

Use GIS Data ReViewer to Inspect ArcFM Feeder Manager Circuits



Copyright © 2010 Esri
All rights reserved.
Printed in the United States of America.

The information contained in this document is the exclusive property of Esri. This work is protected under United States copyright law and other international copyright treaties and conventions. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, or by any information storage or retrieval system, except as expressly permitted in writing by Esri. All requests should be sent to Attention: Contracts and Legal Services Manager, Esri, 380 New York Street, Redlands, CA 92373-8100 USA.

The information contained in this document is subject to change without notice.

Esri, the Esri globe logo, ArcGIS, ArcMap, ArcSDE, ArcCatalog, ArcInfo, www.esri.com, and @esri.com are trademarks, registered trademarks, or service marks of Esri in the United States, the European Community, or certain other jurisdictions. Other companies and products mentioned herein may be trademarks or registered trademarks of their respective trademark owners.

Use GIS Data ReViewer to Inspect ArcFM Feeder Manager Circuits

An Esri White Paper

Contents	Page
Introduction.....	1
ReViewer Session.....	1
ReViewer Table	1
ReViewer Checks.....	2
ReViewer Batch Job	2
ReViewer Service	3
ArcFM Feeder Manager.....	3
Trace All Feeders.....	3
Feeder Symbology	4
FeederInfo Labeling.....	5
FeederInfo Label Expression Advanced.....	6
Feeder Manager Translator.....	9
ReViewer Execute SQL Check.....	9
ReViewer Batch Job Manager	10
ReViewer Toolbar.....	10
ArcFM QA/QC Tool.....	11
Conclusion	13

Use GIS Data ReViewer to Inspect ArcFM Feeder Manager Circuits

Introduction GIS Data ReViewer allows you to conduct visual and automated quality review of data in support of database production. The GIS Data ReViewer environment is made up of a series of tools that support the visual review of geodatabase and shapefile data quality. It is used to identify where corrections, additions, and deletions must be made to spatial data and its attributes; analyze data in your geodatabase; and find possible anomalies that exist. Data validation is performed during ReViewer sessions using data checks as well as the analysis tools that are part of GIS Data ReViewer. Features are symbolized in a consistent manner to facilitate interpretation. This software automates and applies a database process to what was formerly a paper trail of quality control error files and corrections.

ReViewer Session Similar to an edit session, the ReViewer session allows you to interact with your data; however, in a ReViewer session, you are performing quality control tasks. When you start a ReViewer session, you have the ability to review your data, write records to the ReViewer Table, and interact with records in the ReViewer Table. Once the ReViewer session is ended, you can still run checks on the data and do visual quality control but cannot write information to the ReViewer Table. When you are preparing to start a ReViewer session in ArcMap™, you must choose the workspace to use, and you can name the current ReViewer session that is going to contain any anomalies found during the validation process.

You also have the ability to configure more advanced options that allow you to control the way batch jobs are run, access table properties, toggle default checks, and determine the location of database properties. The ReViewer workspace can be created on personal, file, and ArcSDE® geodatabases.

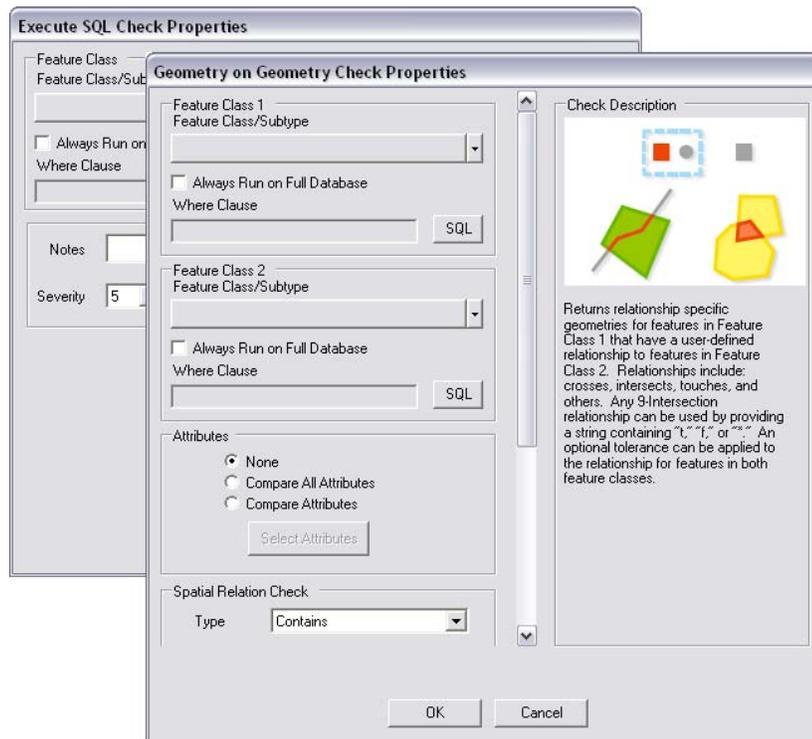
ReViewer Table The ReViewer Table is a tool that stores all the validation problems that have been found using the checks and allows you to analyze the records further. The initial review status, as well as correction and verification status, is stored in the ReViewer Table.

RECORDID	OBJECTID	SUBTYPE	REVIEWERCATEGORY	CATEGORY	REVIEWERCODE
1	-1	AirportAirfield	4	Location Record	998
2	-1	AqueductLine	4	Location Record	998
3	2894	TreesArea	1	Feature Record	165
4	8574	TreesArea	1	Feature Record	165
5	317	BridgeOverpassV	1	Feature Record	165
6	250	PipelinePipeLine	1	Feature Record	165
7	14224	RoadLine	1	Feature Record	166
8	14235	RoadLine	1	Feature Record	166
9	732	TreesArea	1	Feature Record	182
10	733	TreesArea	1	Feature Record	182
11	735	TreesArea	1	Feature Record	182

ReViewer Table

ReViewer Checks

Checks are tools that allow you to validate your data based on specific conditions. Some of the checks allow you to search for conditions, such as polygon slivers or cutbacks, while other checks allow you to search for features that have spatial relationships with each other. For example, you may want to find switches that have been mistakenly placed on top of fuse features.



ReViewer Checks

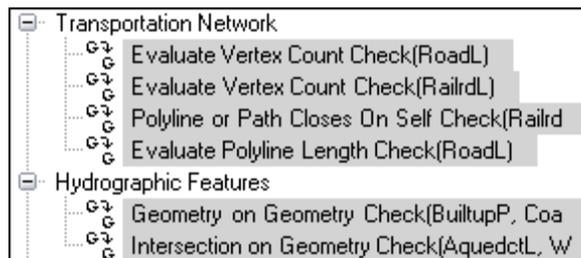
ReViewer Batch Job

Batch jobs are groups of checks that can be saved and run against the data. The checks can be organized and combined into any number of groups within a batch job depending on how you want to classify them. For example, you can divide checks into groups called

Transportation Network and Hydrographic Features to indicate that the checks in these groups relate to these aspects of your database. The checks you add to each of these groups can be applied to different feature classes and can therefore appear in the same group a number of times, depending on the relationships you are evaluating in your data.

Design of the batch job should be the duty of a technical lead in an organization.

The batch job allows users in different locations to utilize a consistent set of automated checks when validating their data.



ReViewer Batch Job

ReViewer Service

ReViewer service is a Microsoft® Windows® service that can be scheduled to run ReViewer batch jobs. Similar to running a batch job using the ReViewer Batch Validate tool, the ReViewer service validates and runs batch jobs and writes the results to the ReViewer Table in a specified ReViewer session. Batch jobs can be scheduled to run once at a specific date and time or to run repeatedly at regular intervals.

ArcFM Feeder Manager

ArcFM's Feeder Manager automatically determines which circuit or sources energizes a piece of electrical equipment with power and tracks which of its electrical phases are energized on the primary side. Feeder Manager responds to edits to the network (such as closing a switch) by updating the FeederID field and energized phase information for all affected features. MMElectricTraceWeight stores all the attribute information about a device or conductor that feeder management needs to perform traces and assign FeederID values. The attribute information contains data on FeederID, tracing impedance (open or closed status) and possibly operating voltage and phasing information. Feeder Manager relies on the weight autoupdater, which is a collection of autoupdaters. The weight autoupdater automatically maintains the value of the MMElectricTraceWeight field for every feature class in the geometric network.

Feeder Manager requires specific classes, fields, relationships, and domains in the geodatabase design. If your geodatabase can't be designed this way, you may need to create a custom weight autoupdater.

Trace All Feeders

ArcFM Trace All Feeders is run on a new geodatabase to properly initialize the Feeder Manager fields. When changes are made to the data after the initialization, Feeder Manager autoupdaters are designed to maintain the FeederID, FeederID2, and FeederInfo fields (and ParentCircuitSourceID for EFM users). There are minimal types of edits that can cause Feeder Manager autoupdaters to *not* maintain that information accurately (use of Esri's Connect and Disconnect tools may be the only example of this type of edit). If this type of edit occurs, select the Trace All Feeders or Trace a Feeder option to update affected conductors and devices in all feeder systems or a selected feeder system. Only

one instance of Trace a Feeder or Trace All Feeders may be run at one time. The tools to trace feeders are available in the following locations:

- **[Trace a Feeder \(ArcCatalog™\)](#)**: This tool traces a single feeder until it comes to the end of the feeder or encounters a de-energized feature.
- **[Trace a Feeder \(ArcMap\)](#)**: This tool traces a single feeder until it comes to the end of the feeder or encounters a de-energized feature.

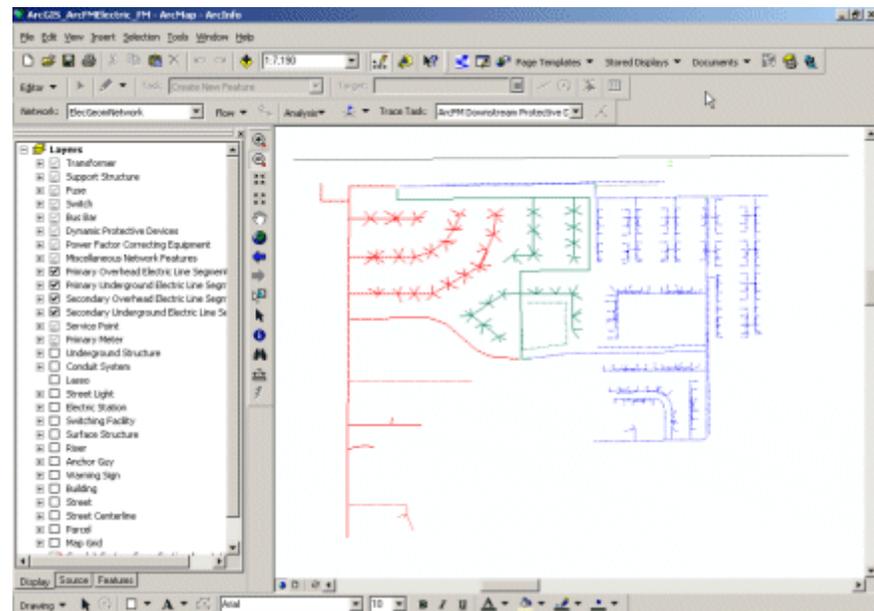
Note: Trace a Feeder stops tracing when it encounters a de-energized feature.

- **[Trace All Feeders \(ArcCatalog\)](#)**: This tool traces all feeders in the geometric network, including those that have no source junction (island) and any feeders downstream from a de-energized junction.
- **[Trace All Feeders \(stand-alone executable\)](#)**: This tool traces all feeders in the geometric network, including those that have no source junction (island) and any feeders downstream from a de-energized junction. This tool also compresses the database after tracing each feeder, making Trace All Feeders faster than when it's run in ArcCatalog.

Note: If Trace All Feeders encounters a de-energized feature, it assigns the FeederID field a value of Null.

Feeder Symbolology

Feeder Manager populates the FeederID, FeederID2, and FeederInfo fields. The FeederID field identifies separate feeder systems and the devices associated with them. The FeederID2 field will be populated if the feature has more than one circuit source.

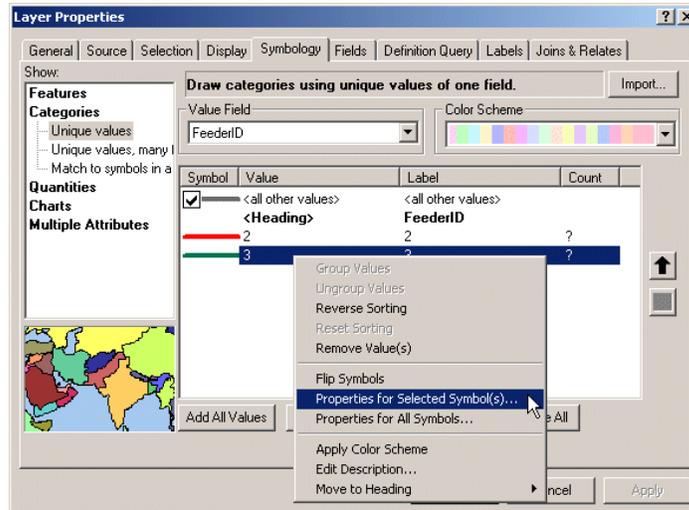


Feeder Symbolology

J-9922

To add electric line segment layers to the map

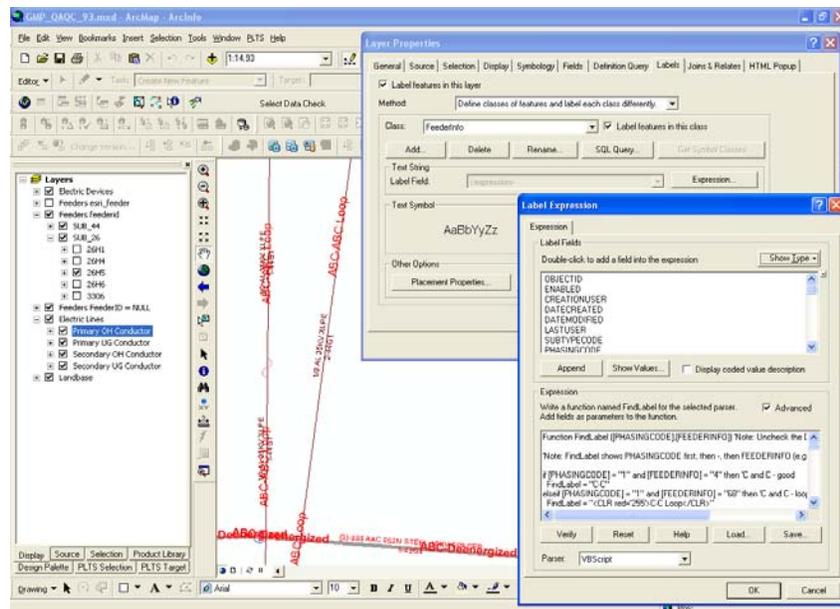
1. Right-click a layer and select Properties. Go to the Symbology tab.
2. Select FeederID in the Value field.
3. Click Add Values or Add All Values.
4. Right-click each value and select Properties for Selected Symbols. Edit the symbol for that particular feature.



Layer Symbology

FeederInfo Labeling

The FeederInfo field contains bit gate information, such as energized phases, loops, extra feeds (double feeds), and islands, and can be used to label the electric line segments.



FeederInfo Label Expression

FeederInfo Label Expression Advanced

Electric line segments labeled using PhasingCode and FeederInfo value combinations—the segments presenting loops, multifeed, de-energized, and bad phase—are labeled in red.

PhaseDesignation domain assigned to PhasingCode field

4 A
2 B
1 C
6 AB
5 AC
3 BC
7 ABC
0 N

```
Function FindLabel ([PHASINGCODE],[FEEDERINFO]) 'Note: Uncheck the Display
Coded Value Description checkbox

'Note: FindLabel shows PHASINGCODE first, then -, then FEEDERINFO (e.g. A-
A)

if [PHASINGCODE] = "1" and [FEEDERINFO] = "4" then 'C and C - good
  FindLabel = "C-C"
elseif [PHASINGCODE] = "1" and [FEEDERINFO] = "68" then 'C and C - loop
  FindLabel = "<CLR red='255'>C-C Loop</CLR>"
elseif [PHASINGCODE] = "1" and [FEEDERINFO] = "132" then 'C and C -
multifeed
  FindLabel = "<CLR red='255'>C-C Multifeed</CLR>"
elseif [PHASINGCODE] = "1" and [FEEDERINFO] = "196" then 'C and C - loop
and multifeed
  FindLabel = "<CLR red='255'>C-C Loop and Multifeed</CLR>"

elseif [PHASINGCODE] = "2" and [FEEDERINFO] = "2" then 'B and B - good
  FindLabel = "B-B"
elseif [PHASINGCODE] = "2" and [FEEDERINFO] = "66" then 'B and B - loop
  FindLabel = "<CLR red='255'>B-B Loop</CLR>"
elseif [PHASINGCODE] = "2" and [FEEDERINFO] = "130" then 'B and B -
multifeed
  FindLabel = "<CLR red='255'>B-B Multifeed</CLR>"
elseif [PHASINGCODE] = "2" and [FEEDERINFO] = "194" then 'B and B - loop
and multifeed
  FindLabel = "<CLR red='255'>B-B Loop and Multifeed</CLR>"

elseif [PHASINGCODE] = "3" and [FEEDERINFO] = "6" then 'BC and BC - good
  FindLabel = "BC-BC"
elseif [PHASINGCODE] = "3" and [FEEDERINFO] = "70" then 'BC and BC - loop
  FindLabel = "<CLR red='255'>BC-BC Loop</CLR>"
elseif [PHASINGCODE] = "3" and [FEEDERINFO] = "134" then 'BC and BC -
multifeed
  FindLabel = "<CLR red='255'>BC-BC Multifeed</CLR>"
elseif [PHASINGCODE] = "3" and [FEEDERINFO] = "198" then 'BC and BC - loop
and multifeed
  FindLabel = "<CLR red='255'>BC-BC Loop and Multifeed</CLR>"
elseif [PHASINGCODE] = "3" and [FEEDERINFO] = "2" then 'BC and B - bad
  FindLabel = "<CLR red='255'>BC-B</CLR>"
elseif [PHASINGCODE] = "3" and [FEEDERINFO] = "4" then 'BC and C - bad
  FindLabel = "<CLR red='255'>BC-C</CLR>"
```

```

elseif [PHASINGCODE] = "4" and [FEEDERINFO] = "1" then 'A and A - good
  FindLabel = "A-A"
elseif [PHASINGCODE] = "4" and [FEEDERINFO] = "65" then 'A and A - loop
  FindLabel = "<CLR red='255'>A-A Loop</CLR>"
elseif [PHASINGCODE] = "4" and [FEEDERINFO] = "129" then 'A and A -
multifeed
  FindLabel = "<CLR red='255'>A-A Multifeed</CLR>"
elseif [PHASINGCODE] = "4" and [FEEDERINFO] = "193" then 'A and A - loop
and multifeed
  FindLabel = "<CLR red='255'>A-A Loop and Multifeed</CLR>"

elseif [PHASINGCODE] = "5" and [FEEDERINFO] = "5" then 'AC and AC - good
  FindLabel = "AC-AC"
elseif [PHASINGCODE] = "5" and [FEEDERINFO] = "69" then 'AC and AC - loop
  FindLabel = "<CLR red='255'>AC-AC Loop</CLR>"
elseif [PHASINGCODE] = "5" and [FEEDERINFO] = "133" then 'AC and AC -
multifeed
  FindLabel = "<CLR red='255'>AC-AC Multifeed</CLR>"
elseif [PHASINGCODE] = "5" and [FEEDERINFO] = "197" then 'AC and AC - loop
and multifeed
  FindLabel = "<CLR red='255'>AC-AC Loop and Multifeed</CLR>"
elseif [PHASINGCODE] = "5" and [FEEDERINFO] = "1" then 'AC and A - bad
  FindLabel = "<CLR red='255'>AC-A</CLR>"
elseif [PHASINGCODE] = "5" and [FEEDERINFO] = "4" then 'AC and C - bad
  FindLabel = "<CLR red='255'>AC-C</CLR>"

elseif [PHASINGCODE] = "6" and [FEEDERINFO] = "3" then 'AB and AB - good
  FindLabel = "AB-AB"
elseif [PHASINGCODE] = "6" and [FEEDERINFO] = "67" then 'AB and AB - loop
  FindLabel = "<CLR red='255'>AB-AB Loop</CLR>"
elseif [PHASINGCODE] = "6" and [FEEDERINFO] = "131" then 'AB and AB -
multifeed
  FindLabel = "<CLR red='255'>AB-AB Multifeed</CLR>"
elseif [PHASINGCODE] = "6" and [FEEDERINFO] = "195" then 'AB and AB - loop
and multifeed
  FindLabel = "<CLR red='255'>AB-AB Loop and Multifeed</CLR>"
elseif [PHASINGCODE] = "6" and [FEEDERINFO] = "1" then 'AB and A - bad
  FindLabel = "<CLR red='255'>AB-A</CLR>"
elseif [PHASINGCODE] = "6" and [FEEDERINFO] = "2" then 'AB and B - bad
  FindLabel = "<CLR red='255'>AB-B</CLR>"

elseif [PHASINGCODE] = "7" and [FEEDERINFO] = "7" then 'ABC and ABC - good
  FindLabel = "ABC-ABC"
elseif [PHASINGCODE] = "7" and [FEEDERINFO] = "71" then 'ABC and ABC -
loop
  FindLabel = "<CLR red='255'>ABC-ABC Loop</CLR>"
elseif [PHASINGCODE] = "7" and [FEEDERINFO] = "135" then 'ABC and ABC -
multifeed
  FindLabel = "<CLR red='255'>ABC-ABC Multifeed</CLR>"
elseif [PHASINGCODE] = "7" and [FEEDERINFO] = "199" then 'ABC and ABC -
loop and multifeed
  FindLabel = "<CLR red='255'>ABC-ABC Loop and Multifeed</CLR>"
elseif [PHASINGCODE] = "7" and [FEEDERINFO] = "1" then 'ABC and A - bad
  FindLabel = "<CLR red='255'>ABC-A</CLR>"
elseif [PHASINGCODE] = "7" and [FEEDERINFO] = "2" then 'ABC and B - bad
  FindLabel = "<CLR red='255'>ABC-B</CLR>"
elseif [PHASINGCODE] = "7" and [FEEDERINFO] = "3" then 'ABC and AB - bad
  FindLabel = "<CLR red='255'>ABC-AB</CLR>"
elseif [PHASINGCODE] = "7" and [FEEDERINFO] = "4" then 'ABC and C - bad
  FindLabel = "<CLR red='255'>ABC-C</CLR>"
elseif [PHASINGCODE] = "7" and [FEEDERINFO] = "5" then 'ABC and AC - bad
  FindLabel = "<CLR red='255'>ABC-AC</CLR>"
elseif [PHASINGCODE] = "7" and [FEEDERINFO] = "6" then 'ABC and BC - bad
  FindLabel = "<CLR red='255'>ABC-BC</CLR>"

```

```
elseif [PHASINGCODE] = "1" and [FEEDERINFO] = "8" then 'Deenergized and
FeederID is null
    FindLabel = "<CLR red='255'><BOL>C-Deenergized</BOL></CLR>"
elseif [PHASINGCODE] = "2" and [FEEDERINFO] = "8" then 'Deenergized and
FeederID is null
    FindLabel = "<CLR red='255'><BOL>B-Deenergized</BOL></CLR>"
elseif [PHASINGCODE] = "3" and [FEEDERINFO] = "8" then 'Deenergized and
FeederID is null
    FindLabel = "<CLR red='255'><BOL>BC-Deenergized</BOL></CLR>"
elseif [PHASINGCODE] = "4" and [FEEDERINFO] = "8" then 'Deenergized and
FeederID is null
    FindLabel = "<CLR red='255'><BOL>A-Deenergized</BOL></CLR>"
elseif [PHASINGCODE] = "5" and [FEEDERINFO] = "8" then 'Deenergized and
FeederID is null
    FindLabel = "<CLR red='255'><BOL>AC-Deenergized</BOL></CLR>"
elseif [PHASINGCODE] = "6" and [FEEDERINFO] = "8" then 'Deenergized and
FeederID is null
    FindLabel = "<CLR red='255'><BOL>AB-Deenergized</BOL></CLR>"
elseif [PHASINGCODE] = "7" and [FEEDERINFO] = "8" then 'Deenergized and
FeederID is null
    FindLabel = "<CLR red='255'><BOL>ABC-Deenergized</BOL></CLR>"

elseif [PHASINGCODE] = "1" and [FEEDERINFO] = "0" then 'Deenergized and
FeederID is null
    FindLabel = "<CLR red='255'><BOL>C-Deenergized</BOL></CLR>"
elseif [PHASINGCODE] = "2" and [FEEDERINFO] = "0" then 'Deenergized and
FeederID is null
    FindLabel = "<CLR red='255'><BOL>B-Deenergized</BOL></CLR>"
elseif [PHASINGCODE] = "3" and [FEEDERINFO] = "0" then 'Deenergized and
FeederID is null
    FindLabel = "<CLR red='255'><BOL>BC-Deenergized</BOL></CLR>"
elseif [PHASINGCODE] = "4" and [FEEDERINFO] = "0" then 'Deenergized and
FeederID is null
    FindLabel = "<CLR red='255'><BOL>A-Deenergized</BOL></CLR>"
elseif [PHASINGCODE] = "5" and [FEEDERINFO] = "0" then 'Deenergized and
FeederID is null
    FindLabel = "<CLR red='255'><BOL>AC-Deenergized</BOL></CLR>"
elseif [PHASINGCODE] = "6" and [FEEDERINFO] = "0" then 'Deenergized and
FeederID is null
    FindLabel = "<CLR red='255'><BOL>AB-Deenergized</BOL></CLR>"
elseif [PHASINGCODE] = "7" and [FEEDERINFO] = "0" then 'Deenergized and
FeederID is null
    FindLabel = "<CLR red='255'><BOL>ABC-Deenergized</BOL></CLR>"

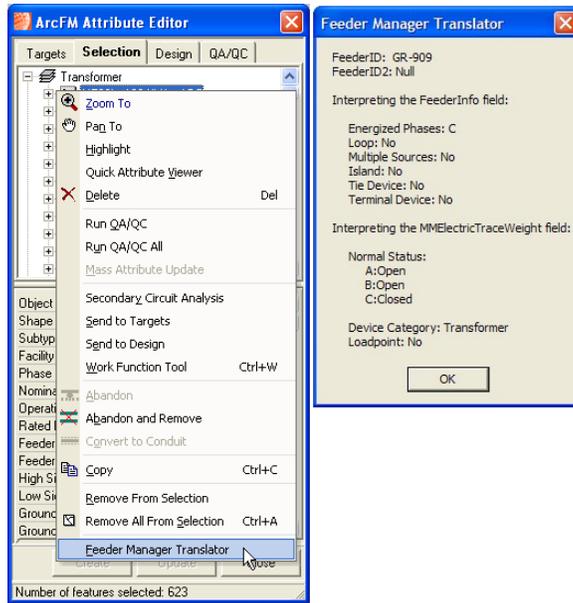
else 'all other cases
    FindLabel = "<CLR red='255'><BOL>Use FM Translator</BOL></CLR>"
end if
End Function
```

Note: Contact Telvent Technical Support to learn the FeederInfo bit gate values for your ArcFM release.

J-9922

Feeder Manager Translator

The Feeder Manager Translator displays feeder information about the selected feature.

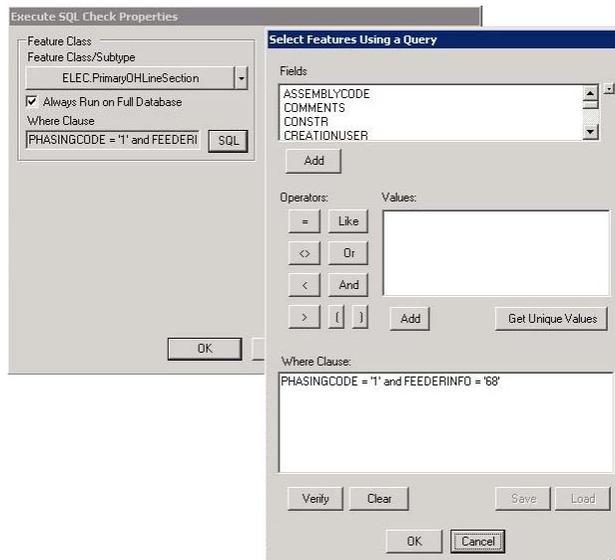


Feeder Manager Translator

ReViewer Execute SQL Check

A SQL query can be used to select features based on a combination of attribute values, for example, if you wanted to find loop, multifeed, de-energized, and bad phase electric line segments using the FeederInfo field. You can record the invalid features in the ReViewer Table for later inspection and correction.

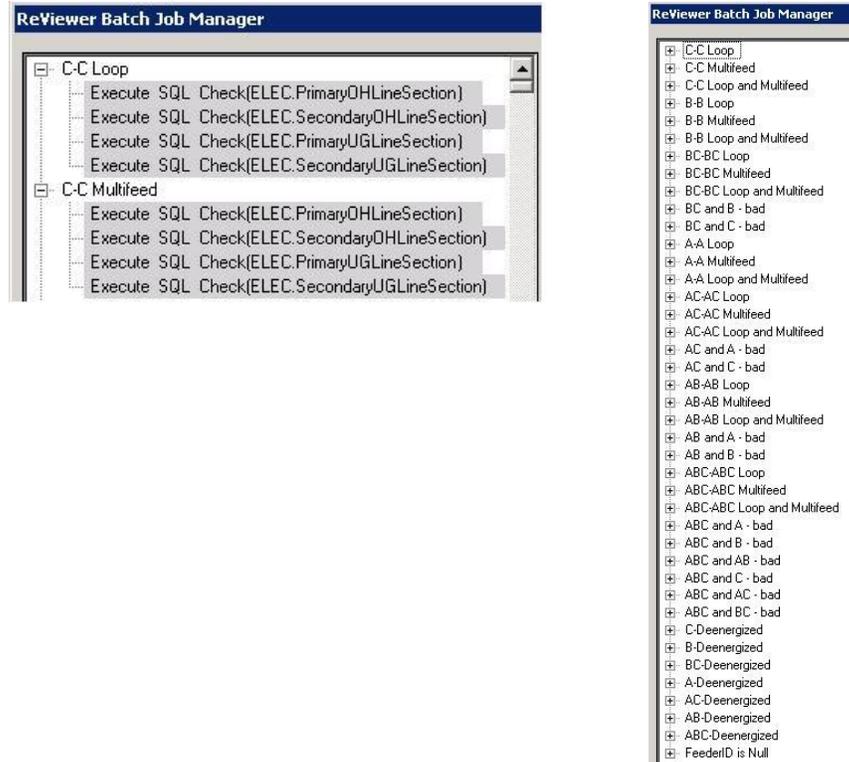
Example: [PHASINGCODE] = "1" and [FEEDERINFO] = "68" then 'C and C - loop



Execute SQL Check

ReViewer Batch Job Manager

In the ReViewer Batch Job Manager, you can add checks to groups, edit checks, and remove them from groups. You can edit the parameters they use to correct an error that has been found during validation or to change the feature class on which it is run. Checks can be added individually to ensure that data complies with product specifications. You can also duplicate checks and apply them to several different feature classes at once. The Duplicate Check Generator allows you to include a check more than once in a batch job without having to individually configure it each time.



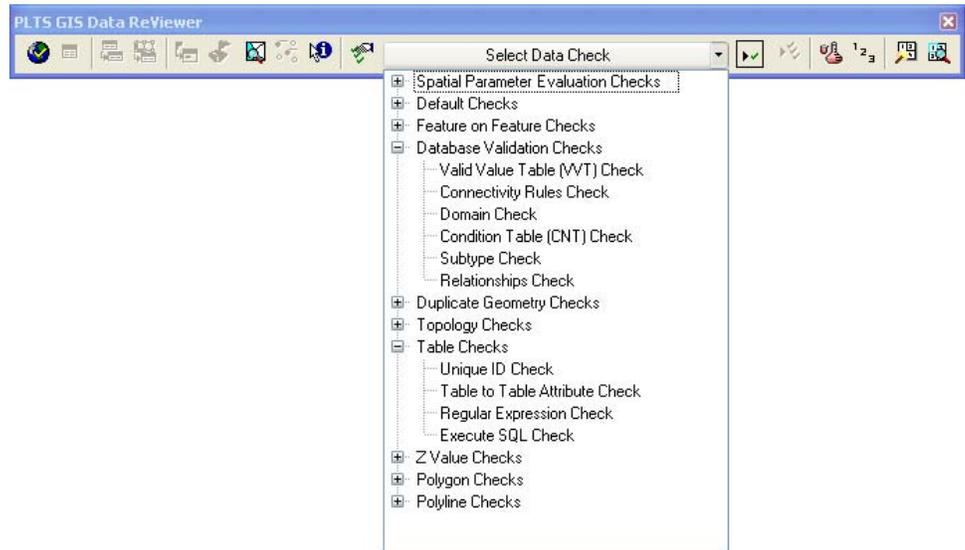
ReViewer Batch Job Manager

ReViewer Toolbar

GIS Data ReViewer offers many checks that can be selected from the drop-down list on the ReViewer toolbar. These checks allow you to perform geometric and attribute validation as well as ensure data integrity. All the validation checks can be applied to an entire feature class or database, features within the current extent, or only the selected set of features. You can also use a SQL query WHERE clause to limit which features to evaluate for a particular check.

In deciding which checks to run on their data, it is customary for people using GIS Data ReViewer to have a data quality specification; for instance, a specification may require buildings to be at least 25 meters from a road.

The Geometry on Geometry check can be used to find any buildings that are less than 25 meters from a feature in the road feature class. There are also checks you can use to ensure that your data does not contain any invalid features. This includes features that do not have the correct number of vertices for their geometry type or have a null extent.



ReViewer Toolbar

Check Name	Description
Connectivity	Finds features that are part of a geometric network and that violate connectivity rules
Domain	Validates coded value and range domains to ensure that all values meet domain constraints
Subtype	Searches for feature classes with improper or null (optional) subtypes
Relationships	Searches for records that are orphans or have improper cardinality in a relationship class
Geometry on Geometry	Finds features that have a specific spatial relationship, either from two different feature classes or within the same feature class, for example, finding transformers on top of switches
Intersection on Geometry	Finds features from one feature class that are within a specified tolerance of an intersection between two other features from two feature classes or within the same feature class
Invalid Geometry	Finds features whose geometry is empty, nothing, or not simple as well as those with empty envelopes
Duplicate Geometry	Finds features of the same geometry type that are collocated and optionally share attributes (Features can be either from two different feature classes or within the same feature class.)

Note: To see the complete list of checks, visit www.esri.com/datareviewer (brochures/poster).

ArcFM QA/QC Tool

Several sample validation rules are included with ArcFM and configured in ArcCatalog. You may configure any one of the validation rules or all of them.

The ArcFM QA tool compares feature attribute values to existing validation and connectivity rules in the geodatabase to ensure that attribute values do not conflict with these rules. The results are displayed on the ArcFM Attribute Editor QA/QC tab.

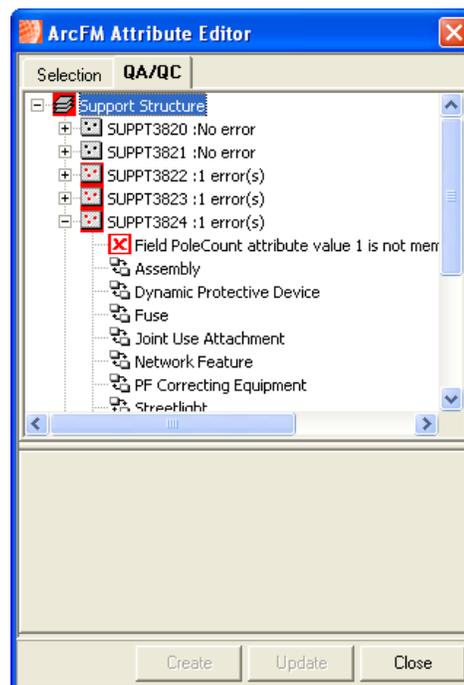
Depending on the option setting, the results may display only the invalid items from the selection or the entire selection with the invalid items identified.

Types of validation rules that the QA tool checks are

- Coded domain values are the set of valid values an attribute can have.
- Range domains are the range of numeric values that a value must fall between.
- Contingent attribute validity is a set of valid values that depends on the value of another attribute.
- Connectivity rules govern which network features can be connected.

Your geodatabase schema can impact the performance of the ArcFM QA/QC tool. When there are multiple relationships, the QA/QC requires more time to evaluate them. Performance improves when both layers related to the object are included in the map.

You can export the QA/QC results to a printable HTML or XML file and import QA/QC information from XML file.



ArcFM QA Tool

Conclusion

GIS Data ReViewer can analyze data both visually and in terms of a geodatabase's contents. GIS Data ReViewer includes data checks that can be used to validate many different aspects of a database. These aspects include feature geometry, topology, relationship classes, and spatial relationships between geometries.

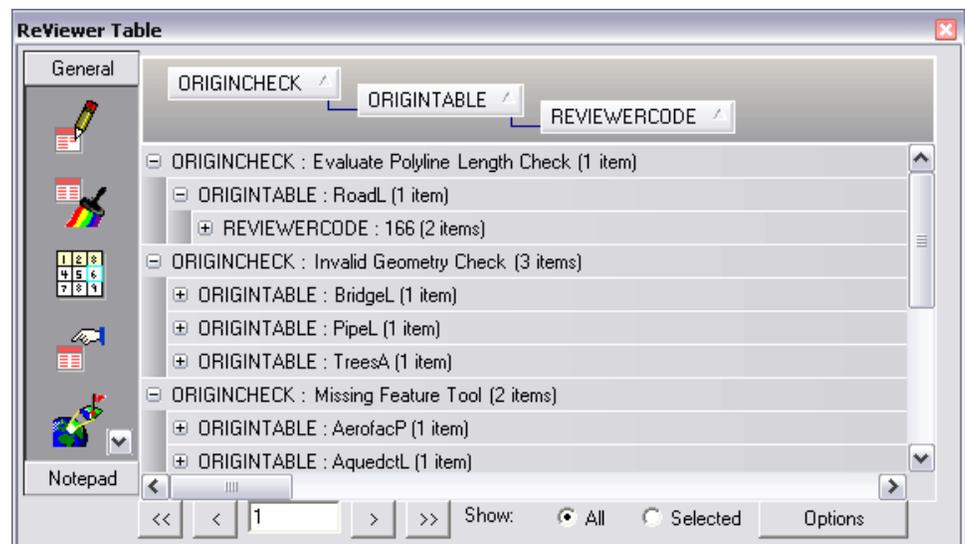
The checks can also be grouped into ReViewer Batch Job (.rbj) files, which allow you to run several checks on your geodatabase at the same time. These batch jobs can be run using the ReViewer Batch Validation tool.

Other analysis tasks include

- Sampling—Generate a report on a specified number or percentage of features in the geodatabase from selected feature classes.
- Obtaining a total feature count—View the total number of features in the geodatabase according to both feature class and subtype.
- Obtaining the frequency of field values—View the number of times a particular field value occurs in a particular feature class.

Visual analysis with GIS Data ReViewer can be managed using the OverView Window, which allows you to record and view which areas have been reviewed. Polygon grids can be created over the study area, with grid cell size determined either by a specified number of rows and columns or map units. Missing features can be drawn in the map using the Notepad and Flag Missing Feature tools.

The ReViewer Table is a tool that stores all the validation problems that have been found using the checks and allows you to analyze the records further. The initial review status, as well as correction and verification status, is stored in the ReViewer Table.



ReViewer Table (sort and group records)

Using tools in the ReViewer Table, you can also

- Select and/or zoom to ReViewer Table records and associated feature geometries.
- Enter correction and verification information for one or many ReViewer Table records.
- Symbolize the records according to their review, correction, and verification status.
- Sort and group records using any of the fields.
- Set table properties.
- Associate a polygon grid with the ReViewer Table.
- Generate statistics on records in the ReViewer Table.
- Repeat existing table records.
- View the locations of missing features.

For more information on GIS Data ReViewer, visit www.esri.com/datareviewer or e-mail datareviewer@esri.com.



About Esri

Since 1969, Esri has been helping organizations map and model our world. Esri's GIS software tools and methodologies enable these organizations to effectively analyze and manage their geographic information and make better decisions. They are supported by our experienced and knowledgeable staff and extensive network of business partners and international distributors.

A full-service GIS company, Esri supports the implementation of GIS technology on desktops, servers, online services, and mobile devices. These GIS solutions are flexible, customizable, and easy to use.

Our Focus

Esri software is used by hundreds of thousands of organizations that apply GIS to solve problems and make our world a better place to live. We pay close attention to our users to ensure they have the best tools possible to accomplish their missions. A comprehensive suite of training options offered worldwide helps our users fully leverage their GIS applications.

Esri is a socially conscious business, actively supporting organizations involved in education, conservation, sustainable development, and humanitarian affairs.

Contact Esri

1-800-GIS-XPRT (1-800-447-9778)

Phone: 909-793-2853

Fax: 909-793-5953

info@esri.com

www.esri.com

Offices worldwide

www.esri.com/locations