). It maintains both the static and the dynamic data describing the distribution network. The data is structured to support all the required functionality of a DMS: real-time operations, network analysis functions, and user interface displays for the distribution network. This functionality, combined with the performance requirements for the efficient operation of a very large distribution network, means that a highly specialized and optimized data structure is required.

Developing and populating the data model of a utility's distribution network is usually the single greatest cost of many system management projects. In many cases, the investment is so great that,

unless the data modeling costs can be leveraged across a number of systems, there may be no real business benefit from each individual system. While it is clear that the easy interface of data among systems is very beneficial, it is also the case that the differing requirements of each system mean that specialized internal data formats are necessary to achieve the required performance and functionality. Therefore, in order to leverage the benefits of common data interface and provide the flexibility within the individual systems, it is essential that a neutral data format be available to facilitate the data interface.

The Common Information Model (CIM) is an abstract model that represents all the major objects in an electric utility enterprise that are typically involved in utility operations. The CIM provides a standard way of representing power system resources as object classes and attributes along with their relationships. This is accomplished by defining a common language (i.e., semantics and syntax). It is a schema, not a database or a specific data file format.

The object classes represented in the CIM are abstract in nature and can be used in a wide variety of applications. The use of the CIM goes far beyond its application in an Energy Management System (EMS) or DMS. The CIM standard should be understood as a tool to enable integration and plug compatibility among applications and systems, independent of any particular implementation.

1.2. Modeling Process

The data types needed to develop the DMS model are:

- Distribution utility Asset Management System (for example, GIS) with land base and distribution connectivity models
- Substation internal connectivity models
- Device settings (for example, recloser, capacitor, and regulator settings)
- Component/facility ratings (lines and devices)
- Equipment impedances and ratings
- System impedance data
- Relay setting data
- Customer information (customer count and load data)
- Load schedules

For GE’s solution for Distribution Management Systems, the information for the distribution facilities (lines, switching devices, transformers, energy consumers, and others) is stored in a CIM-based extensible markup language (XML), or a Comma-Separated Value (CSV) "station externals" file for each distribution substation (DRAFT IEC 61970 Rev. 4).

The data in this station’s externals file typically represents utility assets outside the substation fence, downstream of the station getaway point or feeder head. The source of the facility data may be a distribution utility Asset Management System (for example, GIS) or an output file created through the Substation Editor application

Measurements to be used by the DMS, along with portions of the balanced network model representing the substation internals or the sub-transmission, are modeled in a separate XML file referred to as the "station internals" file. A separate station internals file exists for each station. The data can be manually entered or created through the Substation Editor.

In addition, a global resources file is used to efficiently store modeling data that is common to many station file components.

There is also a defaults data file, which contains default data for selected fields that have not been provided by the source data system. This is an optional file.

1.2.1. Model Validation and Conversion

The data in the two station files (internals and externals files) is combined to create a station binary file that is the single data source for both the network analysis applications and the Network View. This binary file is created using the ADMS Converter application.

The Converter validates and merges the data from the substation internals XML files and the XML or CSV station externals files and then outputs the binary operations model files. The validation process consists of data integrity checks that verify that the data can be processed by ADMS. If data errors exist, the operator is notified with messages indicating the extent of the problem.

When the modeling errors are corrected and new station model files have been created by the Converter, the model files can be utilized in ADMS by placing them in the appropriate ADMS subdirectories. For more information about loading new model files into the production system, see the *ADMS Installation Guide*.

The process for populating the operations model is shown in the following image.

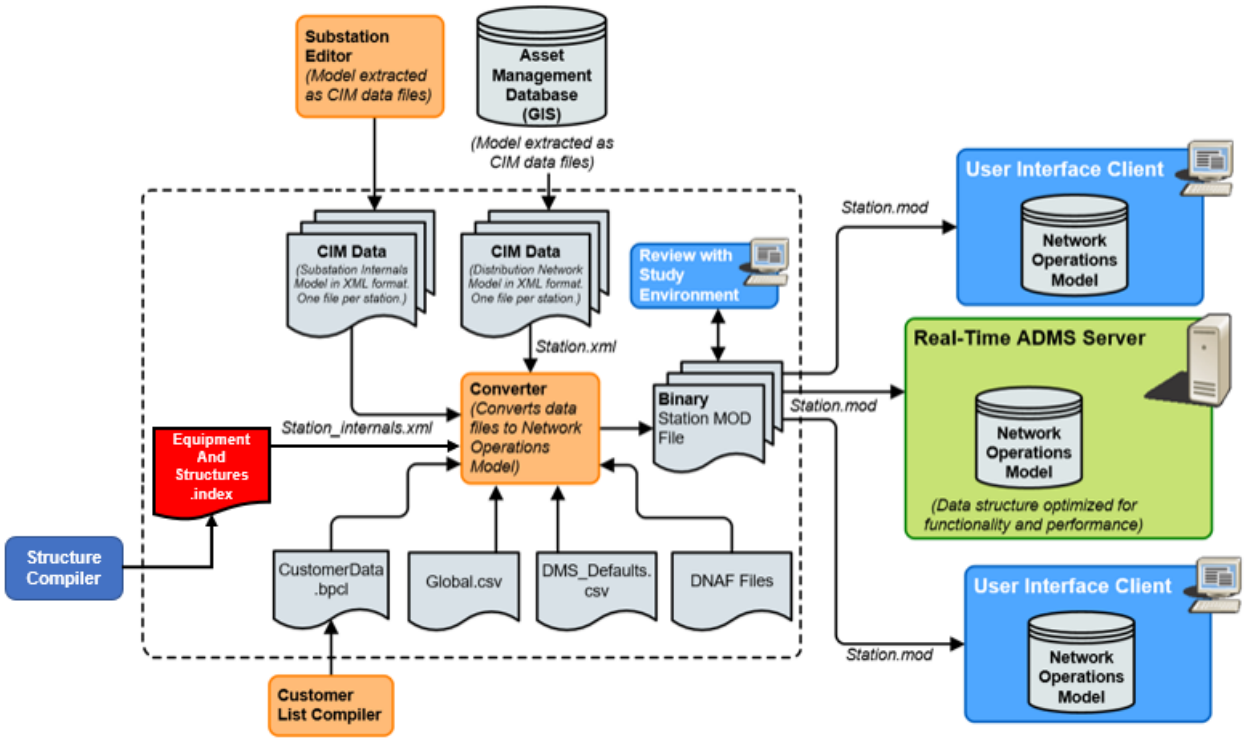


Figure 2. Process Flow for Populating the Operations Model

1.2.2. Connecting Station Internals and Station Externals Files

To connect a station internals file with a station externals file, the Substation Editor uses the Connection Point object. The Connection Point object can be used to represent feeder head connections by setting the connection node ID to a node ID defined within the station externals file, and setting the Feeder Connection option to True.

1.2.3. Boundary Branch Modeling

In the DMS model, connections between adjacent distribution stations are made through boundary branches. These boundary branches, commonly switching devices (although they can be any branch device, including lines/cables, series reactors, or transformers), have a field that indicates the connecting or adjacent station. The boundary branches are the same as regular non-boundary branch devices except for the value set in the connecting station field ("ToStation" parameter of the station externals file). The connecting station field is the globally unique and persistent station name value. For regular branches, the connecting station field is left blank.

For example, if there were a normally open disconnect switch, SwitchAB, between two stations, station A and station B, this boundary switch must be defined in both station externals files. The switch definition would be identical in each file except for the connecting station field. In station A, the connecting station for SwitchAB would be set to Station B; and in station B, the connecting station for SwitchAB would be set to Station A.

During model conversion, the Converter optionally checks to see if the station ID is defined in the station mapping table (record SubName) in the global.csv file, to make sure that a valid station has been specified. If the station ID has been used in the connection station fields instead of the name, the converter attempts to translate the connection station ID to the connecting station name. A message is logged if the connecting station name is not found. In the DMS online model, the boundary branch topology processing for both the server and client is handled using the station name.

If a new station is added or a station ID/Name is changed, the appropriate changes must first be made in the `global.csv` file. After the global file has been updated, all impacted stations, including the adjacent stations, must be converted and then placed into the DMS. If the necessary modeling changes have not been made, the station boundary connections do not function properly in the DMS, impacting tracing and all network applications.

1.2.4. Device ID Namespace and Considerations

A device's ID must be unique within its substation. The namespace for an ID is the substation ID.

In addition, there are assumptions in the product, such as in the implementation of the Find functionality, that a device ID is unique across the system.

When a device ID is no longer used in a substation, ADMS orphans any dynamics associated with it at model load time. This is also true when a device ID is effectively "moved" from one station to another, which can occur through updates in a GIS or other external model file input sources (for example when changing the nominal state of switches to reconfigure a feeder's location from being in one substation to another).

Temporary devices (such as cuts and jumpers), annotations, and dynamics are orphaned by ADMS should the device they apply to be deleted from the model. Since moving a device from one substation to another equates to deleting a device and creating a new one, any dynamics, annotations, and tags attached to the deleted device will not be applied to the new device. The operator might lose obvious visibility to tags that were placed on the device when it belonged to the previous station.

Extreme care should be used when designing and implementing business processes and automation that may impact electrical topology in the model files, especially when a device is dropped from one substation's model and added to another substation's model.

1.2.5. Reference Bus Energization

In the DMS, the energization of the reference bus interface objects must be properly monitored to reflect a transmission or sub-transmission system outage. The monitoring is performed via a switch that is connected to each reference bus interface object.

Two methods for modeling the reference bus interface switch are currently supported:

- The switch with an associated energization status measurement is explicitly modelled in the station internals file.

  If the switch that is connected to each interface bus is explicitly modeled by the user, the flag `IsInterfacewithSwitch` is set to True for the reference bus interface object in the station internals file. With this flag set to True, the Converter iterates the interface bus node and checks for the connected switch. The Converter also checks that a status measurement of the `MeasEnergized` measurement type is associated with this switch.

  If the energization measurement is not associated with the switch, a fatal message, FWF522, is issued. In the real-time DMS, the energization status for the switch is set by SCADA. If the switch's MeasEnergized status is False, which means a transmission system outage, the DMS opens the switch to disconnect the downstream distribution network. If the status changes to True, the DMS re-energizes the downstream distribution network.

- The Converter automatically inserts a normally closed switch and checks the associated voltage measurement to monitor the status of the switch.

  If the customer does not model the switch, the Converter automatically inserts a normally closed pseudo switch that is internally connected to the reference bus interface object. The status of this switch is used to determine the real time interface bus energization status. The user cannot see this switch on the geographical UI, but the device is listed in the Abnormal Summary Display if it is open.

  The Converter also sets the interface voltage flag to true for the voltage magnitude measurements (those connected to a reference bus interface directly or via zero impedance links, such as a jumper, or close to a reference bus interface), if the voltage level at the voltage measurement node is greater than 80% of the source side power transformer voltage. The real-time server energization function only processes voltage measurements that have the interface voltage flag set to true. This helps improve server and network application performance.

  In the real-time DMS, the server checks the voltage measurements that have the interface voltage flag set to true. If the measurement's value is near zero (configured by the `EFFECTIVE_ZERO_VOLTAGE_MEASUREMENT_PERCENTAGE` parameter in `NetServerMainConfig.txt`), a distribution station outage is identified. The DMS finds the correct pseudo interface switch connected to the interface bus and opens it. If the voltage measurement value changes from zero to a non-zero value, the pseudo interface switch is closed and the system is re-energized.

The Substation Editor (SSE) application is commonly used for modeling the electrical network inside the substation fence. Reference bus interfaces are commonly modeled in the SSE, but this tool is not required if the correct device format is given (an XML file format is used to represent the station internals electrical model).

In some distribution networks, lower voltage substations are occasionally energized by higher voltage circuits/feeders. In this connectivity arrangement, the lower voltage station is referred to as a sub-fed station. The voltage levels for these stations are anywhere from 4.16 kV to 13.2 kV (higher voltage levels are possible, but these are common North American distribution voltage levels). The sub-fed station is connected to the higher voltage source feeder through a normally closed boundary switching device.

A common mistake made at model build time is the addition of a reference bus interface at the source side of the step-down transformer in the sub-fed (lower voltage) station. This interface should not be added because the energization for a sub-fed station is provided by the higher voltage feeder or set of feeders. These higher voltage feeders are fed from an HV/MV station that is energized from a real reference bus interface.

To allow for a sub-fed station to be analyzed without the energizing higher voltage feeder/station, there is the concept of a study only reference bus interface object. This object is automatically added to sub-fed (lower voltage) stations at the source side of the power transformer by the Converter (or Model Manager). These objects are only evaluated if the energizing higher voltage station is for some reason not in the model, which is an abnormal case. Typically, the study environment is set-up by initializing from real time, which means the energizing higher voltage station would be loaded into the study environment and the study interface would not be needed (not used in the connectivity model or network applications).

1.2.6. Fault Indicator Modeling

The GE DMS model supports both regular fault indicators and directional fault indicators. For a regular fault indicator, its state is set if a fault occurs downstream of its connection node. If it is telemetered (SCADA), a status measurement with type `FaultStatus must be mapped`. The real time NTP determines the faulted area based on the fault status measurement values. The fault indicator halo is shown in the faulted network and persists for the period defined by the maximum waiting time (NTP_FAULTIND_RESET_TIME in the Simulator NetSrv configuration file). More details about the fault indicator halo can be found in the Fault Location chapter of the *Distribution Network Analysis Functions (DNAF) Standard Applications Analyst Guide*. Once the faulted network is identified, it could also trigger the DMS Fault Isolation and Service Restoration (FISR) advanced application. See details about fault indicator triggering functionality in the *Distribution Network Analysis Functions (DNAF) Standard Applications Analyst Guide*.

The DMS also supports modeling Directional Fault Detectors (DFD), a smart device which can determine fault direction based on phase opposition between transient current and voltage. A Boolean field named **Directional** on the `FaultDetector` object is set to True when modeling DFD. A four state status measurement is also mapped to DFD in the SCADA Model. The status measurement type is `FaultStatusMultiState`. The four states are 00 – UNSET, 01-SIDE1, 10-SIDE2, and 11-SET. A DFD with state 11 means a three phase fault is located downstream from the device, and 00 means a three phase fault is in its upstream network. A DFD with state 01 or 10 means a single phase fault has occurred. The fault is located at the downstream side of the DFD with a state value of 01, and is located at the upstream side of the DFD with a state value of 10. The real time NTP determines the faulted area based on these different states. The fault indicator halo functionality operates in the same manner as when regular fault indicator halos are used.

1.3. Model Files

The following set of files contains the DMS model information:

- **Station Externals Files:** A station externals file generally describes the portion of the distribution network that is located outside of the substation fence, including feeders and laterals. The file contains facilities data for devices such as lines, switching devices, and transformers. This file is named `StationName.xml`; the CSV file format is also supported.
- **Station Internals Files:** A station internals file generally describes the portion of the distribution network that is located inside the substation fence, including switches, breakers, feeders, buses, and power transformers. This file also contains analog and status measurement data and facilities data for a balanced three-phase network. This file is named `StationName_INTERNALS.xml`. It is generated by the Substation Editor.
- **Global Resource File:** Contains equipment models that are referenced by one or more objects in the station externals or station internals files. The models include switch models, transformer models, and conductor models. This file has the name `global.csv`.

  If the Substation Editor is used to generate the station internals file, another global resources file may optionally be created. This file is named `global_internals.xml`.
- **DMS Defaults File:** Contains default data for selected fields that have not been provided within the station externals file. This file has the name `DMS_defaults.csv`. This is an optional file (it is not required for model validation and conversion).
- **Relay and Impedance Data File:** Contains phase and ground trip Amp settings for breakers and reclosers. This is an optional file (it is not required for model validation and conversion).
- **Protection Setting Group File:** Contains protection settings group information for breakers and reclosers. This is an optional file (it is not required for model validation and conversion).
- **Regulation Control File:** Contains regulation control definitions and an associated collection of regulation mode objects for transformers. This is an optional file (it is not required for model validation and conversion).
- **Capacitor Control File:** Contains information for capacitors. This is an optional file (it is not required for model validation and conversion).
- **Automation Scheme Control File:** Contains automation team and scheme control data. This is an optional file (it is not required for model validation and conversion).
- **DNAF-Related Files:** Files required to run the Distribution Power Flow (DPF) analysis during the model conversion. These are the .dat files: `alg.dat`, `msgdef.dat`, `optalg.dat`, `phid.dat`, and `tftype.dat`. The `DNAF_PARAMS.xml` file is optionally read and used in the Converter's DPF solution.
- **Customer File:** The customer file is needed for obtaining customer record information used in the Outage Management System. Also, it may contain the customer load data used for defining the loads in the network model.
- **Equipment and Structures Index File:** The equipment and structures index file is used to determine equipment sharing structures with equipment from different feeders.

The format of these files is described in the next sections.

1.3.1. General File Format

For CSV files (for example, `global.csv` and `DMS_defaults.csv`), the first field in each record of the file is the name of the object. The names of these objects, along with the detailed object descriptions, are referenced in the *ADMS Modeling Reference Tables*.

There is also a comment record, which means that the entire line is ignored when processed by the Converter.

The second field in each record (except for comment records) is the header record indicator. If a record is a header record, the value for this indicator is "0". The header record provides the names of the fields for a specific record. There can only be one header record for each object type in the file. All instances of a specific record, also called data records, have a header record indicator that is set to "1". All data records must follow the format provided in the associated header record.

For all XML files (including the station internals, station externals, and `global_internals.xml` files), there are no header records.

1.3.2. Station Externals Files

There are several object types in a station externals file (for example, `RAVEN.XML`). These objects can be divided into four general categories: grouping objects, branch objects, shunt objects, and non-electrical objects.

The grouping objects include Operational Area, Substation, Feeder, Transformer Bank, Composite Switch, and UI Container. Grouping objects collect associated objects either for ownership representation or UI/model representation.

The branch objects include Lines (including cables), all switching devices (Switches, Fuses, Breakers, Reclosers, Sectionalizers, Network Protectors, and Permanent Jumpers), Transformers, and Series Reactors. Branch objects are represented in the connectivity model by two or three nodes (a three winding transformer is the only branch object that can have three nodes).

Shunt objects have only one connection node. They include Shunt Impedances (capacitors or reactors), Loads (energy consumers), and Generators. Note that Static VAR Compensators (SVCs) are currently not supported in the distribution model. Although fault indicators and bus objects only have a single connection node, they do not impact the topology of the distribution model and therefore are not shunt objects.

The non-electrical objects include Vaults, Switch Cabinets (currently supported), Manholes, Duct Banks, and Splice Boxes (will be supported in future releases). These objects are similar in function to grouping objects but represent physical utility assets.

The UI information for each DMS object is also contained inside the station file. For each geometry type, there is a group of UI placements. When a UI display is called up, all placements for that geometry are displayed.

Four types of placements are defined: poly-placement, point placement, label placement, and combination placement. A poly-placement uses multiple vertices to represent buses, lines, and other objects. A point placement and a label placement use a single vertex, a size, and a rotation

value to display symbol and label placements. A combination placement can be either a poly-placement or a point placement.

The following table shows the DMS objects and the associated placement types.

Table 1. DMS Object Placement Summary

Distribution System Object	Symbol Placement			Label Placement
	Poly-Placement	Point Placement	Combination Placement	
Substation	Yes*			Yes
Feeder				Yes
Bus	Yes			Yes
Line	Yes			Yes
Switching Device		Yes		Yes
Transformer Bank		Yes		Yes
Transformer		Yes		Yes
Series Impedance		Yes		Yes
Shunt Impedance		Yes		Yes
Static VAR Compensator		Yes		Yes
Load		Yes		Yes
Ground/Earthing Device		Yes		Yes
Generator		Yes		Yes
Composite Switch			Yes	Yes
Overcurrent Relay		Yes		Yes
Fault Indicator		Yes		Yes
Duct Bank	Yes			Yes
Vault			Yes	Yes
Switch Cabinet			Yes	Yes
Manhole				Yes
Splice Box	Yes			Yes
UI Container			Yes	Yes

(*) For the Substation object, only one vertex is currently supported. Rotation and size values are not specified like for a point placement.

For a complete field-by-field data description of the station externals files, refer to the *ADMS Modeling Reference Tables*.

1.3.3. Global Resource Files

The `global.csv` and `global_internals.xml` files mostly contain power flow application data, such as conductor and transformer impedances. The global resource files also contain the network elements models. For example, a composite switch model provides the list of valid states for multi-state switching devices.

Each field in the global resources files is a Boolean, a string, an integer, or a floating point number. The Converter does format type checking for all fields.

For a complete description of the fields contained within the global files, refer to the *ADMS Modeling Reference Tables*.

1.3.4. Station Internals Files

A station internals file (for example, `RAVEN_INTERNALS.XML`) contains the measurement data and facilities data for a balanced three-phase electrical network. The measurement data includes data for analogs, statuses, control points, and set points. The facilities data includes typical balanced three-phase network components, which can be found in the sub-transmission system or within the substation perimeter.

Each field in a station internals file is a Boolean, a string, an integer, or a floating point number. The Converter does format type checking for all fields.

Of special note within the station internals file is the distribution equivalent reference bus, which is represented in the model as a slack generator. The real-time DPF can use the distribution equivalent reference bus objects to obtain initial balanced voltage and angle values from the EMS state estimator, to be used as reference values in the DMS solutions.

For a data description of the station internals file, refer to the *ADMS Modeling Reference Tables* (the format is similar to a station externals XML file, however additional fields are output from the Substation Editor).

1.3.5. DMS Defaults File

The `DMS_Defaults.csv` file contains default data for selected fields that have not been provided within the station externals file.

Each field in the DMS defaults file is a Boolean, a string, an integer, or a floating point number. The Converter does format type checking for all fields.

For a complete field-by-field data description of the DMS defaults file, refer to the *ADMS Modeling Reference Tables*.

1.3.6. Relay and Impedance Data File

The `RelayandImpedanceData.csv` file contains various trip settings for breakers as well as system equivalent impedance values, which are used in short circuit analysis.

This file is optional. Trip settings defined in this file will override any trip settings defined on an individual relay or recloser.

1.3.7. Protection Setting File

The `ProtectionSettings.csv` file contains overcurrent protection settings and settings group templates for switching devices (typically, reclosers and breakers).

This file is optional. If defined, values in this file will override trip settings defined on an individual relay or recloser or defined in the `RelayandImpedanceData.csv` file (also optional).

However, any trip settings defined via SCADA measurements will override settings defined in a `ProtectionSettings.csv` file.

The overcurrent protection settings section includes the following data:

- **Settings Groups:** Each settings group represents a particular configuration for a protection device that contains a settings group name, phase trip and ground trip values. It also contains a SCADA index used to map the setting groups to measurements for control and status.
- **Source Groups:** This type of data is optional and is only used when settings groups are selected based on the sub/feeder combination supplying the switch. This option is defined in the FISR and AFR sections of the DNAF Configuration Editor. This data defines which sub/feeder supplies are associated with a specific settings group.

The `ProtectionSettings.csv` file includes the following overcurrent protection settings fields:

Field Name	Type	Description
SWITCH_OCPROT	Text	Name of the switching device.
SWITCHID_OCPROT	Text	Unique ID of the switching device.
NOTES_OCPROT	Text	User-defined text.
NUMSETTINGS_OCPROT	Integer	Number of setting groups to be defined on the device (the next four fields are defined for each group).
SETNAMEn_OCPROT	Text	Display name of the setting group (for example, "S1", "S2").
PHASETRIPn_OCPROT	Number	Phase Trip Current in Amps for this setting group (for example, "800").
GROUNDTRIPn_OCPROT	Number	Ground Trip Current in Amps for this setting group (for example, "400").

Field Name	Type	Description
SCINDEXn_OCPROT	Integer	Ordinal value of the SCADA index associated with this setting group. This value is used to map a setting group to the correct SCADA measurement for status, trip current, and control (for example, "1" for `StatusSettingGroupApplied1`, "2" for `StatusSettingGroupApplied2`, etc.). Note that in cases where status is contained in a single measurement (`StatusSettingGroupPrimAlt`), a value of "1" should be used for the primary setting group and "2" for the alternate.
NUMSOURCES_OCPROT	Integer	Number of source groups defined for the device (the next three fields are defined for each source).
SRCNAMEn_OCPROT	Text	The name of the setting group (see SETNAMEn_OCPROT) to be used when the device is fed from the specified source.
SOURCESUBn_OCPROT	Text	The name of the substation for this source group.
SOURCEFEEDERn_OCPROT	Text	The name of the feeder for this source group.

The settings group template section includes the following data:

- **Settings Group Template:** Users can have multiple protection settings templates for different types of reclosers and automatically assign template groups to individual reclosers in the Converter.
- **Protection Settings Group:** Protection settings group details, including available transitions between protection settings groups.

The `ProtectionSettings.csv` file includes the following settings group template fields:

Field Name	Type	Description
ID_SGTEMPLATE	Integer	ID of the template.
NAME_SGTEMPLATE	Text	Name of the template.
DESCRIPTION_SGTEMPLATE	Text	Description of the template.
NUMSETTINGSGROUPS_SGTEMPLATE	Integer	Number of settings groups associated with the template.
SETGRPNAMEn_SGTEMPLATE	Text	Settings group name, such as SG1 and SG2.
SETGRPIDn_SGTEMPLATE	Integer	Settings group ID.
SGVALUEn_SGTEMPLATE	Integer	Settings group value (index) from 1 to 8.
SGPHASETRIPn_SGTEMPLATE	Float	Phase trip setting in Amps.
SGGROUNDTRIPn_SGTEMPLATE	Float	Ground trip setting in Amps.
MODEn_SGTEMPLATE	Number	Switch mode (Switch, Sectionalizer, Recloser, or Custom).
TRANSITIONn_SGTEMPLATE	Integer	Supported transitions between settings groups.

1.3.8. Regulation Control File

The `RegulationControl.csv` file contains regulation control definitions and the associated collection

of regulation mode objects for transformers.

This file is optional. If no information is defined in this file, data defined in the regulator object will be used to model the regulator and only a single bi-directional or forward-only mode will be available. If data for a regulator is included in this file, it provides advanced modeling capabilities and overrides the data defined in the base regulator object.

The `RegulationControl.csv` file contains the following data:

- **Regulation Control Definition:** Each regulation control definition record represents a particular configuration for a transformer that contains an ID and a name of the regulator object, as well as a list of regulation modes included in the definition.
- **Regulation Mode:** This data includes detailed regulation parameters for different regulation modes, which are included in the control definition.

The fields in the file are as follows:

Field Name	Type	Description
REGNAME_REGCTRLDEF	Text	Name of the regulator object.
REGID_REGCTRLDEF	Text	ID of the regulator object.
DEFREGMODE_REGCTRLDEF	Text	Name of the default regulation mode.
NUMREGMODE_REGCTRLDEF	Integer	Number of modes defined.
REGMODEn_REGCTRLDEF	Text	Name of the mode.
NAME_REGMODE	Text	Name of the mode.
SCADAMEASVALUE_REGMODE	Integer	SCADA analog value for the mode.
FWDVTARGET_REGMODE	Number	Forward regulation target voltage setpoint (V).
FWDVBANDWIDTH_REGMODE	Number	Forward regulation full bandwidth (V).
FWDVREDUCTCAPABLE_REGMODE	Boolean	Forward regulation voltage reduction capable.
FWDVREDUCT1_REGMODE	Number	Forward regulation voltage reduction level 1 (V).
FWDVREDUCT2_REGMODE	Number	Forward regulation voltage reduction level 2 (V).
FWDVREDUCT3_REGMODE	Number	Forward regulation voltage reduction level 3 (V).
FWDLDCR_REGMODE	Number	Forward regulation LDC resistance (V).
FWDLDCX_REGMODE	Number	Forward regulation LDC reactance (V).
FWDTHRESHOLD_REGMODE	Number	Forward regulation flow reverse threshold. This is the value on the secondary side of CT (A).
FWDREGNODE_REGMODE	Enumeration	Forward regulation voltage node, either Primary or Secondary.
FWDREGULATIONMETHOD_REGMODE	Enumeration	Forward regulation method, value of Regulate, Idle, or RunToNeutral.
REVVTARGET_REGMODE	Number	Reverse regulation target voltage setpoint (V).
REVVBANDWIDTH_REGMODE	Number	Reverse regulation full bandwidth (V).
REVVREDUCTCAPABLE_REGMODE	Boolean	Reverse regulation voltage reduction capable.
REVVREDUCT1_REGMODE	Number	Reverse regulation voltage reduction level 1 (V).

Field Name	Type	Description
REVVREDUCT2_REGMODE	Number	Reverse regulation voltage reduction level 2 (V).
REVVREDUCT3_REGMODE	Number	Reverse regulation voltage reduction level 3 (V).
REVLDCR_REGMODE	Number	Reverse regulation LDC resistance (V).
REVLDCX_REGMODE	Number	Reverse regulation LDC reactance (V).
REVTHRESHOLD_REGMODE	Number	Reverse regulation flow reverse threshold. This is the value on the secondary side of CT (A).
REVREGNODE_REGMODE	Enumeration	Reverse regulation voltage node, either Primary or Secondary.
REVRUNNEUTRAL_REGMODE	Enumeration	Reverse regulation method, value of Regulate, Idle, or RunToNeutral.
REACTFLOWSENSE_REGMODE	Boolean	Regulate the source or load voltage based on the reactive current direction.

1.3.9. Capacitor Control File

The `CapacitorControlsData.csv` file contains capacitor control definitions. This file is optional. Capacitor data defined in this file overrides capacitor data defined in station models.

The fields in the file are as follows:

Field Name	Type	Description
NAME_SUB	String	Substation name.
ID_SHIMP	String	Capacitor unique ID. This field is used as a key for overriding capacitor data.
CONTROLABLE_SHIMP	Integer	Control capability enumeration (if any). The field has the following values: 0 = fixed, 1 = local only, 2 = remote only, 3 = remote with local override.
CTRLTYPE_SHIMP	Integer	Associated local controller type enumeration. The field has the following values: 0 = none, 1 = power factor, 2 = time dependent, 3 = temperature dependent, 4 = VAR dependent, 5 = current dependent, 6 = voltage dependent, 7 = time remote.
VREGLINELINE_SHIMP	Boolean	Indicates whether the voltage for the regulated (sensed) phases is measured line-to-line or line-to-ground (True = line-to-line).
REGBRANCH_SHIMP	String	ID of the regulation branch. Applies to VAR, Amp, or power factor locally controlled capacitors.
REGBRANCHTYPE_SHIMP	String	Regulation branch type (Line, Transformer, Switch, Breaker, Recloser, Fuse, Sectionalizer). Applies to VAR, Amp, or power factor locally controlled capacitors.
REGBRANCHDIRECT_SHIMP	Integer	Flow direction (enumeration). The field has the following values: 1 = out of device, 2 = into device. Applies to VAR, Amp, or power factor locally controlled capacitors.

Field Name	Type	Description
REGBRANCHEND_SHIMP	Integer	The end of the branch that is regulated (enumeration). The field has the following values: 0 = from side, 1 = to side, and 2 = tertiary (only if the branch is a three-winding transformer). Applies to VAR, Amp, or power factor locally controlled capacitors.
LOCALOVERRIDE_SHIMP	Boolean	Indicates whether the locally controlled capacitor has voltage override capability (True = has override capability).  This field is deprecated.
LOWLEVELOVER_SHIMP	Float	The low voltage override value (in per unit) for locally controlled capacitors that have an override feature. If the control value is below this value, the capacitor is turned on regardless of other controller settings.
HIGHLEVELOVER_SHIMP	Float	The high voltage override value (in per unit) for locally controlled capacitors that have an override feature. If the control value is above this value, the capacitor is turned on regardless of other controller settings.
ONLEVEL_SHIMP	String	Lower control setting for controller types 1, 3, 4, 5, and 6.
OFFLEVEL_SHIMP	String	Upper control setting for controller types 1, 3, 4, 5, and 6.
PHASESENSE_SHIMP	String	Phases that are measured for the local controller.
TIMESCHED_SHIMP	String	For time controlled banks, the associated on/off schedule name (in the global resources data).
TDELAY_SHIMP	Float	Controller time delay setting (seconds).
CTRLOVERRIDETYPE_SHIMP	Integer	Enumeration indicating the type of local controller override capability (0 = none, 1 = voltage, 2 = time, 3 = temperature).
OVERRIDESCHED_SHIMP	String	For time or temperature override control, the associated on/off schedule name (in the global resources data).
LOCALPERM_SHIMP	Integer	Enumeration indicating the type of local controller permissive operation (0 = none, 1 = time, 2 = temperature).
PERMSCHED_SHIMP	String	For time or temperature permissive control, the associated on/off schedule name (in the global resources data).
PERMCTRLLEVEL1_SHIMP	Float	The permissive control level 1 value for locally controlled capacitors that have a permissive feature. If the control value is out of the range defined by level 1 and level 2, the device is OFF.
PERMCTRLLEVEL2_SHIMP	Float	The permissive control level 2 value for locally controlled capacitors that have a permissive feature. If the control value is out of the range defined by level 1 and level 2, the device is OFF.
LOWTLEVELOVER_SHIMP	Float	The low time or temperature override value for locally controlled capacitors that have an override feature. If the control value is below this value, the capacitor is turned on regardless of other controller settings.
HIGHTLEVELOVER_SHIMP	Float	The high time or temperature override value for locally controlled capacitors that have an override feature. If the control value is above this value, the capacitor is turned on regardless of other controller settings.

Field Name	Type	Description
REGUSEUNREGULATEDNODE_SHIMP	Boolean	Flag indicating the capacitor senses the unregulated node voltage (True = use unregulated voltage). This voltage is effectively located at the transformer secondary terminal, which is not modeled, and is typically upstream of the Load Tap Changer and the transformer secondary node (ToNode).
REGTF_SHIMP	String	For capacitors that sense the unregulated node voltage, this field is the associated transformer ID. For other capacitors, this field can be left blank.

1.3.10. Automation Scheme Control File

The `AutomationSchemes.csv` file contains automation control team and scheme definitions. This file is optional. Automation scheme data defined in this file enables defining automation scheme objects.

The file includes the following data:

- **Automation Schemes:** Automated local peer-to-peer switching schemes used to automate the fault isolation and service restoration functions. A scheme includes a set of switching steps or actions involving one or more teams to isolate a fault and restore faulted sections.
- **Automation Scheme Teams:** A team includes two or more switching devices (with or without SCADA telemetry or control) that operate together within a switching scheme.
- **Automation Scheme Team Switches:** Data for switching devices that are part of switching scheme teams. A switching device can belong to one or more team, but only to one scheme. A switching device must be a part of the primary network.

The automation scheme fields in the file are as follows:

Field Name	Type	Description
ID_SCHEME	String	ID of the automation scheme.
NAME_SCHEME	String	Name of the automation scheme.
DESCRIPTION_SCHEME	String	Description of the automation scheme.

The automation scheme team fields in the file are as follows:

Field Name	Type	Description
ID_TEAM	String	ID of the automation scheme team.
SCHEME_TEAM	String	ID of the associated automation scheme.
NAME_TEAM	String	Name of the automation scheme team.
DESCRIPTION_TEAM	String	Description of the automation scheme team.

The automation scheme team switches fields in the file are as follows:

Field Name	Type	Description
ID_TEAMSW	String	ID of the switching device.

Field Name	Type	Description
TEAM_TEAMSW	String	ID of the associated automation scheme team.
NAME_TEAMSW	String	Name of the switching device.
SECONDID_TEAMSW	String	Alternate ID of the switching device.
STATION1_TEAMSW	String	The first station that is associated with the switching device.
STATION2_TEAMSW	String	The second station that is associated with the switching device. (For boundary switches.)
ROLE_TEAMSW	String	Switching device role (primary or secondary).

1.3.11. DNAF-Related Files

The DNAF-Related Files include the following:

- **alg.dat:** Contains data on algorithm control parameters.
- **msgdef.dat:** This is a message definition file.
- **optalg.dat:** Contains algorithm options for optimization.
- **phid.dat:** Contains phase identifiers.
- **tftype.dat:** Contains transformer type, impedance, and connectivity data.
- **DNAF_PARAMS.xml:** Defines parameters for DNAF solutions. This file is optionally read and used in the Converter's DPF solution.

Using these files, the Converter executes the Distribution Power Flow analysis used for model validation and quality gate evaluation.

1.3.12. Customer File

The `Customer.bpcl` file contains information related to each customer account in the system. This customer information is primarily used in the Outage Management System. In addition, the customer file may contain data on customer loads (if these loads are not defined in the station externals file).

This customer binary file is created by the Customer List Compiler after processing the customer input file (refer to the *Customer List Compiler User Guide)*.

1.3.13. Equipment and Structures Index File

The equipment and structures index file is generated from exported *.geojson files using the `StructureCompiler.exe` tool.

1.4. Substation Editor

The Substation Editor application is used to generate the station internals files, including device data, connectivity data, and associated measurement data.

The measurements can be linked to devices inside or outside of the station fence (devices that are defined in either the station internals or station externals files). The output of the Substation Editor is a complete station internals file and optional placement information files.

For more information about using the Substation Editor, refer to the *Substation Editor User Guide*.

2. Converter Tool

The ADMS Converter is a Windows-only application that reads in the model files, validates the data, and writes out the data into a binary file. This resulting binary file is called a station model file. Station model files have the extension ".mod".

From the station model file, the topology processing application can be executed and, if equipment placement information is modeled, UI displays can be automatically generated.

To start the Converter application from the Microsoft Windows Start menu, navigate to **All Programs > Eterra > Distribution > Tools > Modeling Tools > Converter.exe**.

When the Converter application starts, it automatically opens the default configuration file `converter.xml`, located in the `D:\Eterra\Distribution\FantasyIsland\ToolsWorkspace\Converter` folder.

2.1. Displays Overview

This section provides a description of all Converter displays.

2.1.1. File Menu and Toolbar

The Converter's File menu includes the following options:

- **New Config:** Opens a new blank configuration file.
- **Open Config:** Invokes the Open dialog box to browse to a saved configuration file.
- **Save Config As:** Saves the XML configuration file under a different name.
- **Convert CSV to XML:** Allows an analyst to open the desired CSV file and convert it to XML format.
- **Export Message Definition to XML:** Exports the definition/scope of all messages that can be written out by the Converter.
- **Exit:** Closes the Converter tool.

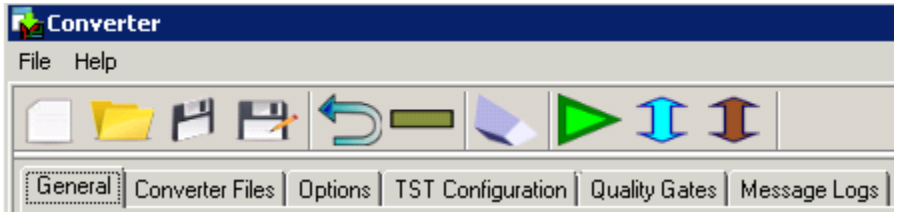


Figure 3. ADMS Converter - Toolbar

The Converter's toolbar includes the following buttons:

- **New Config:** Opens a new blank configuration file.

- **Open Config:** Invokes the Open dialog box to browse to the saved configuration file.
- **Save Config:** Saves the currently open XML configuration file.
- **Save Config As:** Saves the currently open XML configuration file under a different name.
- **Reset Config:** Click to discard changes (unsaved, and not applied) to the XML configuration file.
- **Apply Config:** Select this option to apply the currently open XML configuration file. The applied changes cannot be reset.

Applying changes does not save them. To save the changes, use the Save Config, or Save Config As option.

- **Convert CSV to XML:** Allows an analyst to open the desired CSV file and convert it to XML format.
- **Convert:** Click this button to start the conversion process for the selected substations.
- **Select All:** Click this button to select all the substations in the Substation list on the General tab.
- **Clear All:** Click this button to clear all selections.

2.1.2. General Tab

The General tab is used to select substations for conversion and to provide information about converter input files.

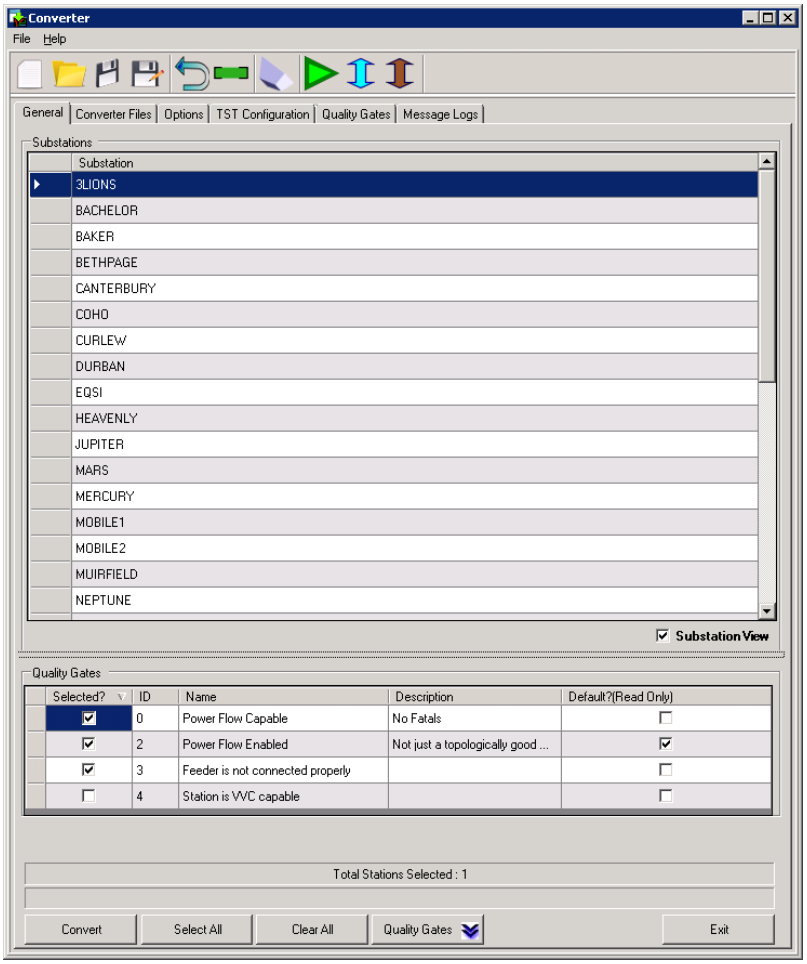


Figure 4. ADMS Converter - General Tab

The General tab includes the following fields and buttons:

- **Substation View:** Switches the view between a simple list of substations and a detailed view of each of the input files.
- **Convert:** Click this button to start the conversion process for the selected substations.

  The station models to be converted may be selected by clicking on individual substations or by clicking the Select All button.
- **Select All:** Click this button to select all the substations in the Substation list.
- **Clear All:** Click this button to clear all selections.
- **Quality Gates:** Displays the Quality Gates pane, from which station quality gates can be assigned to selected stations. All available quality gates are defined on the Quality Gates Tab.
- **Exit:** Click this button to close the Converter tool.

2.1.3. Converter Files Tab

The Converter Files tab is used to manage the Converter folder paths.

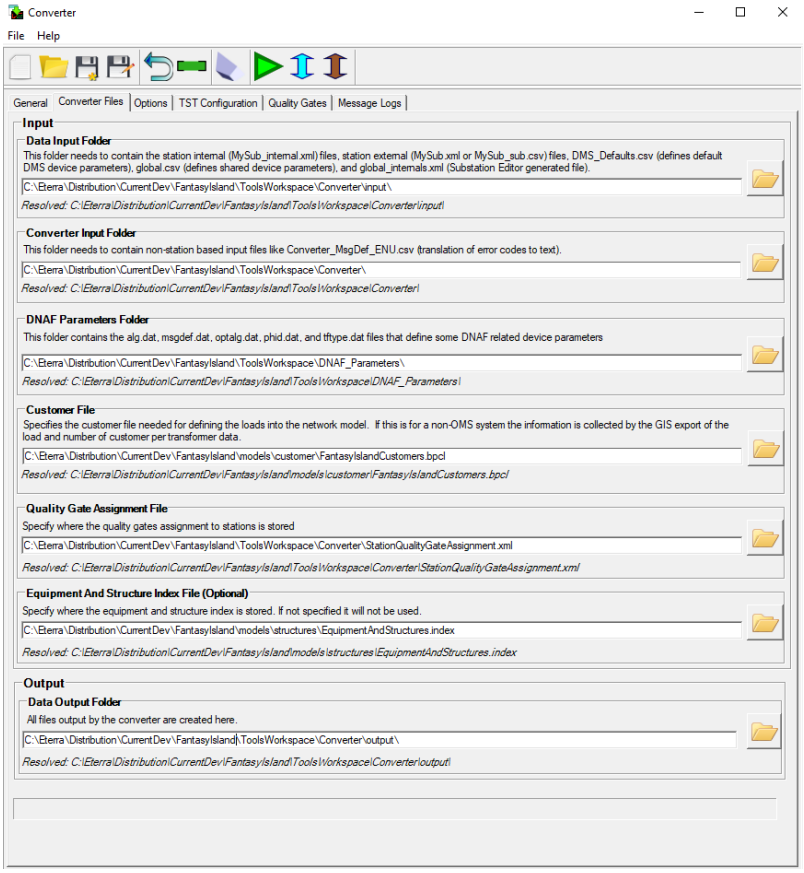


Figure 5. ADMS Converter - Converter Files Tab

The Input group of the Converter Files tab contains the following fields:

- **Data Input Files:** Sets the input folder for:
  - **StationName_internals.xml:** Station internals files.
  - **StationName.xml:** Station externals files.
  - **DMS_Defaults.csv:** Defines default DMS device parameters.
  - **global.csv:** Defines shared device parameters.
  - **Global_Internals.xml:** Substation Editor generated file.
- **Converter Input Files:** Sets the input folder for the non-station-based input files, such as `Converter_MsgDef_ENU.csv`, which contains the translation of error codes to text.
- **DNAF Parameters Folder:** Sets the input folder for the power flow files (which include `alg.dat`, `msgdef.dat`, `optalg.dat`, `phid.dat`, `tftype.dat`, and `DNAF_PARAMs.xml`) that define some DNAF-related algorithm parameters.
- **Customer File:** Specifies the customer file containing the customer data used in the OMS.

The customer file may contain average load (kW) or average monthly power consumption (AveragekWh) data, which can be used to create loads in the Network Model at the Service Points. In addition, the Converter can populate load ID, name, and address. For non-OMS systems, the customer file may not be available, in which case the loads must be defined in the station externals file.

- **Quality Gate Assignment File:** Specifies the Quality Gate Assignment file that can be used to assess the quality of the model based on the Converter validation messages logged during model creation.
- **Equipment And Structure Index File (Optional):** Specifies the Equipment and Structure index file that can optionally be used to generate halos for equipment that share structures with equipment from different feeders. The index file is generated by the StructureCompiler.exe tool. This file is optional. If the Structure Information feature is not enabled, this property can be left empty to indicate that it is not being used.

The Output group of the Converter Files tab contains the following field:

- **Data Output Files:** Sets the folder for all the Converter's output files.

2.1.4. Options Tab

The Converter options can be set on the Options tab.

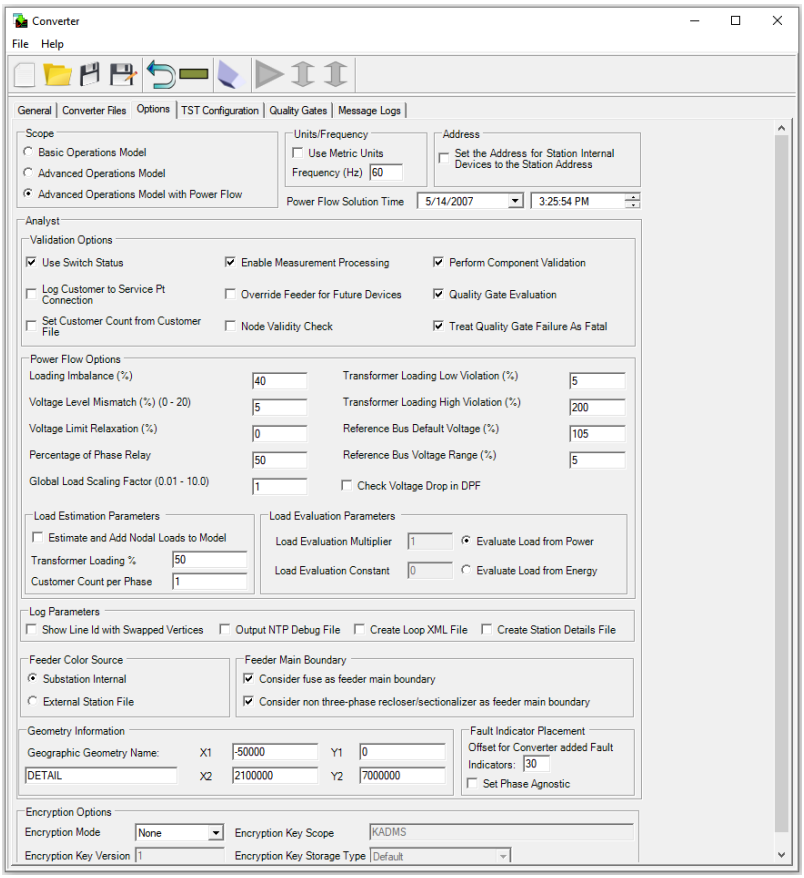
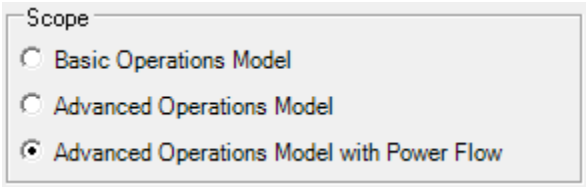


Figure 6. ADMS Converter - Options Tab

2.1.4.1. Scope

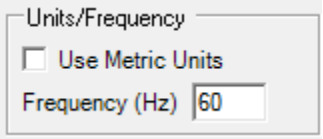


The Converter program supports three scopes. The analyst can choose one of the scope options by selecting the corresponding option on the Scope pane.

- **Basic Operations Model:** In this scope, only data required for operation of the Network view with topology processing is validated. The data that is necessary for power flow–based applications is not validated, and the network analysis applications cannot be executed on the resultant station model binary file.
- **Advanced Operations Model:** In this scope, basic validation of power flow data is added. The network analysis applications cannot be executed on the resultant station model binary file.
- **Advanced Operations Model with Power Flow Solution:** In this scope, a power flow execution is added, which provides validation to ensure that the model is well-conditioned to be used by

the network analysis functions. If it passes the necessary validation checks, the network analysis applications can be executed on the resultant station model binary file.

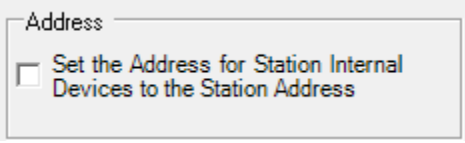
2.1.4.2. Units/Frequency



This group includes the following options:

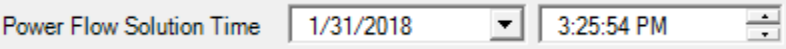
- **Use Metric Units:** Specifies whether metric units are used. For example, the temperature is specified in Celsius instead of Fahrenheit, and line lengths are in meters vs. feet.
- **Frequency:** Specifies the frequency value of distribution stations (in Hz).

2.1.4.3. Address



This option specifies whether the address for station internal devices is set to the station address.

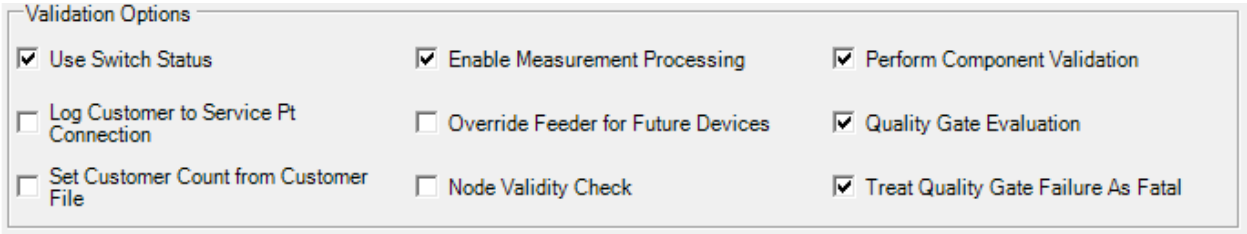
2.1.4.4. Power Flow Solution Time



This group includes the following option:

- **Power Flow Solution Time:** The solution time used by the Converter when running Distribution Power Flow calculations as part of model validation. When needed, the solution time can be manually updated by the user.

2.1.4.5. Validation Options



This group includes the following options:

- **Use Switch Status:** Option used to determine whether the normal state of the switching device should be considered when checking the node connectivity. This option should always be selected.
- **Perform Component Validation:** Option that enables or disables component validation range checking for numeric attributes. This option should always be selected.
- **Enable Measurement Processing:** Option that enables or disables processing of measurements. This option only impacts cases when there is a station internals file (this file contains the measurements). This option should always be selected.
- **Process Customer Data:** Option that enables or disables validation of customer files. If selected, this option checks if a customer number in a customer file matches the number in a station file for the same service point. It also writes the matched service point along with the customer number and the meter number into another log file, station_customer.log, which is in the same folder as the converter log file.
- **Set Customer Count from Customer File:** If selected, for all service points with single loads, if the number of customers in the station external file does not match the number in the customer file, the number from the customer file is used for that load/service point.

  The customer count is not altered for service points with multiple loads, because an incorrect number of customers could be assigned to different loads/phases.
- **Override Feeder for Future Devices:** If selected, the nominal feeder assignment (determined by the Network Topology Processor for topology devices) marked as Future is overridden by the feeder given in the Nominal Feeder field in the static model (for example, NOMINAL_FEEDER_ID_LINE). For field names in CSV/XML format, refer to the *Modeling Reference Tables*.
- **Node Validity Check:** If selected, the Converter checks for node mismatches in step transformers and regulators. If a mismatch is found, the Converter adjusts the associated nodes automatically.
- **Quality Gate Evaluation:** Option to run the Quality Gates Evaluation during conversion (see Using Quality Gates).
- **Treat Quality Gate Failure as Fatal:** Option for the Converter to treat the quality gate failure as a fatal error during conversion.

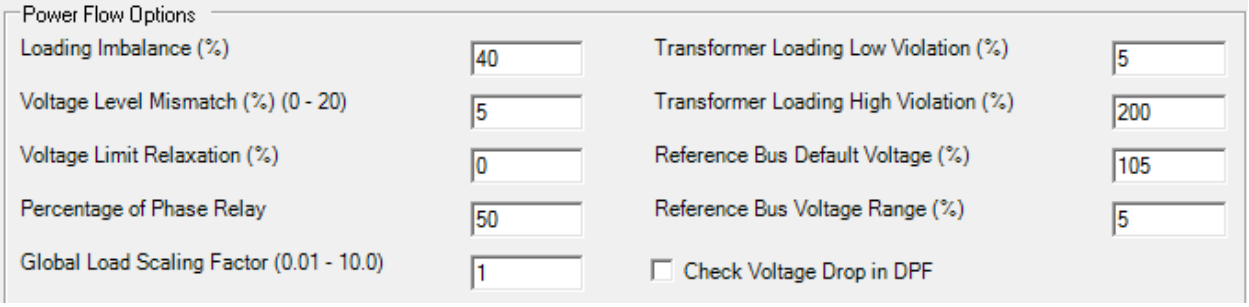
2.1.4.5.1. Phase Augmentation Options

Phase Augmentation adds jumpers to bridge missing downstream phases for switching devices and regulators.

This group includes the following options:

- **Phase Augmentation:** Determines whether jumpers will be added to bridge missing downstream phases.
- **Default Switch Model:** Displays the default assigned Switch Model for jumpers.

2.1.4.6. Power Flow Options



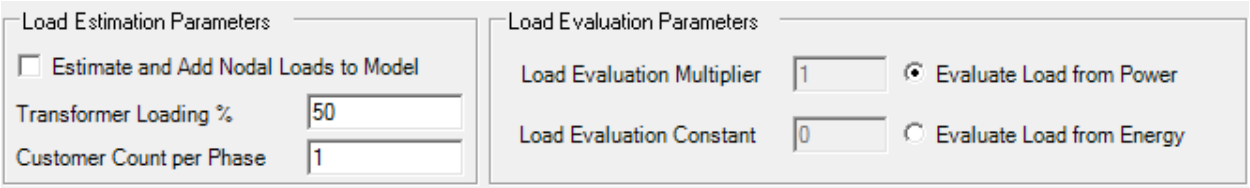
This group includes the following options:

- **Loading Imbalance (%):** Loading imbalance (in percent) at which a warning is reported by the Converter running on power flow scope. The loading imbalance is the difference between highest-loaded and lightest-loaded phases for three-phase loads.
- **Voltage Level Mismatch (%) (0-20):** This is the maximum allowable voltage mismatch (in percent) between the rated equipment voltage and system operating voltage level. A Voltage Level Assignment (VLA) message is issued for each piece of equipment (transformer, capacitor, or reactors) that has a mismatch greater than this percentage.
- **Voltage Limit Relaxation (%):** The Voltage Limit Relaxation is used to change the load voltage limits that have been set in the global file. For example, if the high voltage limit is 1.06 p.u. (set under Load_Category) and if VoltageLimitRelax is set to 3 (percent), the new high voltage limit will be 1.09 p.u. Any power flow solved will be compared against this new voltage limit rather than the global voltage limit specified in the load category.
- **Percentage of Phase Relay:** Feeder head loading percentage if no peak feeder head amp value is provided in the station model file. For example, if the feeder breaker has a phase trip setting of 600.0 amps and the percentage of phase relay setting is 80.0, the bus load allocation uses 480.0 amps (0.80 * 600) when performing the load balancing.
- **Global Load Scaling Factor (0.01–10.0):** Load scaling factor. The nominal load scaling factor value is 1.0. The load P and Q values (represented by PNomA, QNomA, PNomB, QNomB, PNomC, and QNomC) are permanently scaled by the value given in this field. For example, if an A phase load has a PNomA value of 30 kW and a QNomA value of 10 kVAR, and the load scaling factor is set to 0.5, the load would be scaled down to (15 + j5) kVA. The newly scaled nominal load values are stored in the binary model file for use by the real time and study mode DNAF applications.
- **Transformer Loading Low Violation (%):** Option to set the minimum distribution transformer loading percentage.
- **Transformer Loading High Violation (%):** Option to set the maximum distribution transformer

loading percentage.

- **Reference Bus Default Voltage (%):** The scaling factor applied to the interface object voltage (i.e., nominal voltage is scaled by the reference bus voltage percentage and is applied to the interface object).
- **Reference Bus Voltage Range (%):** Reference bus voltage range to use when transmission side source voltage is back-calculated during tap estimations.
- **Check Voltage Drop in DPF:** Option to check for voltage drops for lines and transformers in the DPF. This is a good debugging tool to see if any abnormal voltage drops occur during DPF. If a voltage drop happens beyond a certain limit, error messages are logged, which can be used to spot-check the model.

2.1.4.7. Load Evaluation and Estimation Parameters



In the current architecture, DNOM load objects can be created in the following ways:

- Explicitly modeling loads in the station external file. The Converter reads these loads and creates DNOM load objects.
- Providing load information (kW or AverageKWh) in the customer file (.bpcl) and giving it as an input to the Converter. The Converter creates DNOM load objects from those customer records.
- Explicitly providing loads in the station external file and the customer file for the same service point. The Converter merges the two loads into a single DNOM load object.
- Using the "Estimate and Add Nodal Loads to Model" option. The Converter creates loads as follows:
  - If no loads are available in the station external file, the Converter creates DNOM load objects for every distribution transformer and transformer bank.
  - If loads are available in the station external file, the Converter creates loads only at service points that do not already have explicit loads.
  - If loads are available both in the station external file and the customer file, the Converter creates loads only at service points that do not have loads (from the station external file) or customer records (from the customer file).
  - If loads are not available in the station external file but the customer records are available in the customer file, the Converter creates loads only at service points that do not have customer records (from the customer file).

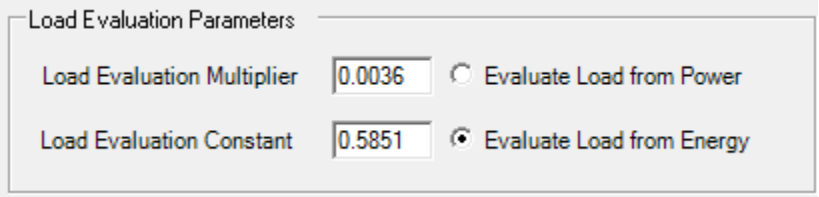
- Loads can also be entered in the station externals/internals files using the Load record. The node of these loads must be tied to an existing transformer

in the station. P and Q values for each phase can be entered using the Load records.

- There are instances where all the load data (power consumption) is present in the Customer *.bpcl file. Since, the Customer file does not have any geometry information, no load symbols get created (even for Primary Meters). To circumvent this, in the station external file, dummy Primary Meter loads (with "0" P and Q values) can be modeled with relevant geographic information at the same Service Point location. This would cause Primary Meter symbols to be shown on the DMS.

Load Evaluation Parameters

The Customer List Compiler tool provides options to calculate the average load (kW), which can be inserted at service points. It can also calculate the average monthly power consumption (AveragekWh). For more details, refer to the "Modeling Load Using Customer Compiler" section in the *Customer List Compiler User Guide*.



The load calculation can be further adjusted using the Converter tool as follows:

- If the Customer Compiler was used to calculate Average Load (kW), select "Evaluate Load from Power".

  When this option is selected, the Converter uses load values from the customer file.

- If the Customer Compiler was used to calculate Average Consumption (AveragekWh), select "Evaluate Load from Energy".

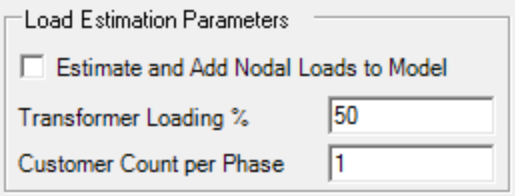
  When this option is selected, the Converter calculates load as follows:

  - The AveragekWh value is multiplied by the "Load Evaluation Multiplier". "Load Evaluation Constant" is then added to the resulting value to get the final average load (kW) value.

    The calculated values are inserted into the available phases across the relevant load.

Load Estimation Parameters

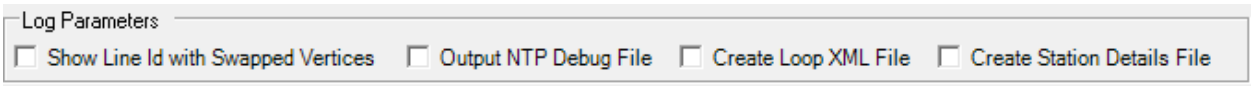
When the Estimate and Add Nodal Loads to Model option is selected, the loads are calculated based on the transformer rated kVA (multiplied by the "Transformer Loading %" parameter value). The obtained load objects are added to the model.



For example, if the Transformer Loading % value is set to 200, a 200% load will be created for every transformer in the station (i.e., a 50 kVA load will be created for a 25 kVA distribution transformer).

The Customer Count per Phase field sets the number of customers per transformer phase when the Estimate and Add Nodal Loads to Model option is used.

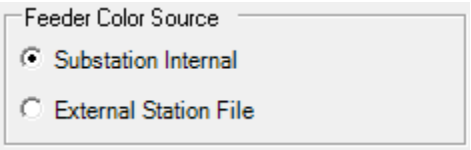
2.1.4.8. Log Parameters



This group includes the following options:

- **Show Line ID with Swapped Vertices:** Specifies whether the IDs of lines with swapped vertices are listed in the Converter log file.
- **Output NTP Debug File:** Specifies whether debug information is written to an `NTPD_CNVT_<TimeStamp>_<Station>_<Index>.txt` file.
- **Create Loop XML File:** Specifies whether information about loops is written to a `<Station>_Loop.XML` file. This includes a topology-ordered list of switching devices for every switch (typically a fuse) that has an associated Loop End Label object.
- **Create Station Details File:** Specifies whether information with Model Manager automated model review criteria is written to a `<Station>_StationDetails.dat` file. This includes backbone feeder lengths, distribution transformer counts (per feeder), feeder loads, feeder switching device counts, feeder customer counts, and downstream traced transformer and customer counts for feeder backbone switching devices. The Model Review feature in the Model Manager application compares differences in values in a generated model and values in the last generated model against configured threshold values to identify suspicious models that should to be manually reviewed. For more details, refer to the *Model Manager User Guide*.

2.1.4.9. Feeder Color Source

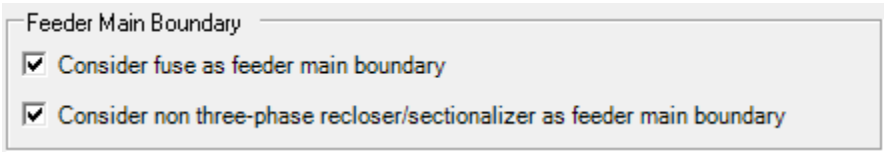


The source of the feeder color (an integer that is associated with a specific color in the Viewer Configuration Editor) can be entered either in the station externals file or in the station internals

file. The following options determine the color source.

- **Substation Internal:** Station internals file is the source of the feeder color.
- **External Station File:** Station externals file is the source of the feeder color.

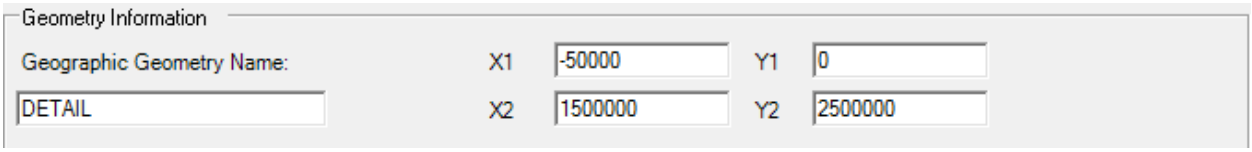
2.1.4.10. Feeder Main Boundary



This group includes the following options:

- **Consider fuse as feeder main boundary:** This option sets a fuse as a feeder main boundary device between the feeder main (backbone) and lateral. If this option is selected, any fuse and its downstream devices are considered to be lateral. If the FeedMainBound field is set for a particular fuse in the externals file, the FeedMainBound will override the FuseFdrMainBound option for that particular device.
- **Consider non three-phase recloser/sectionalizer as feeder main boundary:** This option sets non-three-phase reclosers and sectionalizers as feeder main boundary devices between the feeder main (backbone) and lateral. If this option is selected, any non-three-phase recloser/sectionalizer and its downstream devices will be considered as lateral. If the FeedMainBound field is set in the externals file, the FeedMainBound field's value will override the Non Three-Phase Recloser/Sectionalizer as the Feeder Main Boundary option for that particular device.

2.1.4.11. Geometry Information



Geometry Information parameters are used to validate the coordinates for the geographic geometry. If an object's geographic UI placement falls outside of the range (a rectangle defined by the lower left point X1 and Y1 and the upper right point X2 and Y2), the placement is removed from the model and a warning message is issued in the Converter log.

This group includes the following options:

- **Geographic Geometry Name:** Name of the geographic geometry.

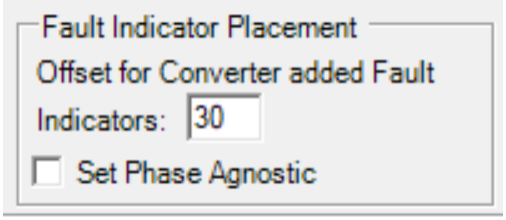
  This value is also set in the name of the Geographic Background Geometry field of the Viewer Configuration Editor's (VCE) General tab. In the VCE, this value specifies the geometry where the geographic background should be displayed (when different from the "geographic

background"). When this field is blank, the geographic background will be displayed on the geographic geometry.

The coordinate system used in the SCHEMATIC geometry is limited by the range for the Min and Max X and Y bounding rectangle coordinates in the Converter from -99,999,999 to +99,999,999.

- **X1, Y1** is the lower left bounding point.
- **X2, Y2** is the upper right bounding point.

2.1.4.12. Fault Indicator Placement

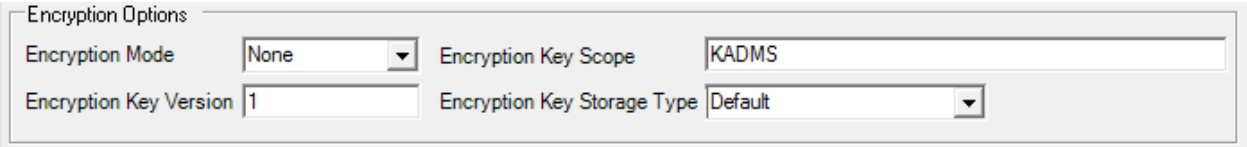


The Converter automatically adds fault indicators for telemetered breakers and reclosers that do not have the associated fault indicators in the station file (*.xml).

This group includes the following option:

- **Offset for Converter Added Fault Indicators:** Specifies the offset between the automatically added fault indicators and their associated devices. The value must be between 10 and 50 px.
- **Set Phase Agnostic:** Specifies whether fault indicators that are automatically added by the Converter are considered phase-agnostic (that is, the phase detected by the fault indicators during faults is disregarded).

2.1.4.13. Encryption Options



The station model file encryption requires an encryption key to be defined in the system. For more information, refer to the *ADMS Installation Guide*.

The encryption options specify whether the Converter creates encrypted station model files.

This group includes the following options:

- **Encryption Mode:** Specifies whether the Converter creates encrypted model files. Possible values are None, Encrypted, and Both.
  - **None:** If set to None, the tool does not encrypt model files. (Model files are created with the

usual ".mod" suffix.)

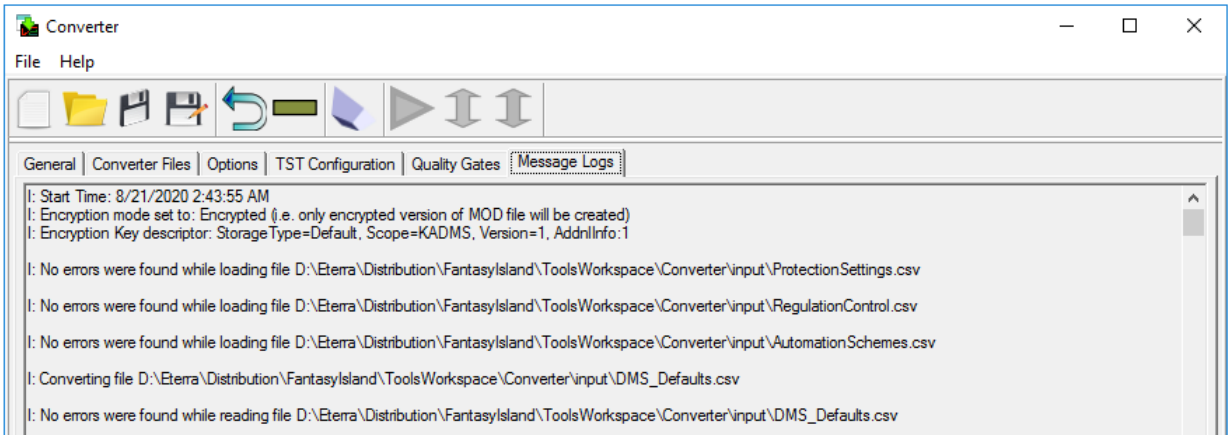
- **Encrypted:** Specifies that the tool encrypts model files. (Model files are created with the usual ".mod" suffix.)
- **Both:** Specifies that the tool produces both encrypted and unencrypted model files (the ".mod" files are created for encrypted models, and ".mod.unencrypted" files are created for not encrypted models). This mode can be used for testing and during migration phases.

- **Encryption Key Scope:** The scope property enables using different key names for different subsystems (for example, ADMS, Modeling, Test) or different sites (for example, GEADMS, TESTADMS, PROJECTADMS). The scope field maximum length is 11 characters.
- **Encryption Key Version:** The version property enables changing the encryption key to be used while maintaining the same `EncryptionKeyScope` value.

  The <Encryption Key Scope>_<Encryption Key Version> value is used as the entry to look up the actual value of the encryption key in a key storage. This value is also stored in the header of encrypted files, so that the value can be used to look for the encryption key when a file needs to be decrypted.
- **Encryption Key Storage Type:** Encryption key storage type. The storage type property designates where the key is stored. Different storage types are supported (Default, EnvironmentVariable, EnvironmentVariableDpApi). The value "Default" internally resolves to "EnvironmentVariableDpApi".

The encryption key scope, version, and storage type values must match attributes of the encryption key that is currently defined in the system.

When configuring encryption, check that the message log shows encryption information and does not contain errors.



To use encrypted files, the same encryption key must be defined for users that encrypt files and for users that run applications which use the encrypted files.

2.1.5. TST Configuration Tab

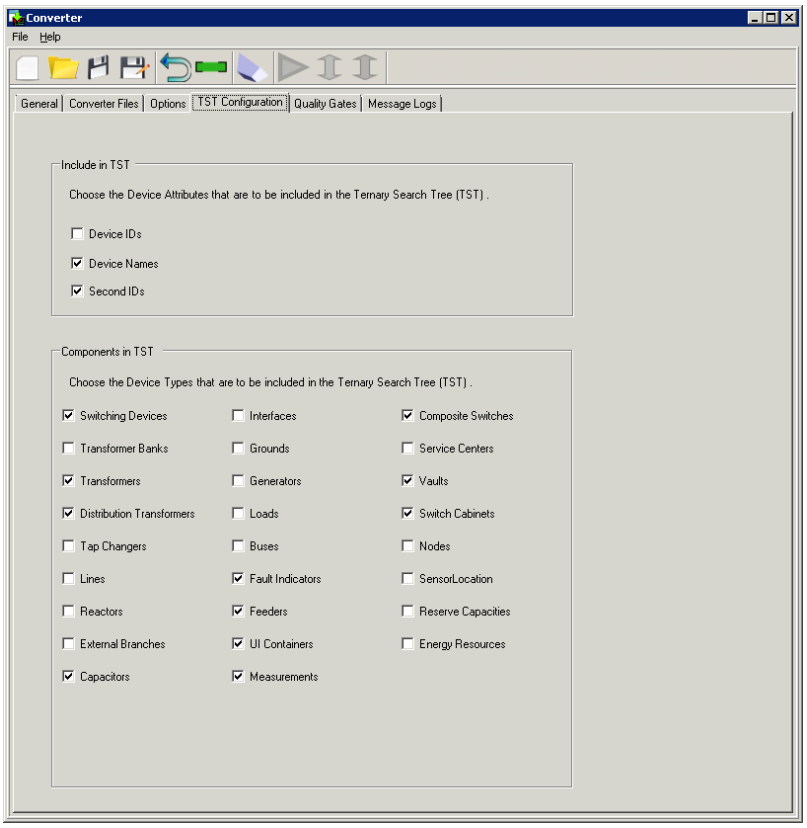


Figure 7. ADMS Converter - TST Configuration Tab

On the TST Configuration tab you can set the following configurations for the Ternary Search Tree (TST):

- **Include in TST:** Determines the DNOM search content for the ternary search tree: device IDs, device Names, or Second (alternate) IDs.
- **Components in TST:** Determines the object types that are added to the TST (impacts results of the DMS searches).

2.1.6. Quality Gates Tab

Quality gates can be used to prevent a model update based on Converter messages and user-defined criteria (see Using Quality Gates).

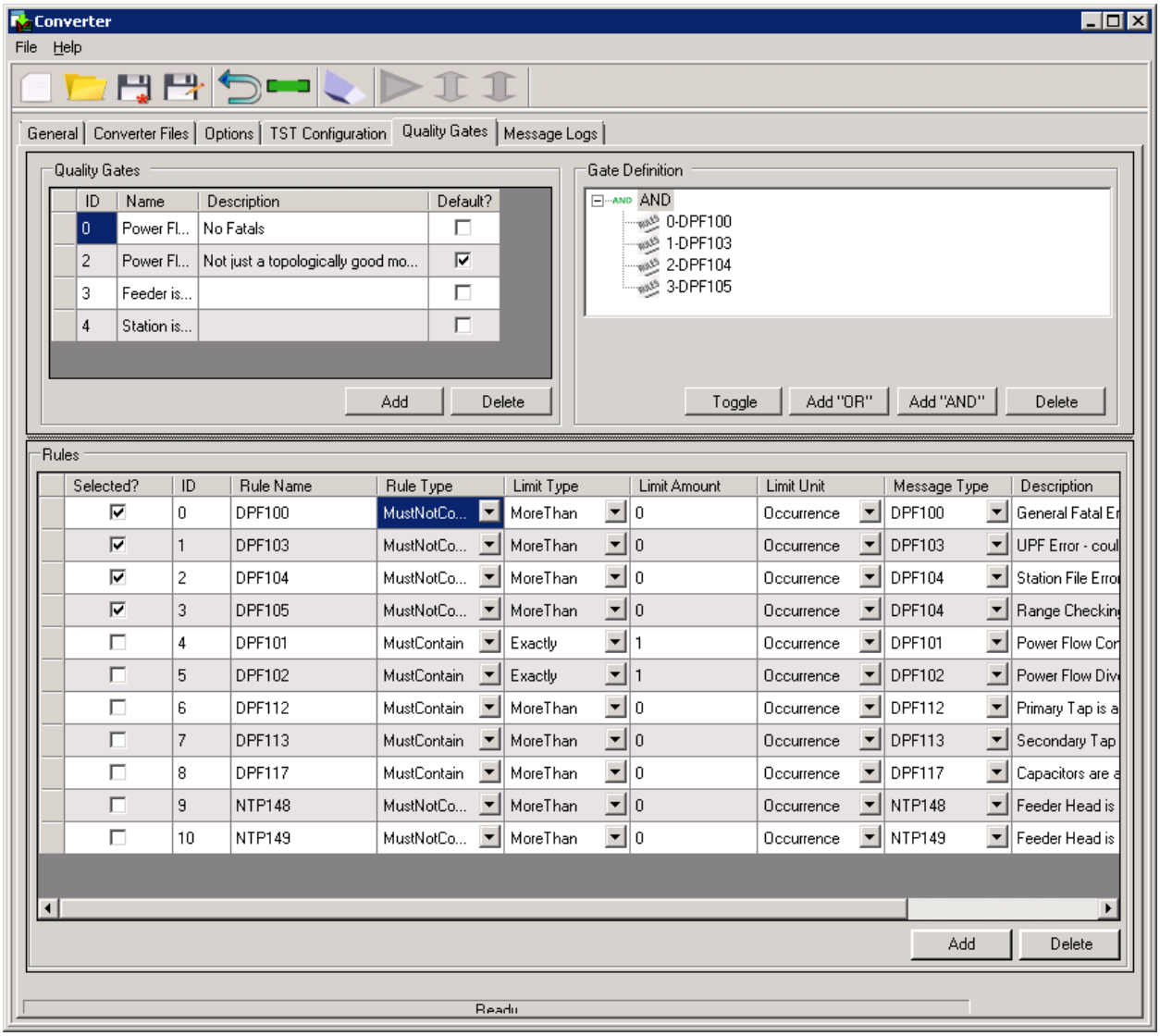


Figure 8. ADMS Converter - Quality Gates Tab

The Quality Gates tab includes the following fields:

- **Quality Gates:**
  - **ID:** Quality gate ID.
  - **Name:** Quality gate name.
  - **Default?:** When the Default gate option is selected, the gate is evaluated after conversion even when no stations are assigned.
  - **Add:** Click to create a new gate.
  - **Delete:** Click to delete a selected gate.

    The Quality Gates list is displayed on the General Tab when clicking the Quality Gates button.
-

Gate Definition: Displays rule assignments along with "And/Or" logic for the selected quality gate.

- **Toggle:** Switches the logic between "And" and "Or" for the selected gate.
- **Add "OR":** Created a new "Or" logic point for the selected gate.
- **Add "AND":** Created a new "And" logic point for the selected gate.

- **Rules:** Displays rules, which can be assigned to quality gates.
  - **Selected?:** Check to add the rule to the selected quality gate. The rule is added to the Gate Definition pane.
  - **ID:** The rule ID number that references this rule.
  - **Rule Name:** The model quality rule name.
  - **Rule Type:** Select one of the following:
    - Must Contain
    - Must Not Contain
  - **Limit Type:** Select one of the following:
    - Less Than
    - No More Than
    - Exactly
    - No Less Than
    - More Than
  - **Limit Amount:** Numerical value related to the Limit Unit.
  - **Limit Unit:** Units in which rule limitations are set (occurrence(s) or percentage). The percentage limit unit is used with certain message types within the Converter message log's Statistics Summary section.
  - **Message Type:** Message types for all messages in Converter_MsgDef_ENU.csv (warning, error, fatal, information) during conversion.
  - **Description:** Rule description.
  - **Add:** Click to add a new rule.
  - **Delete:** Click to delete a selected rule.

2.1.7. Message Logs Tab

The Message Log tab contains log entries generated during the Converter's operation.

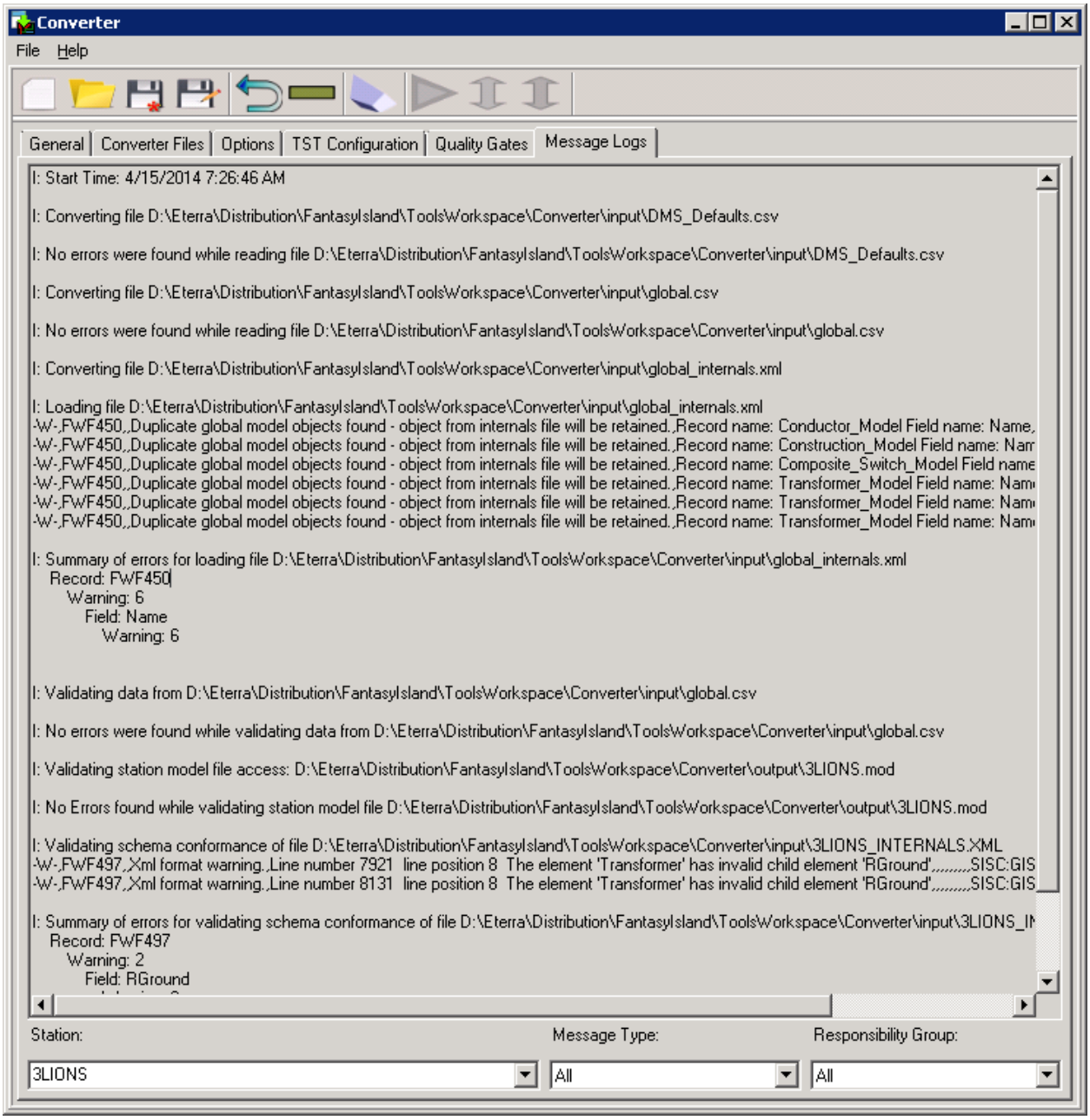


Figure 9. ADMS Converter - Message Log Tab

The log messages can be filtered by station, message type, and responsibility group as necessary using the drop-down menus at the bottom.

2.2. Using the Converter Tool

2.2.1. Reviewing Input Files

Before running the converter, the various model files should all be placed in the same directory. For example, for a station named PLUTO, the following model files need to reside in the same directory:

- PLUTO.xml
- PLUTO_INTERNALS.xml
- PLUTO_PLACEMENTS_*.XML
- global.csv
- DMS_Defaults.csv
- Distribution power flow data files (DAT files, including alg.dat, msgdef.dat, optalg.dat, phid.dat, tftype.dat), and optionally the DNAF_PARAMS.xml file.

  These files are required when executing the converter using the Advance Operations Model with the Power Flow Scope option.

You can set the path to the Input and Output directories via the Input and Output panes on the Converter Files tab.

2.2.2. Converting Station Models

To convert a station model:

1. Select the correct subdirectory using the input and output directory list boxes on the Converter Files Tab.

   To change the folders, click the Browse button  or manually type the new path.

2. To convert a single station, select the station from the Substation list on the General tab.

   Multiple stations can be selected/de-selected by clicking the Select All/Clear All buttons, or by manually selecting another station while holding down the Ctrl key.

Individual files can be viewed and selected for conversion by deselecting the Substation View option.

3. Select the Scope option on the Options tab, and review and set other configurations, as described in Options Tab.
4. Ensure that the files described in Reviewing Input Files are present in the Converter Input folders.
5. Click the Convert button.

The converter starts processing the selected files.

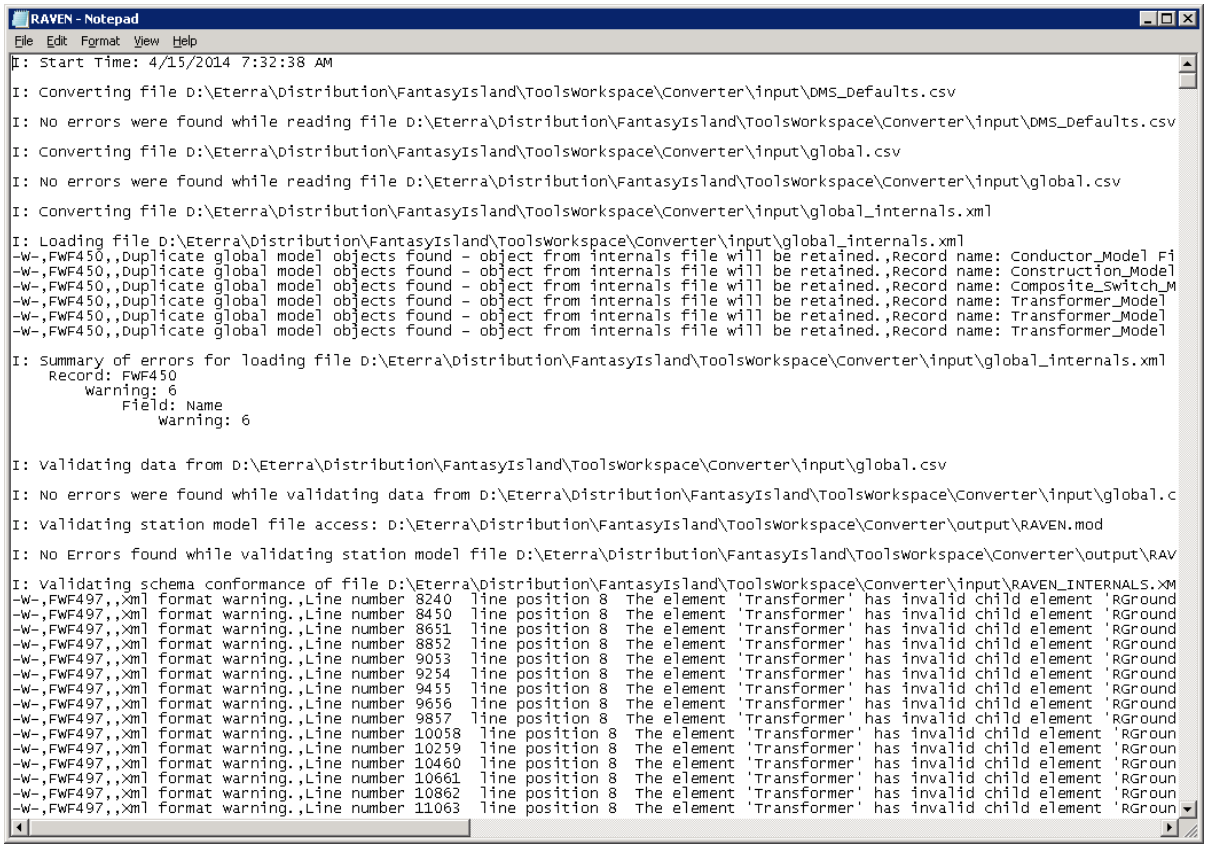
6. After processing is complete, review the error messages on the Message Log tab.

2.2.3. Reviewing Message Logs

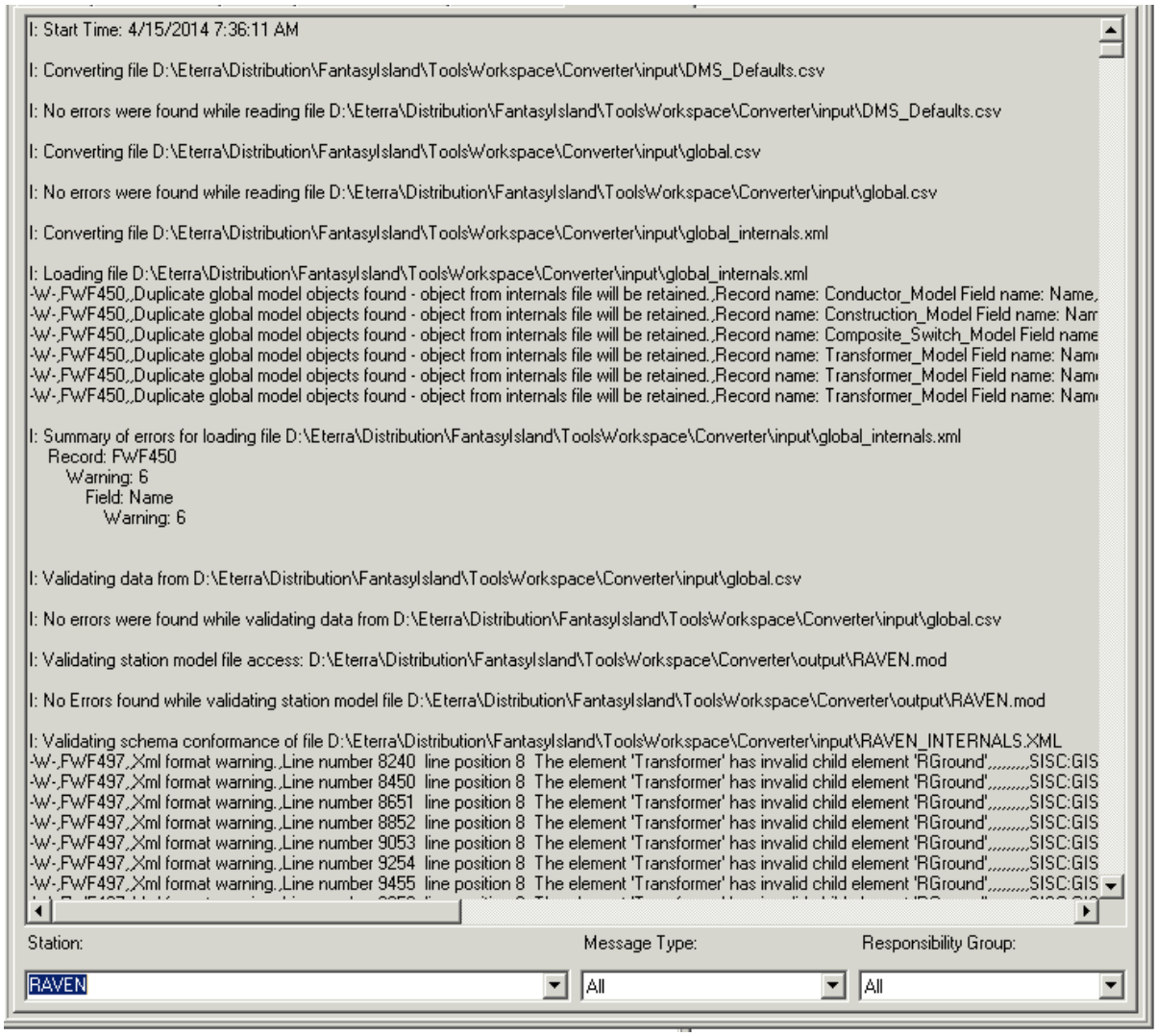
Depending on the errors, there may also be a station model file generated with the name `StationName.mod`. The error log has the name `StationName.log` and is stored in the Converter Output folder.

To review the converter error log, do one of the following:

- Locate the `StationName.log` file in the Converter Output folder, and open the log file with a text editor.



- Open the Message Logs tab and select the desired station and message type options from the drop-down lists at the bottom of the form.



2.2.4. Converting CSV to XML

The Converter provides a way to convert CSV-formatted externals files to XML format.

To convert CSV files to XML files:

1. Open the Converter.
2. From the File menu, select the Convert CSV to XML option.

   The Open dialog box appears with the CSV File Filtering option enabled.
3. Select the CSV file you need to convert into XML format, and click Open.

   The Save As dialog box appears.
4. Enter the desired name for the file to be converted to XML format, and click Save.

2.2.5. Exporting the Error Message Debugging Guide to an XML File

The Converter allows you to export the Error Message Debugging Guide to an external XML file for reference.

To export the Error Message Debugging Guide to an external XML file:

1. Open the Converter.
2. Navigate to File > Export Message Definition to > XML.

   The Save dialog box appears with the XML File Filtering option enabled.
3. Enter a file name and click Save.

The XML formatted file can be opened in Excel as a spreadsheet. Open it with Excel and select "As an XML table" in the dialog box. The content will be shown after clicking OK.

2.2.6. Running the Converter Using the Command Prompt

The Converter can also be run from the command prompt or from other batch operations in scripts.

The syntax and command-line arguments for Converter.exe are as follows:

- **Help (or H, ?):** Shows the help for command line options.
- **BATCH (or B):** Specifies using the command-line mode. This must be specified when running from the Windows command line or from other batch mode operations.
- **CONFIG:** Specifies the configuration file the Converter opens on startup. This must be specified for all operations.
- **DNAFINPUT:** Specifies the directory containing the DNAF-related device parameters files, such as `DNAF_PARAMS.XML`, `alg.dat`, `msgdef.dat`, `optalg.dat`, `phid.dat`, and `tftype.dat`.
- **CONVINPUT:** Specifies the directory containing non-station-based input files like `Converter_MsgDef_ENU.csv` (translation of error codes to text).
- **CUSTFILE:** Specifies the customer file needed to define loads for the network model.
- **INPUT:** Specifies the directory containing station internals (`MySub_internal.xml`) files, station externals (`MySub.xml` or `MySub_sub.csv`) files, `DMS_Defaults.csv` (defines default DMS device parameters), `global.csv` (defines shared device parameters), and `global_internals.xml` (Substation Editor generated file).
- **INDEXFILE:** Optional parameter that specifies the Equipment and Structure index file to be used. If this parameter is not specified, no index file will be used.
- **OUTPUT:** Specifies the directory to store the Converter's output files.
- **CSV2XML:** Specifies the source CSV file and the target XML file used by the Converter tool to

convert the CSV file to an XML file.

- **HPATH:** Specifies full path to the `distribution_modeling_reference_guide.chm` file folder.
- **SCHEMA:** Specifies the schema file.
- **MSGDEF2XML:** Specifies the exported message definition XML file.
- **SAVEAS:** Exports the Converter's configuration to an XML file.
- **TARGET XXXX:** Specifies the station names to be processed by the Converter. The following arguments can be used:
  - -Target "stationName1,stationName2,stationName3"

    Specifies the station names to be processed by the Converter in one string. Station names are separated with commas.

    For example:

    ```
    -Target "3LIONS,BACHELOR,RAVEN,PEBBLE BEACH"
    ```

  - -Target *

    Specifies all stations in the input folder for conversion.

  - -Target txtFilePath

    Specifies all stations in the TXT file for conversion. It provides the path to the TXT file, including the file name.

    Example:

    ```
    -Target D:\eterra\Distribution\FantasyIsland\ToolsWorkspace\Converter\input\StationNames.txt
    ```

    The TXT file lists all of the stations to be processed by the Converter, with one station per line.

The example below demonstrates the command-line syntax to convert all stations stored in the Input folder.

```
Converter.exe -CONFIG d:\Eterra\Distribution\FantasyIsland\ToolsWorkspace\Converter\Converter.xml
-CONVINPUT D:\Eterra\Distribution\FantasyIsland\ToolsWorkspace\Converter -DNAFINPUT
D:\Eterra\Distribution\FantasyIsland\ToolsWorkspace\DNAF_Parameters\ -CUSTFILE
D:\Eterra\Distribution\FantasyIsland\models\customer\FantasyIslandCustomers.bpcl -Input
D:\Eterra\Distribution\FantasyIsland\ToolsWorkspace\Converter\input -Output
D:\Eterra\Distribution\FantasyIsland\ToolsWorkspace\Converter\output\ -Batch -Target *
```

2.2.7. Using Quality Gates

The Converter provides an optional Quality Gate feature to evaluate Converter messages as a set and to use the results to gate the workflow for pushing a model update into production.

For example, a utility may want to ensure that several of their stations are always capable to run with Heuristic-Mode LVM. A Quality Gate can be defined and associated with those stations to ensure this, and therefore block a model update that degrades the quality of that model based on warnings or errors.

The workflow decision depends upon:

- Evaluation of the messages generated by Converter for the substation model.
- Application of user-defined criteria based on the substation and related warning messages.

The Converter can be configured to issue a Fatal message when a Quality Gate evaluation fails. In this case, the Model Manager can use this result and prevent a model update from going into production.

A Quality Gate evaluation for a station effectively raises the severity of a set of messages. For example, messages that normally have a Warning severity can be evaluated together and result in the Converter issuing a Fatal message.

2.2.7.1. Enabling the Quality Gate Feature

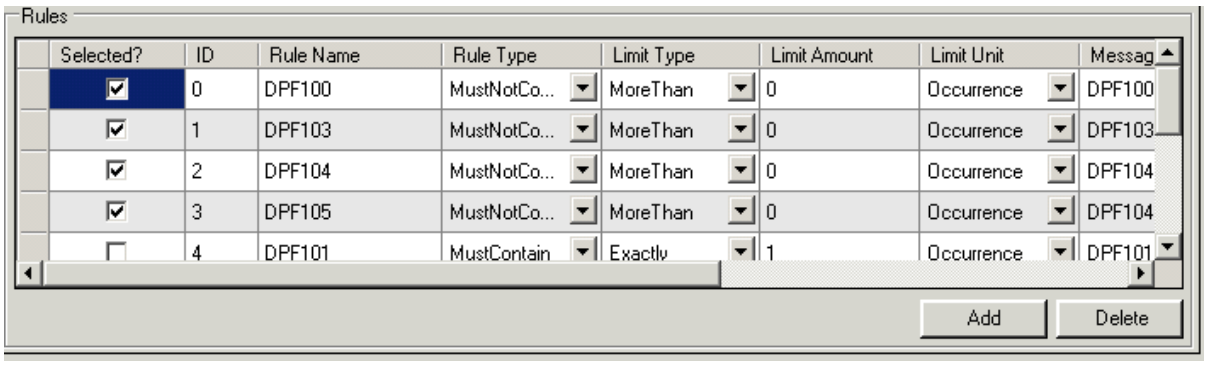
To enable the Quality Gate feature, select the Quality Gate Evaluation option on the Options tab of the Converter tool.

Note that the Converter will create and maintain a StationQualityGateAssignment XML file in the Converter's folder. That file contains the Station List and the Station/Gates Assignment.

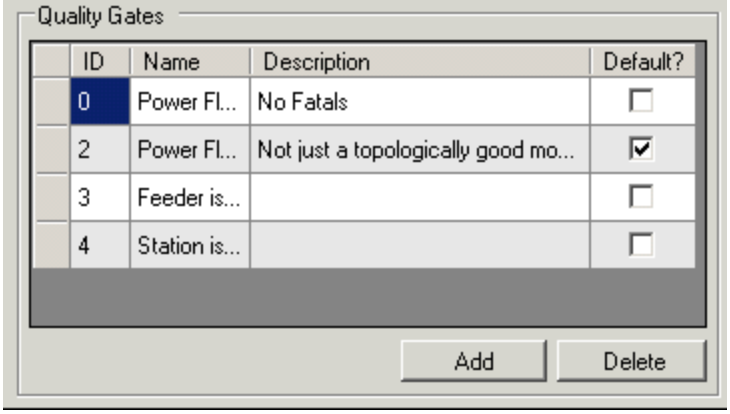
2.2.7.2. Creating a Quality Gate

To create a quality gate:

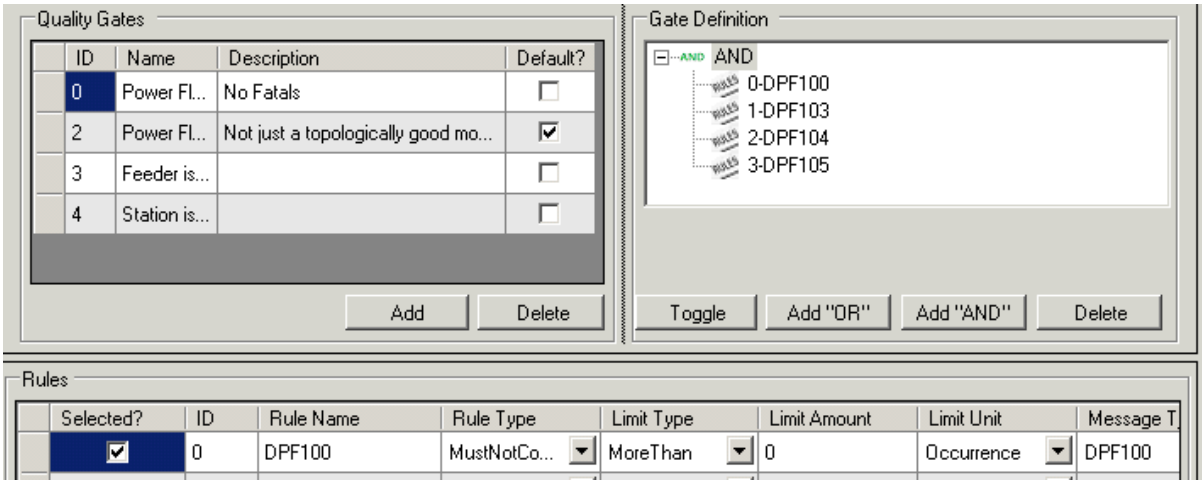
1. Start the Converter and navigate to the Quality Gates tab.
2. On the Rules pane, click Add.



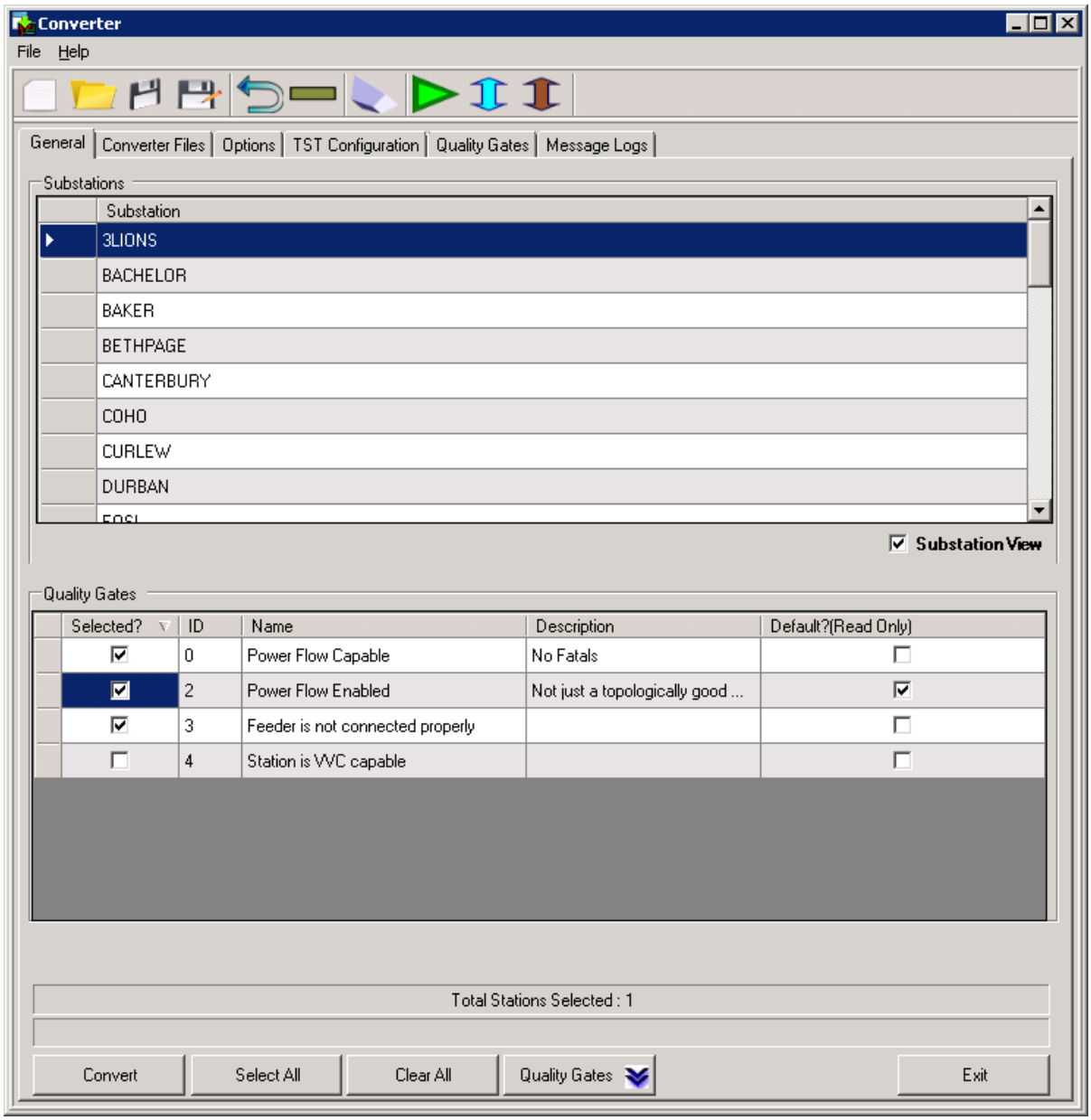
3. Enter the parameters in the Rules grid, as described in Quality Gates Tab.
4. On the Quality Gates pane, click Add to create a new gate.



5. In the Quality Gates grid, enter data as necessary.
6. Select a gate to view rule assignments along with "And/Or" logic.



7. In the Rules group, select check boxes in the "Selected?" column to enable rules as needed.
8. On the Gate Definition pane, select a Logic Point and click Toggle to switch the logic between "And" and "Or" for the selected gate.
9. Click "Add OR" or "Add AND" to create a new logic point for the selected gate.
10. Open the General tab to assign new gates to substations.
11. Select a substation on the Substation panel.
12. On the Quality Gates panel, select gates to assign to the substation.



13. All selected gates are now associated with substations.

After conversion, information messages for Pass results and error messages for Failed results will refer to existing Converter Message Log Severities (see Reviewing Converter Messages).

2.2.7.3. Sample Gates and Rules in the Fantasy Island Expansion Pack

The `Converter.xml` and the `StationQualityGateAssignment` XML files in the Fantasy Island expansion pack contain rules and station assignments that can be used as examples for project-specific configurations.

2.3. Reviewing Converter Messages

The converter error messages are stored in the `Converter_MsgDef_ENU.csv` file. This file contains the following fields:

- Application message number (identifier)
- Responsibility Group: Party that is responsible for correcting the error. This is an optional field. The party identifier is used by the model management environment.

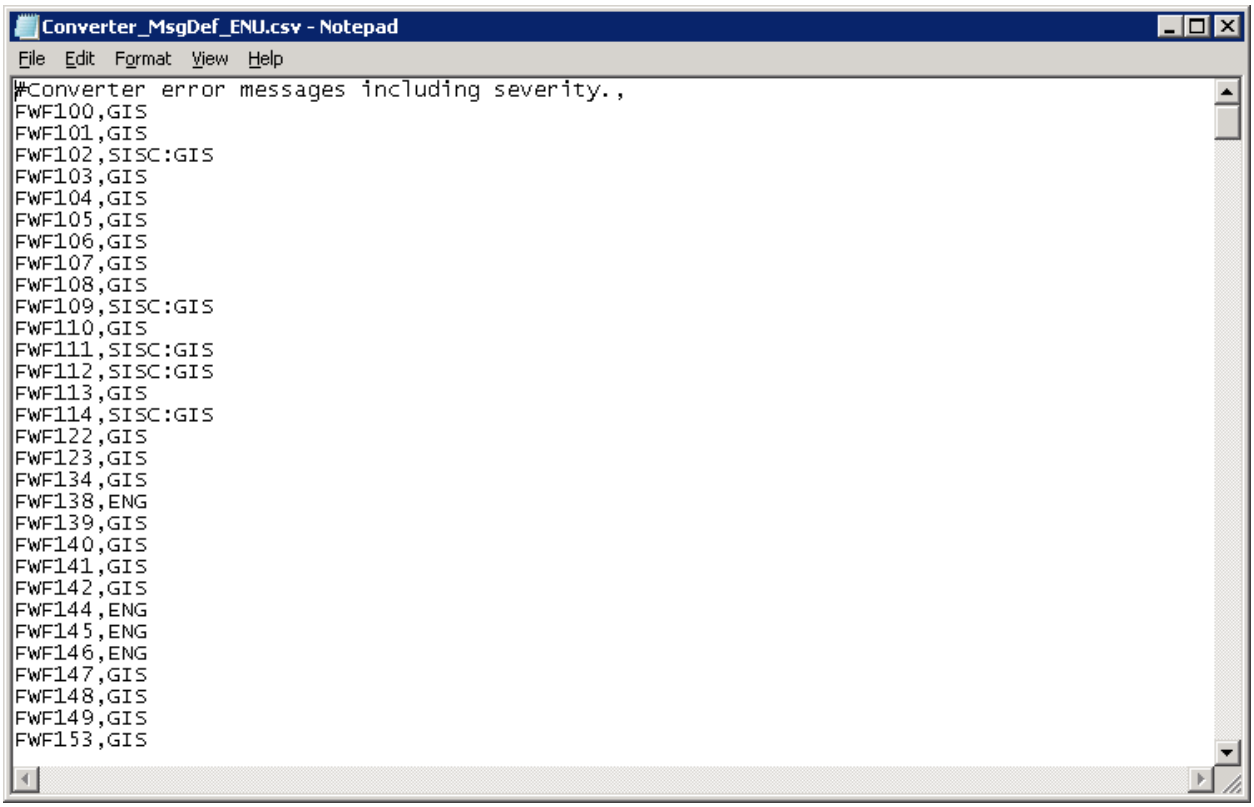


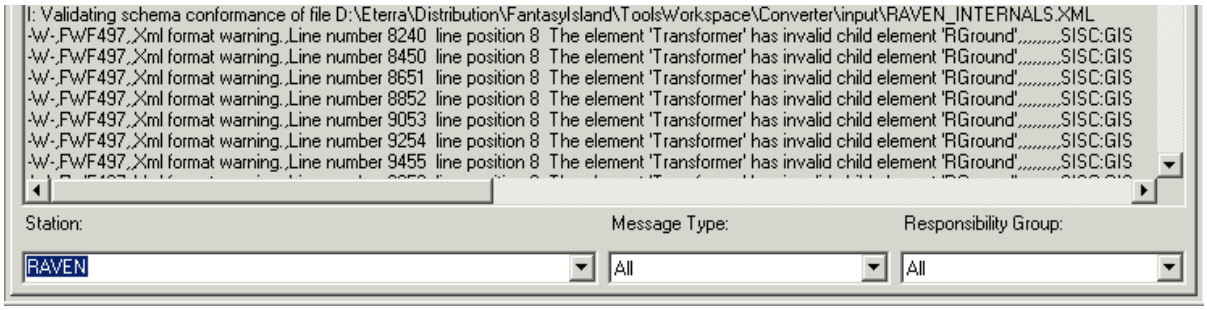
Figure 10. Example of the Converter_MsgDef_ENU.csv File

The conversion process is not started if the Converter_MsgDef_ENU.csv file is absent in the specified Converter input folder. If this message file is missing some messages, these messages are logged as Fatals, with a main text string that says, "Message not found – check the message file".

The default location of the Converter_MsgDef_ENU.csv file is:
`D:\Eterra\Distribution\FantasyIsland\ToolsWorkspace\Converter\`

All converter messages have the same comma-separated format:

- Severity, Application Message Number, File Line Number, Line Position, String 1, String 2, String 3, Value, Phase/Unit, High Value, Low Value, Substation, Feeder ID, Device ID, and Responsibility Group.



This format is described in the following sections.

2.3.1. Severity

The message severity has the following values:

- Informative (I)
- Warning (W)
- Error (E)
- Fatal (F)

The following design rules have been used in assigning the message severity to the converter output messages:

- Duplicate IDs and indexes are only warnings (duplicate station records are not loaded into the model).
- All other file well-formed data issues result in fatal errors (incorrect headers, incorrect data types, incorrect model or object associations, missing required files, incorrect data ranges such as for enumerations).
- All range checking errors are fatal.
- Voltage level assignment problems are warnings.
- Feeder head de-energization results in a fatal message. All other topology-related messages, including de-energization status and looped status, result in warnings.
- A problem in solving the topology processor function results in a fatal message. This is typically due to invalid topology.
- If the power flow does not have the necessary input data, or if the power flow cannot be solved, this results in a fatal message.
- A power flow limit violation (flow or voltage) results in an error message.

Whenever possible, the Converter always outputs a model file. For example, if there are errors resulting from the power flow (flow limit or voltage violations) but the power flow converged, a model file is generated.

2.3.2. Application Message Number

Each error is identified by a unique string consisting of two or three letters followed by an error number. For example, a message identifier of FWF100 indicates a file well-formed type error.

There are six error types:

- File Well-Formed (FWF)
- Range Checking (RC)
- Voltage-Level Assignment (VLA)
- Line Model Creation (LMC)
- Network Topology Processor (NTP)
- Distribution Power Flow (DPF)

The Converter first performs the file well-formed checks. These checks test for valid data header records (such as making sure that a field is recognized for a specific record type), that valid field types have been given for each field (string, Boolean, integer, or floating point), and that special fields such as repeating fields have the correct number of entries.

The second check that is performed is range checking. These checks test to see if numeric values are inside of their acceptable range (the valid range is listed in the Low Value and High Value columns of the *ADMS Modeling Reference Tables* spreadsheet). Most of the fields that are subject to this test are power flow application–specific attributes such as conductor model resistance.

The third piece of processing that is done in the Converter assigning the nominal voltage levels to all nodes in the station model. The nominal voltage is based on the voltage given in the transformer and feeder records. For example, if there is a 22.9 kV feeder head that has a downstream transformer with a rated primary voltage of 13.2 kV, a voltage-level assignment problem is reported for this transformer primary node.

The fourth piece of processing that is done by the Converter is line model creation. The line models are only built if the user sets the Converter scope to at least advanced operations model and if there were no fatal errors in the previous processing. The line models are used by the power flow applications to store the series impedance matrix and shunt admittance matrix values.

The next application that is solved in the Converter is the NTP. The NTP assigns the nominal feeder to every component and determines de-energization, and loops in the topology components. It calculates energized distribution transformer phases and the energized customer number of the station and of each feeder. The calculated values are written to the message log.

The last piece of processing that is done by the converter is the power flow solution. If the converter scope is set to Advanced Operations Model with Power Flow Solution and there were no previous fatal errors, the power flow is solved. If the power flow converges, all voltage and flow violations are written to the message log.

2.3.3. File Line Number and Line Position

Where possible, the specific input file line number and line position are given in the error message. This guides the user to the location of the data error.

2.3.4. String 1, String 2, String 3

These strings provide additional error information such as the record, field, or ID of a specific device or line with a problem.

The strings include the Internal flag if the device is from internal files (either from Substation Editor or generated by other tools, such as the EMP2DNOM tool), or the External flag if the device is from external files (GIS, etc.).

2.3.5. Value

For certain messages, the incorrect value is provided in this field.

2.3.6. Phase/Unit

For certain messages, this field can display either the failed phase of the object or the unit of the value depending on the message. For example, the field could have a kVA/Amp value for some instances, and other values such as A, ABC, etc. in other instances. This is commonly done for de-energized (or looped) topology devices such as lines, switches, and transformers.

2.3.7. High Value, Low Value

For certain messages, the numeric value range is provided in these strings. If no applicable range exists, these values are 0.0 for both the high value and the low value.

2.3.8. Substation

For certain messages, the substation name is provided as part of the message.

2.3.9. Feeder ID

For certain messages, the GIS feeder ID is provided as part of the message.

2.3.10. Device ID

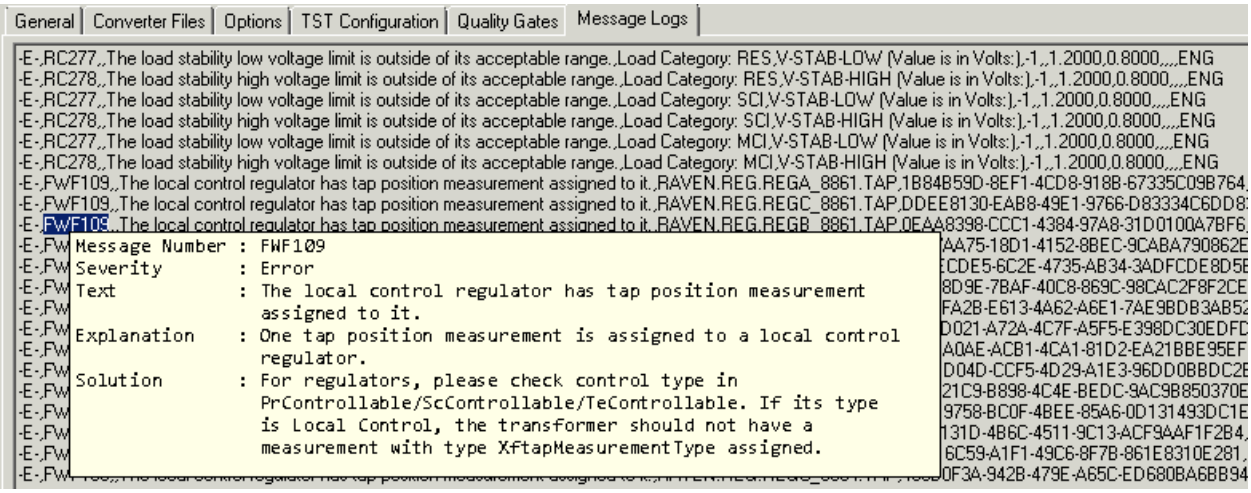
For certain messages, the device ID is provided as part of the message.

2.3.11. Responsibility Group

Responsibility groups in Converter_MsgDef_ENU.csv are determined by customers. All non-informative messages (-I-) must be assigned to one or more responsibility groups. The user can parse the error codes based on this field.

2.4. On-Line Explanation and Solution to Error Messages

On the Converter's Message Logs tab, when you select an error message number (for example, RC106), a tool tip containing the error's severity, text, description, and solution will be shown for reference. This can help you find the cause of the error being reported and how to solve the problem.



2.5. Troubleshooting

2.5.1. Model Troubleshooting

The Converter messages provide users with detailed information about where data problems exist within the model file. Where possible, the line number within the model file is referenced to provide an easy way to locate the source of the error.

When reviewing converter messages, the file well-formed errors are shown first, followed by the data range validation errors, voltage-level assessment errors, line model creation errors, topology-related errors, and power flow errors. Topology errors may be due to the file well-formed errors, so it is wise to handle the file well-formed errors first.

If topology errors exist after all the file well-formed errors have been cleaned up, there may be errors in the connectivity given in the station model file. By loading the model file into the study

model file directory, it is possible to graphically view the topology problems and make corrections back in the source station model file.

A consolidated log for the Fatal, Warning, Error, and Info messages can be obtained by using the Model Quality Stat Logger tool. In addition, the DNOM statistics can be obtained by using the Model Quality Stat Logger tool (see Using the Model Quality Stat Logger Tool).

2.5.2. Problems and Solutions

2.5.2.1. Boolean Values in XML Files

Symptom: In XML modeling files (for example, station input files such as <Station>.xml and Substation Editor files such as `<Station>_subeditor.xml`), values of "1" and "0" are resolved as "True" and "False" in Boolean fields.

A Boolean is a simple type, which serializes using the XML standard, and assumes that "1" and "0" are correct Boolean values for "True" and "False". No exception is shown in this case.

Solution: It is recommended to use "True" or "False" values for Boolean fields, and avoid using "1" and "0".

3. Using the Model Quality Stat Logger Tool

The Model Quality Stat Logger tool reads files in the Converter's output folder and extracts statistics about the quality of the models and Converter error messages into consolidated log files.

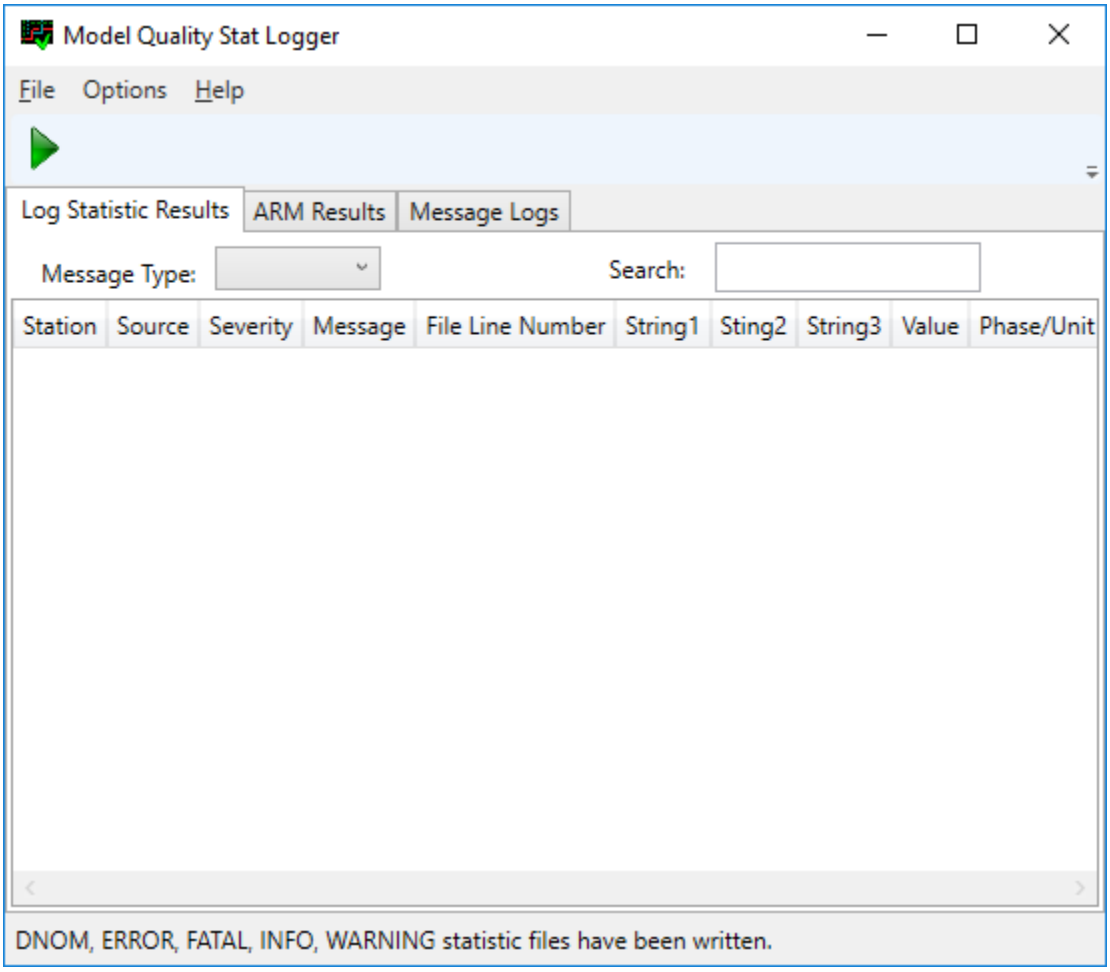


Figure 11. Model Quality Stat Logger Tool

3.1. Model Quality Stat Logger Displays and Controls

3.1.1. Menu and Toolbar Options

The Model Quality Stat Logger includes the following menu and toolbar options:

- **File > Exit:** Closes the application.
- **Options > Configuration:** Opens the Model Quality Stat Logger Configuration Editor tool.
- **Consolidate Log Messages:** Writes the results into consolidated log files according to the parameters specified in the Model Manager Stat Logger Configuration Editor.

3.1.2. Model Quality Stat Logger - Log Statistics Results Tab

The Log Statistic Results tab of the Model Quality Stat Logger shows a tabular summary for consolidated log statistics.

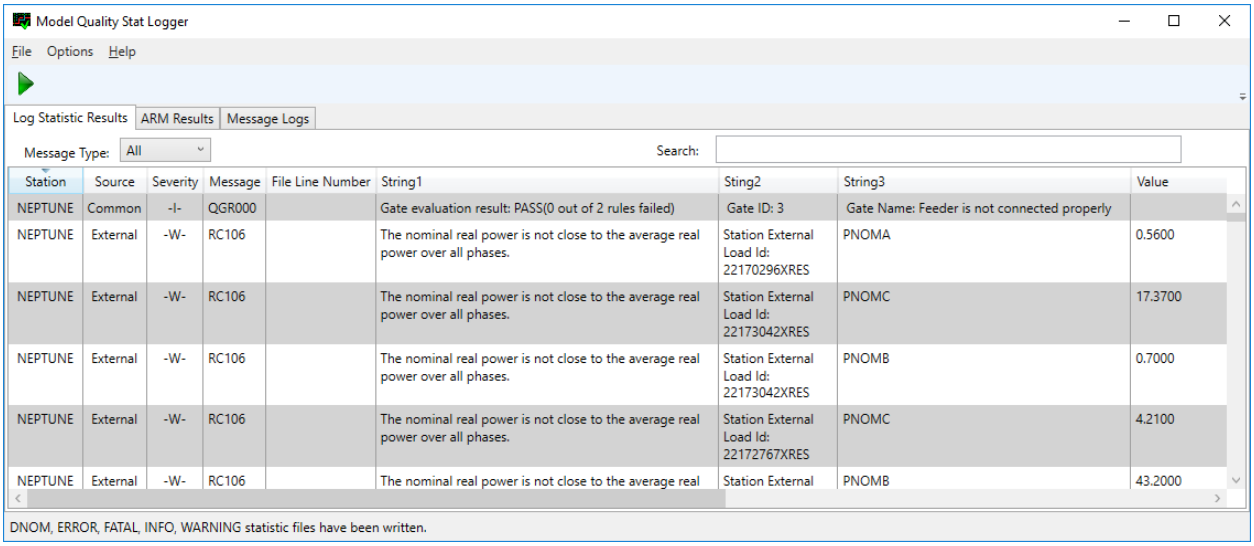


Figure 12. Model Quality Stat Logger - Log Statistic Results Tab

3.1.3. Model Quality Stat Logger - ARM Results Tab

The ARM Results tab of the Model Quality Stat Logger shows the model application readiness summary.

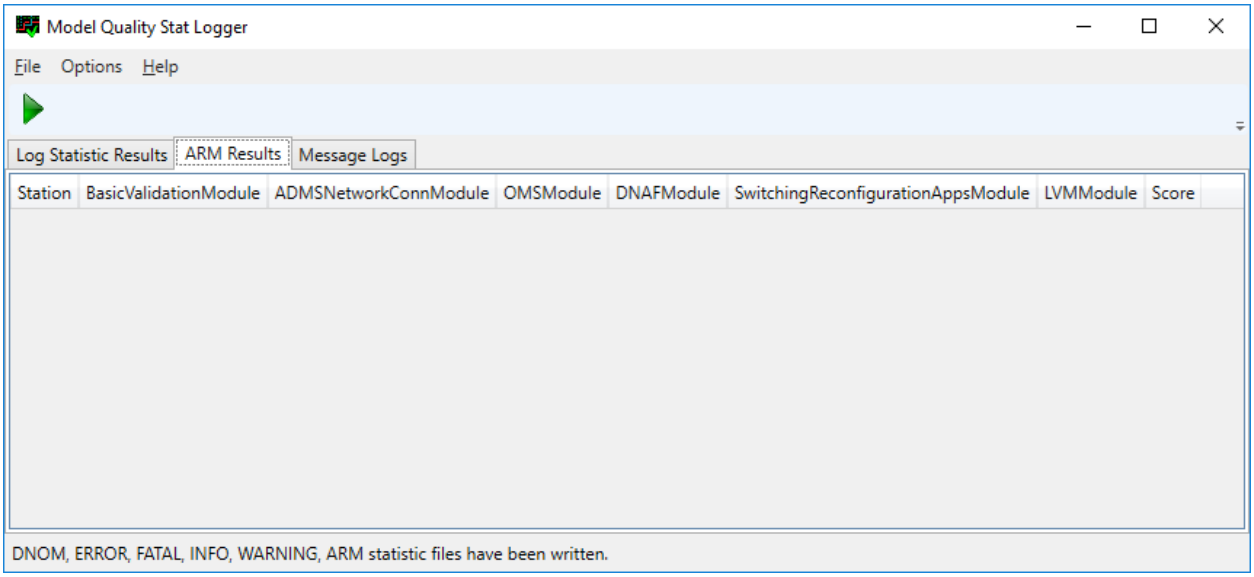


Figure 13. Model Quality Stat Logger - ARM Results Tab

3.1.4. Model Quality Stat Logger - Message Logs Tab

The Message Logs tab of the Model Quality Stat Logger shows the log of the Model Quality Stat Logger operations.

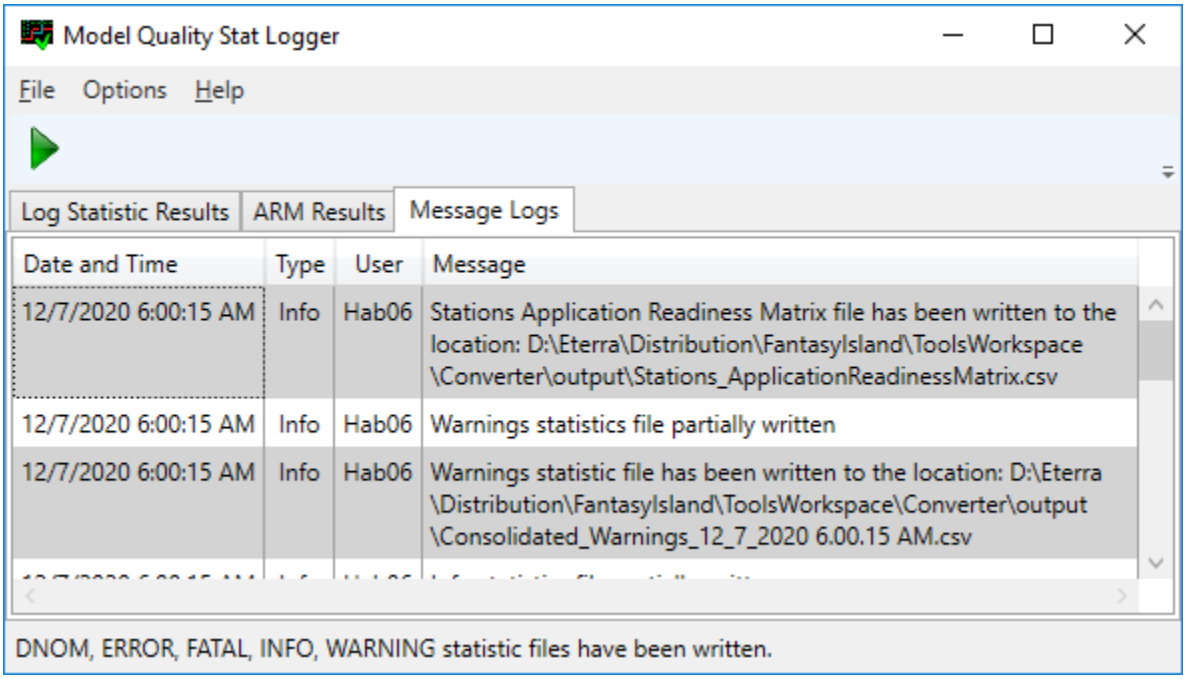


Figure 14. Model Quality Stat Logger - Message Logs Tab

3.1.5. Model Quality Stat Logger Configuration Editor - Folders Tab

Use the Folders tab of the Model Quality Stat Logger Configuration Editor to specify input and output paths for the Model Quality Stat Logger operations.

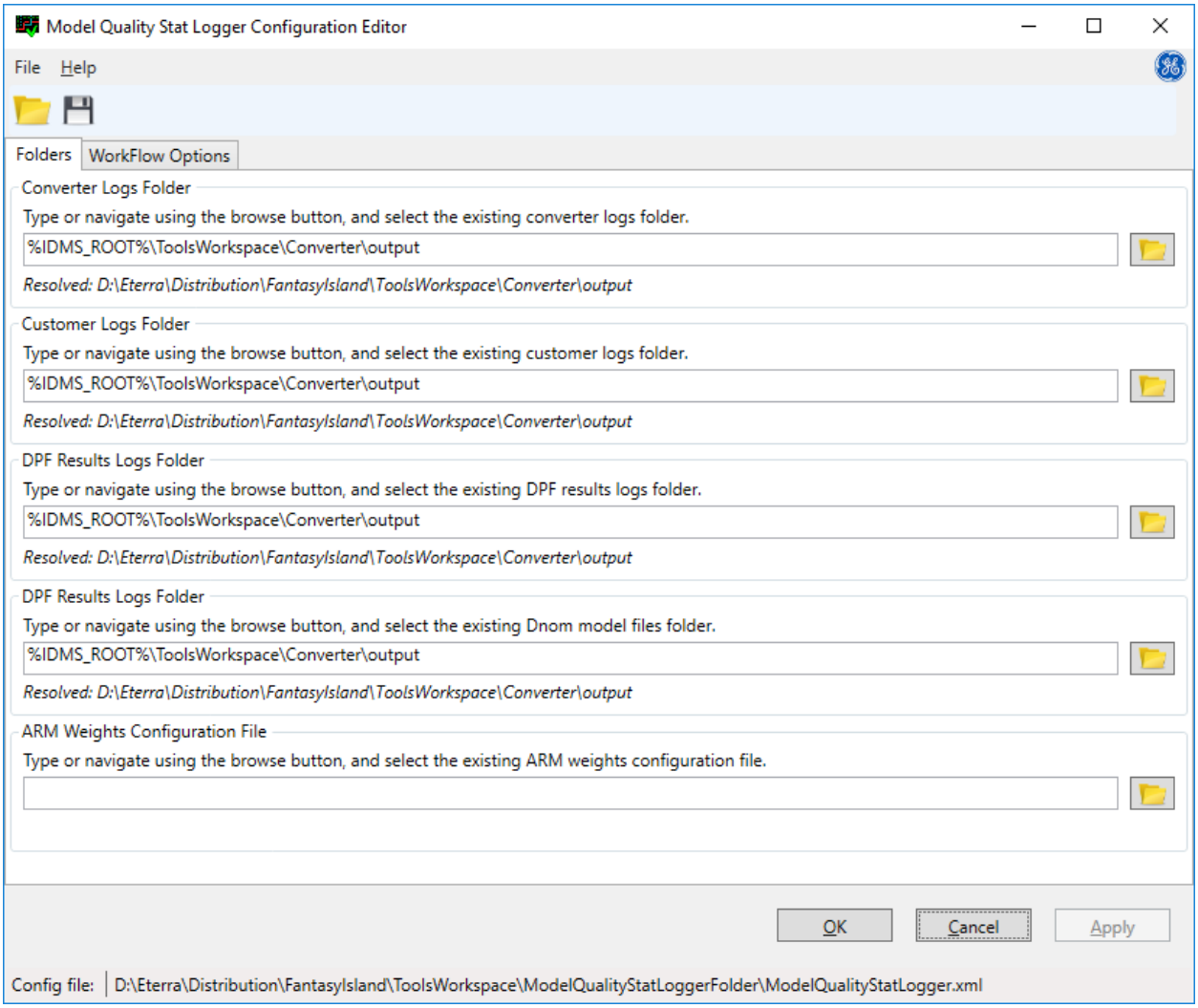


Figure 15. Model Quality Stat Logger Configuration Editor - Folders Logs Tab

The Folders tab includes the following options:

- **Converter Logs Folder:** The path to the Converter output folder. For more information about the Converter's folders and files, see Converter Files Tab.
- **Customer Logs Folder:** The path to a folder with customer logs created by the Converter.
- **DPF Results Logs Folder:** The path to a folder with Distribution Power Flow (DPF) logs created by the Converter.
- **ARM Weights Configuration Files:** The path to a file with Application Readiness Matrix (ARM) configuration file.

3.1.6. Model Quality Stat Logger Configuration Editor - Folders Tab

Use the Workflow tab of the Model Quality Stat Logger Configuration Editor to specify the Model Quality Stat Logger workflow settings.

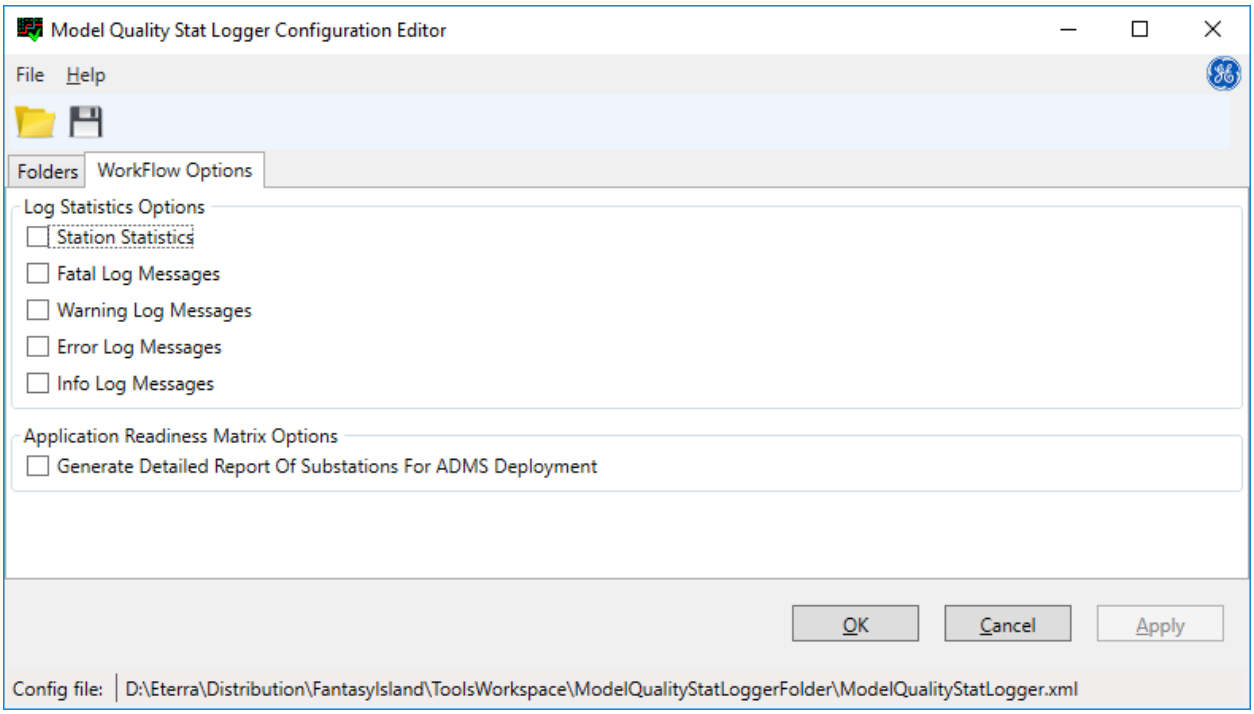


Figure 16. Model Quality Stat Logger Configuration Editor - Workflow Options Tab

The Workflow Options tab includes the following options:

- **DNOM Log Messages:** Specifies whether the Model Quality Stat Logger creates a file named `Consolidated_Stats_<datetime>.log`. The file's data includes:
  - The total number of station log files that are read by the tool
  - The relevant device counts
  - The energization statistics across all stations
  - The DNAF statistics (transformers with voltage violations, loads with voltage violations, loop counts, etc.) across all stations
  - Count and list of all stations with the power flow status converged, diverged, could not be solved, or iteration count less than 10.
- **Fatal Log Messages:** Specifies whether the Model Quality Stat Logger extracts all fatal messages from the log files into the `Consolidated_Fatals_<datetime>.log` file.
- **Warning Log Messages:** Specifies whether the Model Quality Stat Logger extracts all warning messages from the log files into the `Consolidated_Warnings_<datetime>.log` file.
- **Error Log Messages:** Specifies whether the Model Quality Stat Logger extracts all error

messages from the log files into the `Consolidated_Error_<datetime>.log` file.

- **Info Log Messages:** Specifies whether the Model Quality Stat Logger extracts all information messages from the log files into the `Consolidated_Infos_<datetime>.log` file.
- **Generate Detailed Report of Substations for ADMS Deployment:** Specifies whether the Model Quality Stat Logger generates the application readiness matrix (ARM) file with detailed report of model readiness for using in ADMS.

3.2. Using the Model Quality Stat Logger Tool

3.2.1. Starting the Application

To start the Model Quality Stat Logger application:

- From the Microsoft Windows Start menu, navigate to **All Programs > Eterra > Distribution > Tools > Modeling Tools > Model Quality Stat Logger**.

3.2.2. Writing Consolidated Log Files

To write consolidated log files using the Model Manager Stat Logger tool:

1. Run the Model Manager Stat Logger tool.
2. Click the Consolidate Log Messages button on the toolbar.

   The results are written into consolidated log files according to the parameters specified in the Model Manager Stat Logger Configuration Editor.

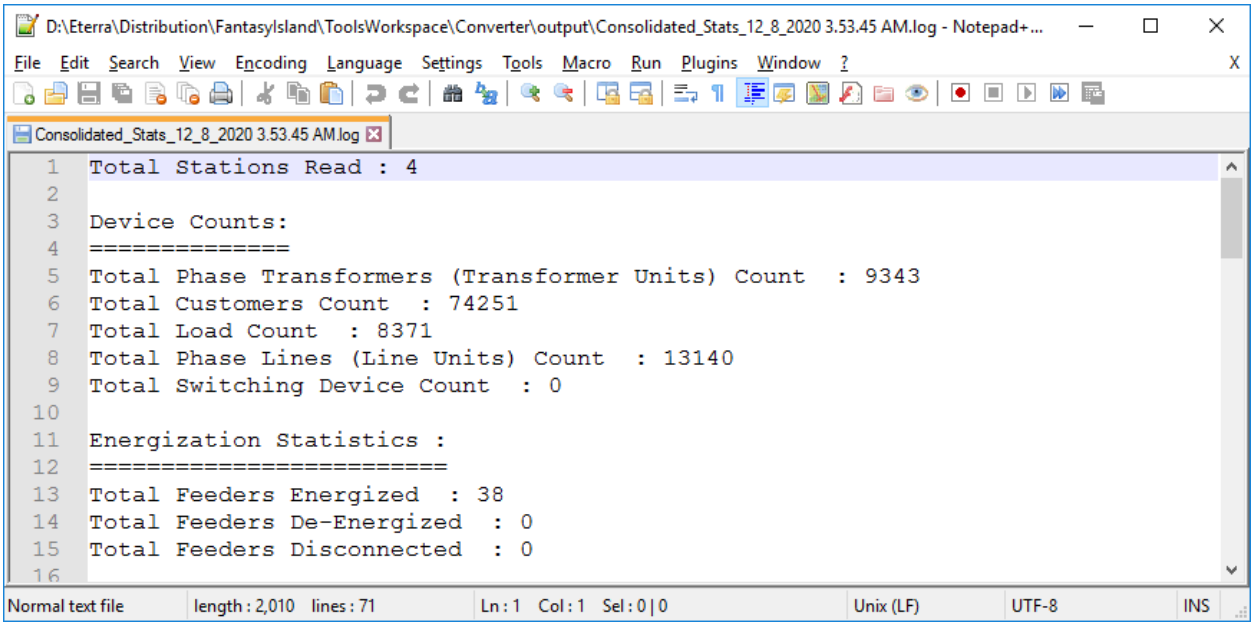


Figure 17. Model Quality Stat Logger - Consolidated Log File

3.2.3. Running the Application Using the Command Prompt

The following are command-line arguments that can be used when running the Model Quality Stat Logger from the command prompt.

- **-PATH:** This argument should be followed by the converter's output path.
- **-DNOM:** When this argument is specified in the command argument, Model Quality Stat Logger creates the `Consolidated_Stats_<datetime>.csv` file.
- **-F:** When this argument is specified in the command line argument, Model Quality Stat Logger extracts all fatal messages from the log files into the `Consolidated_Fatals_<datetime>.csv` file.
- **-W:** When this argument is specified in the command line argument, Model Quality Stat Logger extracts all warning messages from the log files into the `Consolidated_Warnings_<datetime>.csv` file.
- **-E:** When this argument is specified in the command line argument, Model Quality Stat Logger extracts all error messages from the log files into the `Consolidated_Errors_<datetime>.csv` file.
- **-I:** When this argument is specified in the command line argument, Model Quality Stat Logger extracts all error messages from the log files into the `Consolidated_Info_<datetime>.csv` file.

The example command-line string below creates all of the files listed above:

```
D:\Eterra\Distribution\Tools\Bin\ModelQualityStatLogger.exe -PATH
D:\Eterra\Distribution\FantasyIsland\ToolsWorkspace\Converter\output -DNOM -F -W -E -I
```

4. Using the DNOM Unique ID Checker Tool

The DNOM Unique ID Checker tool evaluates a set of model files for unique object IDs to ensure that the DMS system is functioning optimally. When objects IDs are not unique, issues might occur with the dynamics application process.

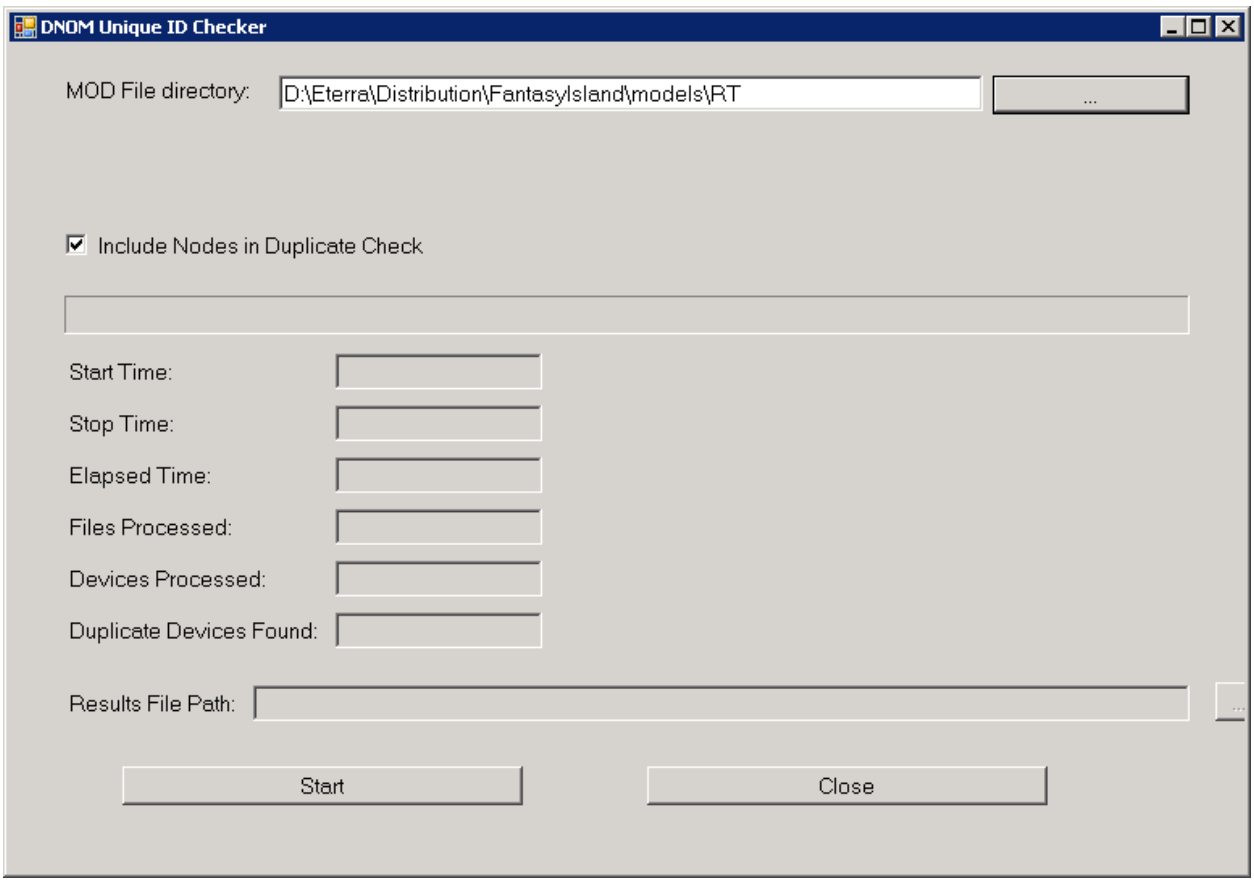


Figure 18. DNOM Unique ID Checker Tool

4.1. Starting the DNOM Unique ID Checker

To start the tool, navigate to **Start > All Programs > Eterra > Distribution > Tools > Modeling Tools > DNOM Unique ID Checker**. To run the program, enter the path to the set of MOD files to be analyzed and click the Start button.

The tool opens and parses each MOD file in a folder, extracts IDs of all the objects in the file, and creates a sorted list of IDs. The tool then compares IDs and reports duplicates in an output file.

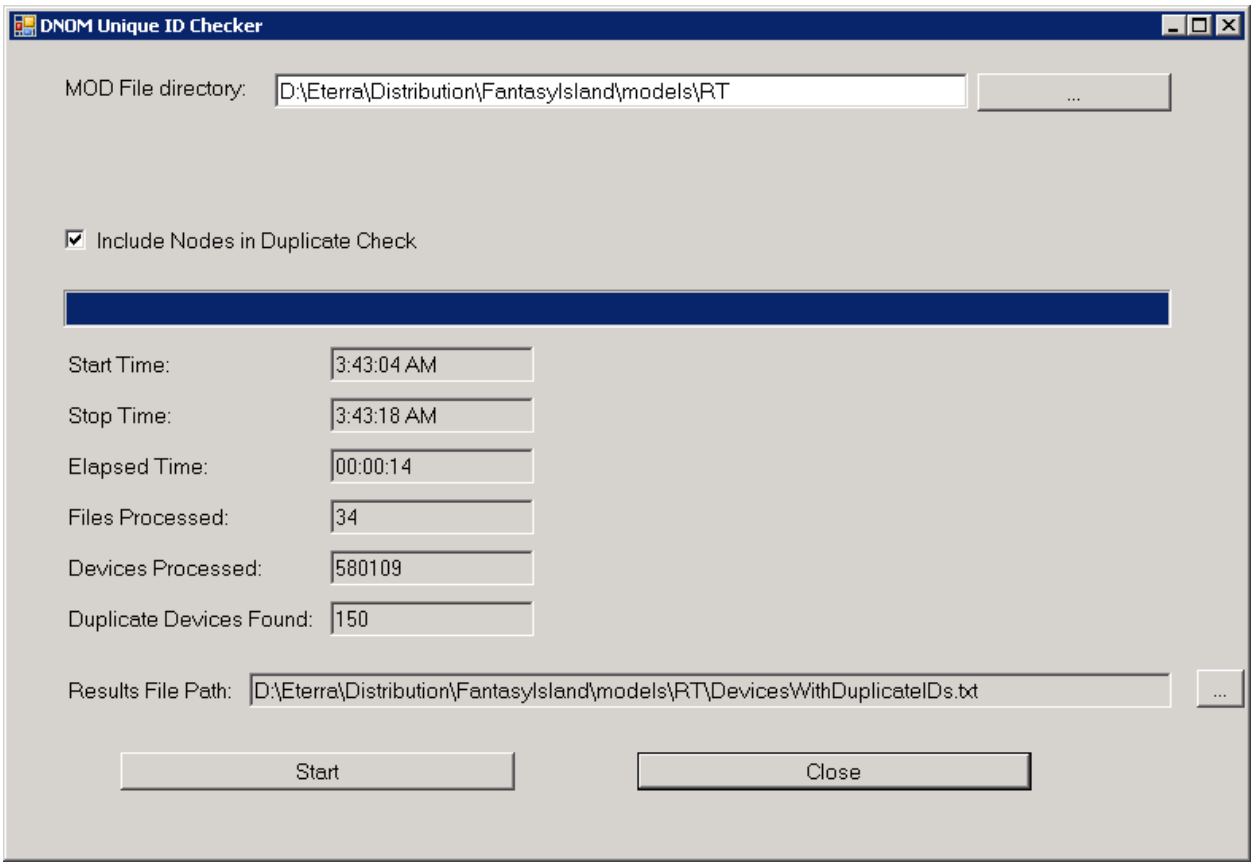


Figure 19. DNOM Unique ID Checker Results

Open the results file to see the list of duplicate IDs. To eliminate duplicated IDs, edit model inputs.

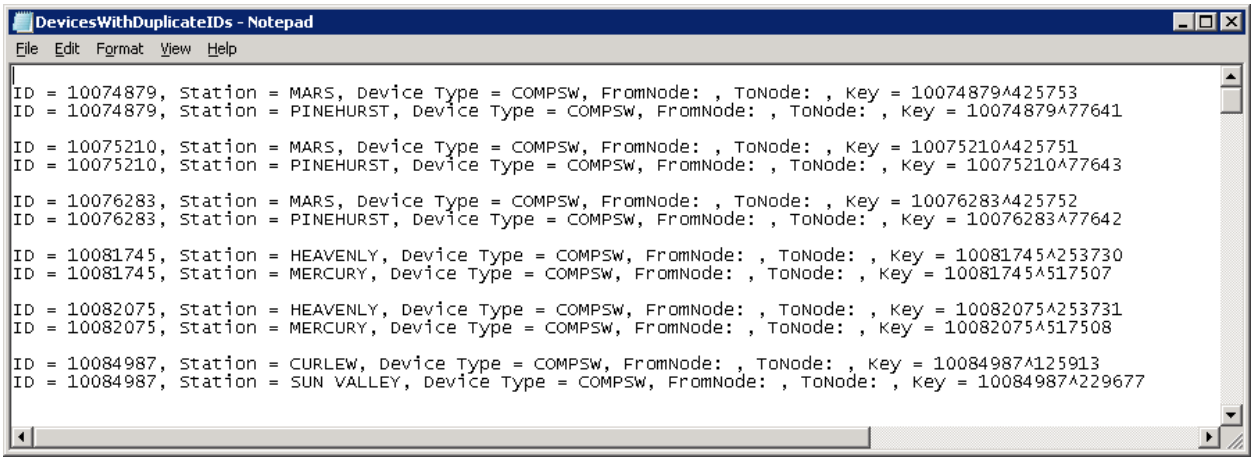


Figure 20. DNOM Unique ID Checker Output

The DNOM Unique ID Checker output contains ID, station, device type, and the From and To nodes of the located duplicates.

5. Using Special Characters

This chapter describes the special characters used by SCADA and the Converter.

5.1. Characters Restricted by the Converter

Special characters restricted by the Converter are defined in the converter configuration file, `converter.xml`.

```
<RestrictedCharacters>[*#?|^=",;`&lt;&gt;[\]\{}\\:()/&]</RestrictedCharacters>
```

Do not remove restricted characters from Converter.xml.

This table lists the restricted characters and describes why they are restricted.

Character Name	Symbol	Restriction Reason	Planned Support?
Asterisk	*	ID is not recognized	Yes. Converter and downstream applications should be fixed to support this
Hash	#	Switch order ignores this character	Yes
Question Mark	?	Toolbar does not work	Yes
Pipe	|	Switch Order converts this to "%7C"	Yes
Caret	^	Switch order converts this to "%5E"	Yes
Equal to	=	Search does not work	No (restricted by SCADA as well)
Double Quotes	"	Locate does not work	No (restricted by SCADA as well)
Comma	,	Breaks CSV files	No
Semi Colon	;	Locate does not work	Yes
Back Tick (below Tilde)	`	Switch order converts this to "%60"	No (restricted by SCADA as well)
Less Than	< (&lt)	Causes XML serialization error	No
Greater Than	> (&gt)	Causes XML serialization error	No
Forward Slash	/	Toolbar does not work	Yes
Backward Slash	\	Switch Order converts this to "%5C"	No (restricted by SCADA as well)
Square Braces	[]	Switch Order converts this to "%5B%5D"	Yes
Curly Braces	{}	Switch Order converts this to "%7B%7D"	Yes
Ampersand	&	Causes XML serialization error	No
Colon	:	Restricted by SCADA\Tagging	No
Open parenthesis	(	Restricted by SCADA\Tagging	No
Close parenthesis	)	Restricted by SCADA\Tagging	No
Single Quote	‘	Restricted by SCADA\Tagging	No

5.2. Characters Restricted by SCADA

The following table lists the special characters that are restricted by SCADA.

Character Name	Character Symbol
Colon	:
Semi Colon	;
Comma	,
Back tick (below tilde)	`
Open parenthesis	(
Close parenthesis	)
Equal to	=
Backward slash	\
Single Quote	‘
Double Quotes	"

5.3. Characters Allowed by the Converter

The following table lists special characters that are allowed by the Converter.

Character Name	Character Symbol
Period (Dot)	.
Dash	-
Exclamation	!
At sign	@
Dollar	$
Tilde	~
Percentage	%
Plus	+
Underscore	_

6. Setting the PermitAreaId

This chapter describes the Permission Area ID settings for DMS devices with DSCADA (SCADA control device) permissions.

The following steps can be used as a reference for station-level permission settings.

1. Locate the PERMIT server parameters in the DMSMAIN server configuration file (`NetServerMainConfig_XXXX.txt`).
2. Enable permission area enforcement by setting the DMS server permission setting PERMIT_CHECK_ENABLED to 1.

```
'------------------------------------------------------------
'e-terrahabitat PERMIT Server Parameters
'------------------------------------------------------------
PERMIT_CHECK_ENABLED = 1
PERMIT_SERVER_IP = 127.0.0.1
PERMIT_SERVER_PORT = 5345
PERMIT_SERVER_TIMER_PERIOD = 9
PERMIT_LOG_ENABLED = 0
PERMIT_JUMPER_OR_MODE = 0
PERMISSION_PER_STATION = 0
```

3. Restart the DMSMAIN server.

The Converter detects the Permission Area ID in multiple locations (i.e., GIS and Internal File).

This procedure uses the Substation Editor PermitAreaId field to perform station-level permission control, which allows using an existing permission area name in SCADA to apply to the DMS model. The PERMIT application database in Habitat should not require additional area entries.

In the example used for this procedure, multiple places with Permit Area ID are defined:

- **GIS Device level:** Permit Area ID
- **GIS Station level:** Permit Area ID
- **Station Internal %DNOMSTATION%:** Permit Area ID field
- **Station Internal Device:** Permit Area ID field

The code checks for permissions in the following order:

1. **Device level (include GIS and Internal Device):** Permit Area ID
2. **GIS Station level:** Permit Area ID
3. **Station Internal level:** Permit Area ID

The "%DNOMSTATION%" element is used to populate the <Substation><PermitAreaId> field only if that field is empty. The field can be empty when the "Substation Properties" dialog box was not used to populate it. Once <Substation><PermitAreaId> is populated, it must be changed via the "Substation Properties" dialog box in the Substation Editor tool.

To edit the station-level Permit Area ID for Substation Editor file:

1. Access the "Substation Properties" dialog box. From the Tools menu, point to Substation Editor, and click Substation Properties.

   The Substation Properties dialog box has a text box to enter `PermitAreaId`.

2. Change "PermitAreaId" value and click OK.
3. Save the file and re-open it to verify that the `PermitAreaId` in the XML file for <Substation> is updated.