openFLE

open source Fault Location Engine

1024887



openFLE v1.0

open Source Fault Location Engine

1024887

Software Manual, November 2012

DISCLAIMER OF WARRANTIES AND LIMITATION OF LIABILITIES

EPRI RESERVES ALL RIGHTS IN THE SOFTWARE MANUAL AS DELIVERED. THE SOFTWARE MANUAL OR ANY PORTION THEREOF MAY NOT BE REPRODUCED IN ANY FORM WHATSOEVER, WITHOUT THE CONSENT OF EPRI.

THIS NOTICE MAY NOT BE REMOVED FROM THE PROGRAM BY ANY USER THEREOF.

NEITHER EPRI, ANY MEMBER OF EPRI, THE ORGANIZATION(S) BELOW, NOR ANY PERSON ACTING ON BEHALF OF ANY OF THEM:

1. MAKES ANY WARRANTY OR REPRESENTATION WHATSOEVER, EXPRESS OR IMPLIED, INCLUDING ANY WARRANTY OF MERCHANTABILITY OR FITNESS OF ANY PURPOSE WITH RESPECT TO THE SOFTWARE MANUAL ; OR

2. ASSUMES ANY LIABILITY WHATSOEVER WITH RESPECT TO ANY USE OF THE SOFTWARE MANUAL OR ANY PORTION THEREOF OR WITH RESPECT TO ANY DAMAGES WHICH MAY RESULT FROM SUCH USE.

RESTRICTED RIGHTS LEGEND: USE, DUPLICATION, OR DISCLOSURE BY THE GOVERNMENT IS SUBJECT TO RESTRICTION AS SET FORTH IN PARAGRAPH (G) (3) (I), WITH THE EXCEPTION OF PARAGRAPH (G) (3) (I) (B) (5), OF THE RIGHTS IN TECHNICAL DATA AND COMPUTER SOFTWARE CLAUSE IN FAR 52.227-14, ALTERNATE III.

REFERENCE HEREIN TO ANY SPECIFIC COMMERCIAL PRODUCT, PROCESS, OR SERVICE BY ITS TRADE NAME, TRADEMARK, MANUFACTURER, OR OTHERWISE, DOES NOT NECESSARILY CONSTITUTE OR IMPLY ITS ENDORSEMENT, RECOMMENDATION, OR FAVORING BY EPRI..

The embodiments of this Program and supporting materials may be independently available from Electric Power Software Center (EPSC) for an appropriate distribution fee.

Electric Power Software Center (EPSC)  
9625 Research Drive  
Charlotte, NC 28262

The following organization(s), under contract to EPRI, prepared this report:

Grid Protection Alliance, Inc.

NOTE

For further information about EPRI, call the EPRI Customer Assistance Center at 800.313.3774 or   
e-mail askepri@epri.com.

Electric Power Research Institute, EPRI, and TOGETHER…SHAPING THE FUTURE OF ELECTRICITY are registered service marks of the Electric Power Research Institute, Inc.

Copyright © 2012 Electric Power Research Institute, Inc. All rights reserved.

Acknowledgments

The following organization(s), under contract to the Electric Power Research Institute (EPRI), prepared this report:

Grid Protection Alliance, Inc.  
1206 Broad Street  
Chattanooga, TN 37402

Principal Investigator  
F. Robertson

This report describes research sponsored by EPRI

Microsoft, Windows, Windows Vista, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Software Description

The open source Fault Location Engine (openFLE) analyzes grid-sourced data to locate faults on transmission lines.

Description

Using a fault location algorithm developed by EPRI, openFLE examines disturbance data from fault recorders and power quality meters to locate faults on transmission lines.

openFLE runs as a Windows service that processes data files as they arrive at the host computer. It can read data in two standard formats: Power Quality Data Interchange Format (PQDIF), which is the format specified by IEEE Std. 1159.3, and Common Format for Transient Data Exchange (COMTRADE), which is the format specified by the draft standard IEEE PC37.111.

Benefits and Value

openFLE offers the following benefits and values:

* By identifying where to send repair crews, the fault location algorithm in openFLE can save time.
* By running as a Windows service, openFLE can seamlessly integrate into existing data processing streams in the enterprise.
* As an open source project, openFLE facilitates collaborative efforts as other contributors share algorithms or make other improvements to the code base.

Platform Requirements

Windows® XP, Windows Vista®, Windows 7, or Windows Server® 2008

Keywords

Fault Location  
Open Source  
PQDIF  
COMTRADE

Contents

*1* Introduction 1-1

*2* Installing openFLE 2-1

Installation of EPRI Software at Client Site 2-1

If you experience difficulties accessing the application 2-1

System Requirements 2-1

Before Installation 2-2

Installation Procedure 2-2

*3* Using openFLE 3-1

Tutorial 3-1

Running the openFLE Manager 3-1

Input fields and Controls in the Manager Window 3-2

Folders 3-3

Fault Detection 3-5

Fault Location 3-6

Units 3-7

Fault Detection and Fault Location Algorithm drop down selections 3-8

Control Buttons 3-9

Closing the openFLE Manager 3-11

Stopping the openFLE Service 3-12

Perform Fault Location 3-13

*A* Example of input xml file 1

Event File Definition - An Event File Definition (XML) File 1

*B* Examples of output files in results folder 1

Log File - A Log (TXT) File 1

Cycle Data - A Portion of a Cycle Data (CSV) File 1

Measurement Data - A Portion of a Measurement Data (CSV) File 2

Results File – A Results (XML) File 3

Other Files 3

List of Figures

Figure 2‑1 openFLE Installer: Extract All 2-2

Figure 2‑2 openFLE Installer: Destination for Extracted Files (on Windows 7) 2-3

Figure 2‑3 openFLE Installer: Setup Folder (on Windows 7) 2-3

Figure 2‑4 openFLE Installer: Download .NET Framework 4 Error 2-4

Figure 2‑5 .NET Framework 4 Download: Web Page 2-4

Figure 2‑6 .NET Framework 4 Download: Microsoft Suggestions 2-5

Figure 2‑7 File Download in Internet Explorer 9 2-5

Figure 2‑8 .NET Framework 4 Installer: Download Progress Window 2-6

Figure 2‑9 .NET Framework 4 Installer: Initialization Progress Windows 2-6

Figure 2‑10 .NET Framework 4 Installer: .NET Framework 4 Setup Window 2-6

Figure 2‑11 .NET Framework 4 Installer: Installation Is Complete Window 2-7

Figure 2‑12 openFLE Installer: Welcome Window 2-8

Figure 2‑13 openFLE Installer: Select Installation Folder Window 2-9

Figure 2‑14 openFLE Installer: Browse for Folder Window 2-10

Figure 2‑15 openFLE Installer: Confirm Installation Window 2-10

Figure 2‑16 openFLE Installer: Installation Progress Window 2-11

Figure 2‑17 openFLE Installer: Exit the Installer Confirmation 2-11

Figure 2‑18 openFLE Installer: Installation Interrupted Window 2-12

Figure 2‑19 openFLE Installer: Installation Complete Window 2-13

Figure 2‑20 openFLEManager.exe (in Windows 7) 2-14

Figure 3‑1 openFLEManager.exe 3-1

Figure 3‑2 openFLE: The openFLE Manager window 3-2

Figure 3‑3 openFLE: Process Delay 3-4

Figure 3‑4 openFLE: The Manager window showing a modified Results folder path 3-5

Figure 3‑5 openFLE: The Manager window with default Fault detection input field contents 3-6

Figure 3‑6 openFLE: The Manager window with default Fault location input field contents 3-7

Figure 3‑7 openFLE: The Manager window showing Units drop down list 3-8

Figure 3‑8 openFLE: Fault Detection and Fault Location algorithm drop down lists 3-9

Figure 3‑9 openFLE: Save Button and confirmation dialog box. 3-10

Figure 3‑10 openFLE: Console Monitor Button, Console Monitor Window, Red X to close. 3-11

Figure 3‑11 openFLE: openFLE Manager Red X to close. 3-12

Figure 3‑12 openFLE: Stopping the openFLE service 3-13

Figure 3‑13 openFLE: Input COMTRADE and xml files for Event 5043 3-13

Figure 3‑14 openFLE: Contents of Results folder after event 5043 processed 3-14

Figure 3‑15 openFLE: PQDIF input file example. 3-14

# Introduction

The open source Fault Location Engine (openFLE) analyzes power quality event data from remote sensing devices, such as disturbance recorders or power quality monitors, to locate a fault on a transmission line. To determine whether or not a fault occurred, openFLE reviews the event data. If a fault is found, then openFLE combines the event data with specific parameters of the electric network model to identify what type of fault occurred as well as to calculate the distance to the fault on the affected transmission line.

Event data can be formatted in either COMTRADE or PQDIF format. openFLE uses openPQDIF, which is another open source project published by EPRI, to extract waveform data from input files in PQDIF format.

openFLE runs as a Windows service and is bundled with a configuration manager called openFLE Manager, which runs as a Windows desktop program. openFLE Manager is used to select the folders for input and output files, to set the unit of measure for calculations (miles or kilometers), and to select the algorithm to apply when analyzing data. The service detects input files as they arrive in the specified folder and then creates results in the output folder. After the analysis is complete, input files are moved to the output folder.

As of 2012, only one fault location algorithm is available. By leveraging the collaborative nature of open source software, openFLE can be expanded to include additional algorithms.

# Installing openFLE

## Installation of EPRI Software at Client Site

EPRI develops software using a number of third party software products and tools that run on various operating systems and server platforms. Reports from the software industry suggest there are known security issues with some products and systems. EPRI recommends that, if you are using EPRI software, you review its use with your Information Technology (IT) department and their overall strategy to ensure that all recommended security updates and patches are installed as needed in your corporation. If you have any concerns, please call the EPRI Customer Assistance Center (CAC) at 1-800-313-3774 (or email [askepri@epri.com](mailto:askepri@epri.com)).

## If you experience difficulties accessing the application

If you experience difficulties accessing the application after standard installation on Windows XP, Windows Vista, or Windows 7, please consult your IT department personnel to have proper access permissions set up for your use. If the problem cannot be resolved, please call the EPRI Customer Assistance Center (CAC) at 1-800-313-3774 (or email [askepri@epri.com](mailto:askepri@epri.com)).

## System Requirements

* One of the following Microsoft® Windows® operating systems:
* Windows XP with Service Pack 3 (SP3)
* Windows Vista® with SP2
* Windows 7 32-bit or 64-bit with SP1
* Windows Server® 2008 Standard with SP2
* 800 MHz processor
* 512 MB of memory
* 5 MB of free hard drive space
* Microsoft .NET Framework 4 \*

\* If necessary and possible, the openFLE installer will install Microsoft .NET Framework 4 for you. On Windows Server 2008, you must use the Add Features Wizard instead of the openFLE installer to install Microsoft .NET Framework 4.

## Before Installation

In order to install openFLE, you must have Administrator privileges. If you do not have Administrator privileges, ask your IT department to help you.

In order to run openFLE, Microsoft .NET Framework 4 must be installed. If Microsoft .NET Framework 4 is not already installed, the openFLE installer automatically attempts to install it for you. After the installation begins, you can cancel it, but this is not recommended, as openFLE cannot function without it. On Windows Server 2008, you must use the Add Features Wizard instead of the openFLE installer to install Microsoft .NET Framework 4.

The openFLE installer is available for 32-bit and 64-bit versions of selected Windows versions. If you are not sure which installer you should use, contact your IT department. The installation illustrated below is the 64-bit installation of openFLE; the installation procedure is the same for the 32-bit installation.

## Installation Procedure

These steps assume that you are installing openFLE on Windows 7. The steps may vary slightly on other operating systems. To install openFLE:

1. Right-click the .zip file, and then click **Extract All (**Figure 2‑1).

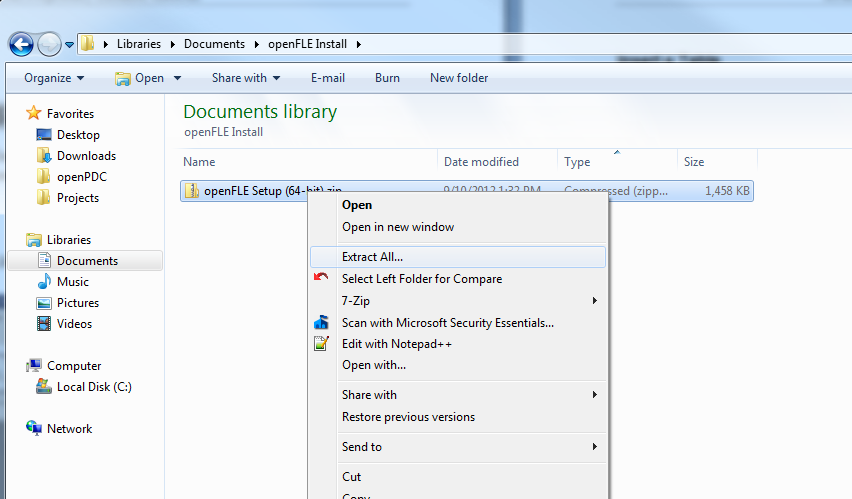


Figure ‑  
openFLE Installer: Extract All

1. Click the **Extract** button (Figure 2‑2).

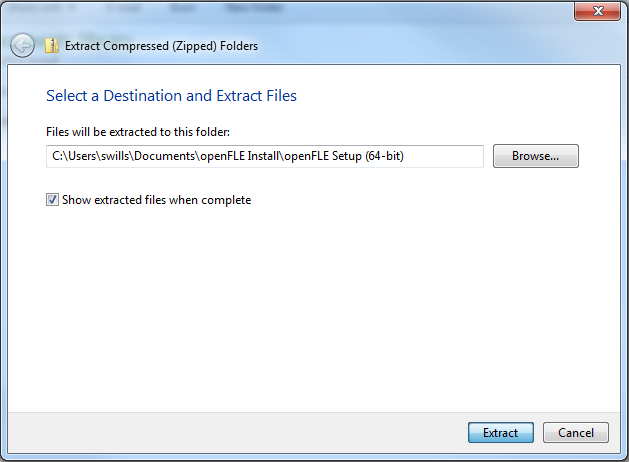


Figure ‑  
openFLE Installer: Destination for Extracted Files (on Windows 7)

1. After the files have unzipped, browse to the **Setup** folder (Figure 2‑3) and double-click **setup.exe** to start the installer.

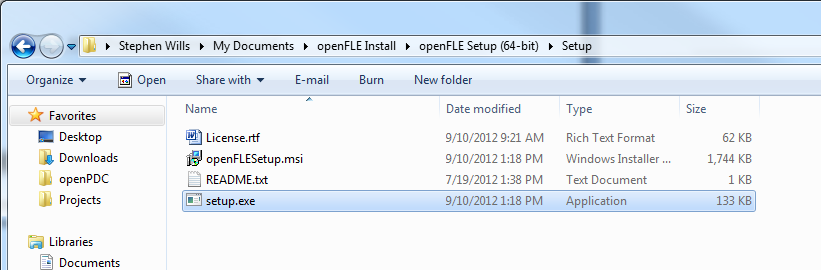


Figure ‑  
openFLE Installer: Setup Folder (on Windows 7)

The openFLE setup program begins its initialization process. During this time, it checks to see if the .NET Framework 4 is installed. If it is installed, then the openFLE installer displays its license agreement followed by the Welcome window (Figure 2‑12). Otherwise, an error message () is displayed.   
  
**Note:** If the .NET Framework 4 is already installed on your computer proceed to step 13.

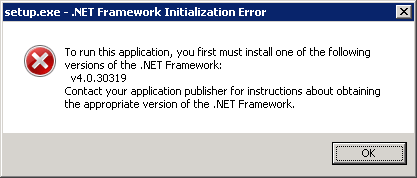


Figure ‑  
openFLE Installer: Download .NET Framework 4 Error

1. Navigate to the download page for the .NET Framework 4 located at <http://www.microsoft.com/en-us/download/details.aspx?id=17718>.
2. On the Microsoft .NET Framework 4 page (), click **Download** to start the download of the .NET Framework 4 installer.   
     
   **Note:** For best results, allow pop-ups on this page.

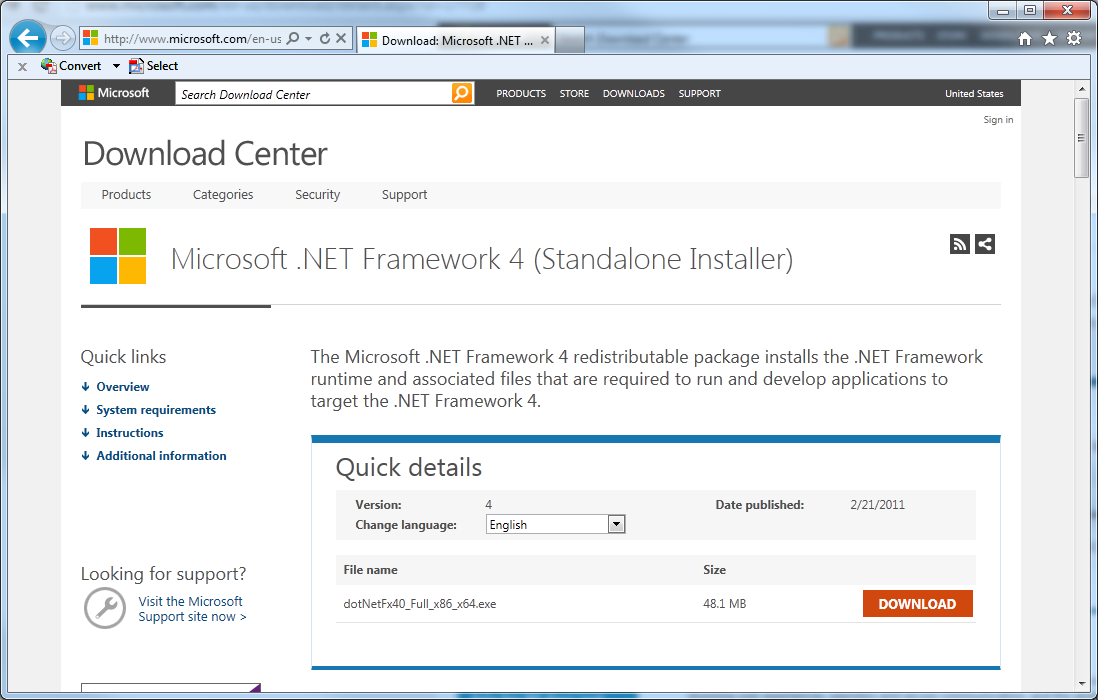


Figure ‑5  
.NET Framework 4 Download: Web Page

1. Microsoft offers suggestions for additional downloads that are not needed to run openFLE. If you do not wish to install the additional downloads, click **NO THANKS AND CONTINUE** to proceed to the next step.

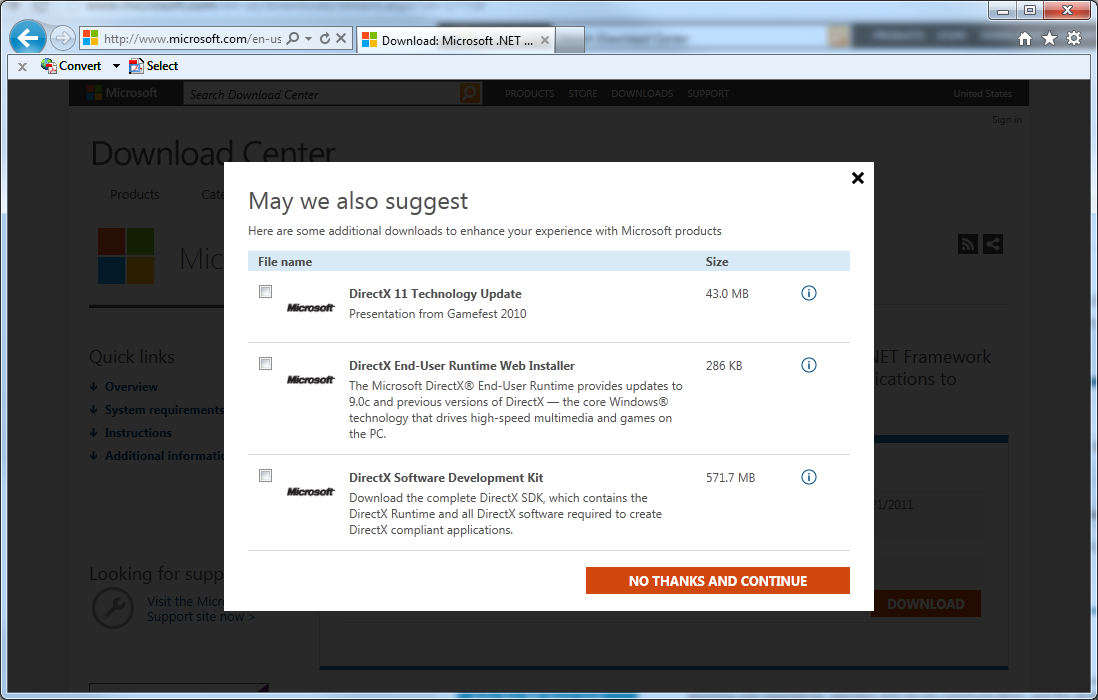


Figure ‑6  
.NET Framework 4 Download: Microsoft Suggestions

1. Before the download begins, a message appears at the bottom of the screen ().   
     
   You can click either **Run** or **Save**. Click **Run** to begin the download and installation of the .NET Framework 4 software. You can instead choose to click **Save**, but these instructions do not include the steps that you must follow for that scenario.



Figure ‑  
File Download in Internet Explorer 9

**Note:** The message box in is specific to Internet Explorer 9. In response to a request to download a file, other browsers may behave differently and/or display different messages.

1. After you click **Run**, the file **dotNetFx40\_Full\_x86\_x64.exe** begins to download, and a progress message appears (Figure 2‑8).



Figure ‑8  
.NET Framework 4 Installer: Download Progress Window

1. Once the download is complete, the installer will run automatically. As the framework installer initializes, one message box showing a progress bar appears (Figure 2‑9).

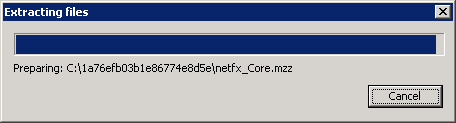


Figure ‑9  
.NET Framework 4 Installer: Initialization Progress Windows

1. After it initializes, the framework installer shows its welcome window, which also contains the license agreement for the .NET Framework 4 .0. You must accept the terms of the license agreement before the framework can install.  
     
   Click “**I have read and accept the license terms**.” and then click **Install**.

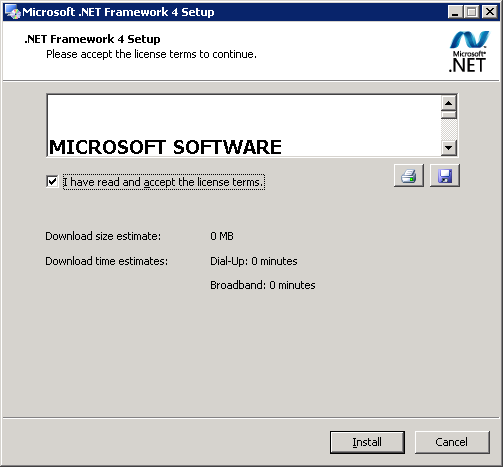


Figure ‑10  
.NET Framework 4 Installer: .NET Framework 4 Setup Window

1. The framework installer begins the process of downloading and installing the .NET Framework 4 (). You may be asked to exit running programs during this process.

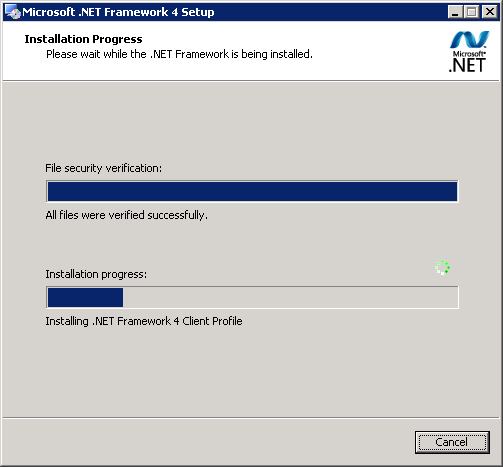


Figure 2-11  
.NET Framework 4 Installer: Installation Progress Windows

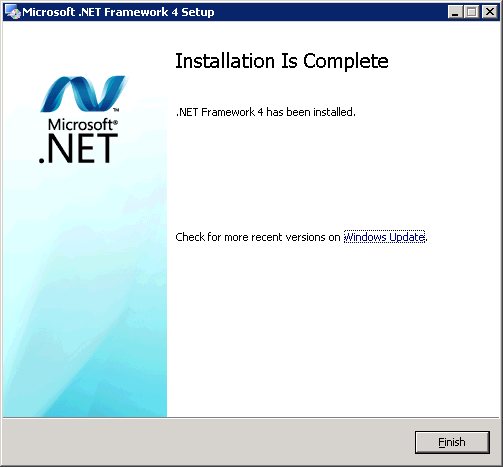


Figure ‑11  
.NET Framework 4 Installer: Installation Is Complete Window

1. After the .NET Framework 4 is downloaded and installed, the Installation Is Complete window appears (). Click **Finish** to close the framework installer and return the openFLE installer. If you are asked to restart your computer, please do so before continuing.
2. To run the openFLE Setup Wizard, double-click **setup.exe**.  
     
   **Note:** If you get a warning about an unknown publisher, click Run to continue.
3. The Welcome window is displayed (Figure 2‑12). Click **Next**.
4. The Select Installation Folder window is displayed (Figure 2‑13) with a suggested installation folder. If you prefer to install in another location, click **Browse** to select your preferred installation folder. Otherwise, proceed to step 16.

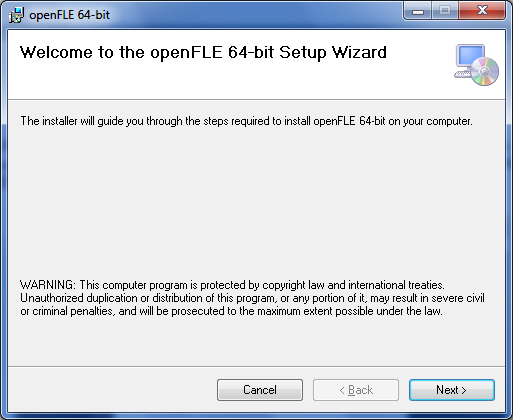


Figure ‑  
openFLE Installer: Welcome Window

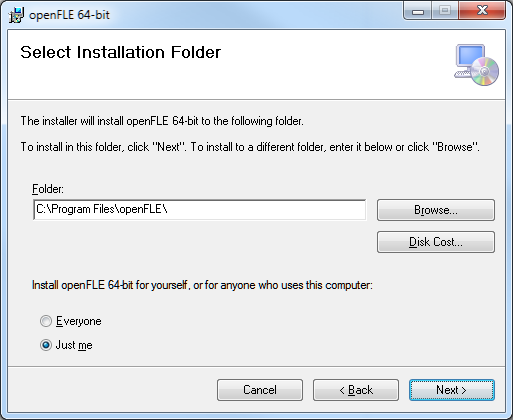


Figure ‑  
openFLE Installer: Select Installation Folder Window

1. If you clicked **Browse** to select a different installation folder, the Browse for Folder window appears (Figure 2‑14). If you need assistance selecting an installation folder using this window, contact your company’s IT department.   
     
   If you would rather use the originally suggested installation folder, click **Cancel**.  
     
   Once you have decided on an installation folder, click **OK** to redisplay the Select Installation Folder window (Figure 2‑13).
2. On the Select Installation Folder window, click **Next**.   
     
   **Note:** The **Drop** and **Results** folders used for input and output files, respectively, are created in the installation folder. These locations can be changed through the openFLE Manager.
3. The Confirm Installation window is displayed (Figure 2‑15). Click **Next** to confirm installation.

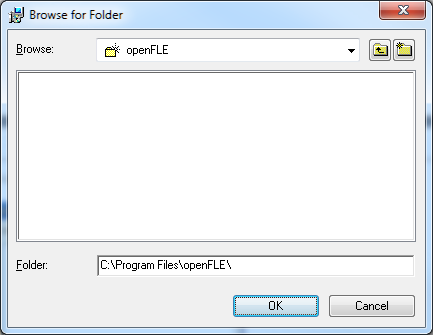


Figure ‑  
openFLE Installer: Browse for Folder Window

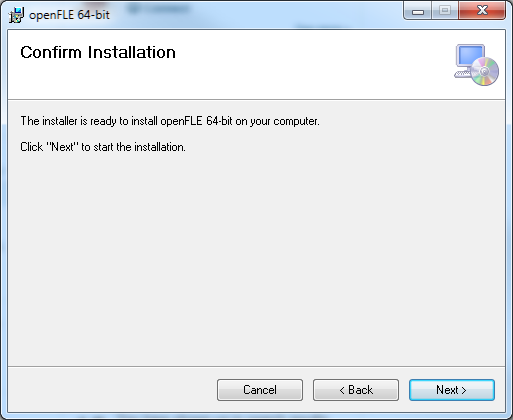


Figure ‑  
openFLE Installer: Confirm Installation Window

1. The installer displays a progress window (Figure 2‑16). No action is required during the installation process unless you desire to cancel the installation. Because this window has a Cancel button, it is possible to exit the installation even though files may have already been copied to your computer.  
     
   **Note: *Do not follow this step unless you want to stop the installation of the openFLE*.**  
     
   If you click the **Cancel** button shown in Figure 2‑16, the window in Figure 2‑17 appears. Click **Yes** to stop installation and exit the installer.

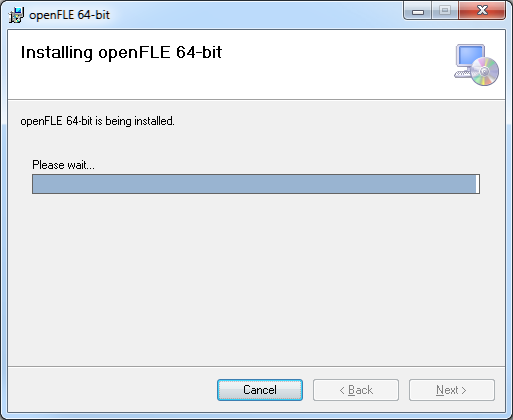


Figure ‑  
openFLE Installer: Installation Progress Window



Figure ‑  
openFLE Installer: Exit the Installer Confirmation

1. If the installation of the openFLE is canceled, the Installation Interrupted window is displayed (Figure 2‑18). Click **Close** to exit the openFLE installer, and disregard any further instructions in this manual.

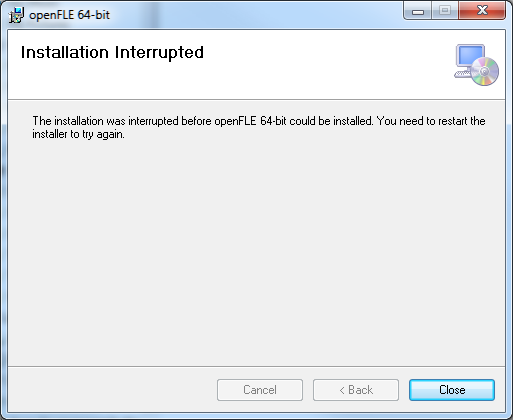


Figure ‑  
openFLE Installer: Installation Interrupted Window

1. If in step 19 you elected to let the installer continue, the Installation Complete window will be displayed (Figure 2‑19). Click on the **Close** button to finish the installation.
2. After the openFLE is installed, the openFLE service is started automatically and the openFLE Manager can be run by navigating to the installation folder and double clicking on the openFLEManager.exe (Figure 2‑20). See Chapter 3 for information on using the openFLE.

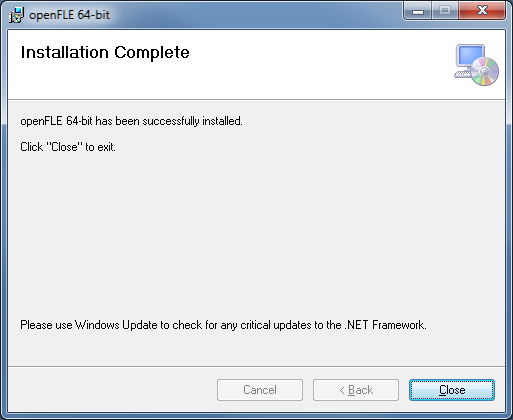


Figure ‑  
openFLE Installer: Installation Complete Window

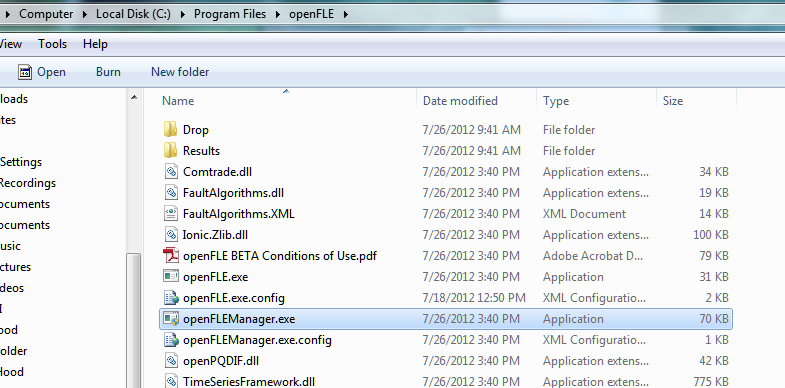


Figure ‑  
openFLEManager.exe (in Windows 7)

# Using openFLE

The openFLE is designed to run as a Windows service that automatically processes files as soon as they are placed in the “Drop” folder. It creates the corresponding output files in the “Results” folder.

## Tutorial

### Running the openFLE Manager

Navigate to the folder where openFLE is installed (see step 17 of the installation instructions), and double-click the **openFLEManager.exe** file name (Figure 3‑1).



Figure ‑  
openFLEManager.exe

**Note:** Based on your view settings in Windows Explorer or the version of Windows that you are using, this window may vary for you.

If you are running a Beta version of openFLE, a window that shows the terms and conditions of your Beta license agreement will appear first. To continue, you must click **Accep**t.

The openFLE Manager window will then be displayed (Figure 3‑2).

**Note:** Based on the version of openFLE that you are using, the graphic displayed on this window may vary for you.

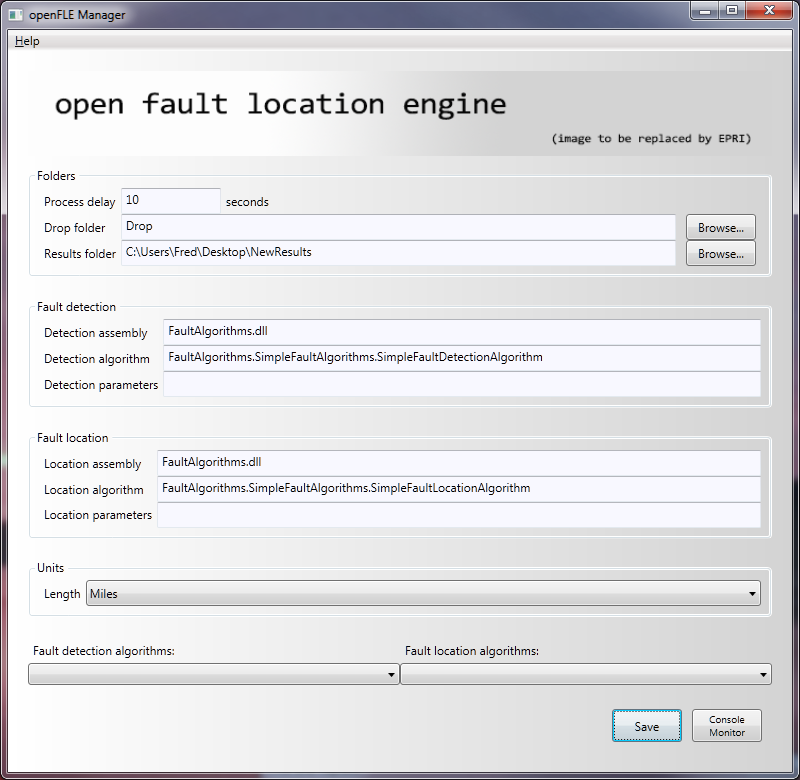


Figure ‑  
openFLE: The openFLE Manager window

### Input fields and Controls in the Manager Window

There are three input sections and seven control features, in addition to an About command on the Help menu that displays the software version and support information.

The inputs and controls are described below.

Folders

Input files to be processed must be placed in the **Drop** folder in groups, where an associated XML parameter file must accompany every event file to be processed. The type and number of input files varies by file format:

* For PQDIF input files, the group would consist of an XML parameter file, and a .pqd data file.
* For simple COMTRADE input files, the group would consist of an XML parameter file, a .cfg file and a single .dat data file.
* For larger COMTRADE input files, the group would consist of an XML parameter file, a .cfg file, and multiple .DXX files, where the .DXX files are named sequentially .D00, .D01, .D02, etc.

**Process delay** is an input field to specify the length of time in seconds to delay before processing files placed in the **Drop** folder that include .DXX files (Figure 3‑3). This delay only applies to COMTRADE files, when the data files include one or more .DXX files.

The **Process delay** input parameter is available to accommodate potential process delays that would cause .DXX files to arrive in the **Drop** folder with some amount of time between their arrivals. All other files are processed as soon as the input group is complete. For example, if an XML parameter file is placed in the **Drop** folder, it is ignored by the openFLE service until an associated data file arrives. File groups are identified by their names. Files placed in the **Drop** folder which are not recognized by openFLE will be ignored and left in the **Drop** folder.

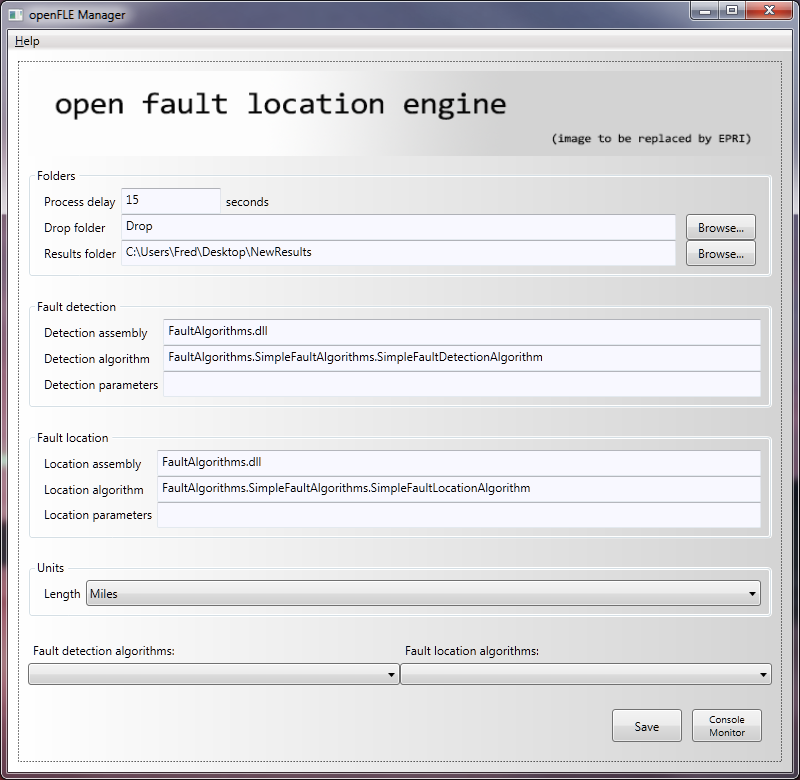


Figure ‑  
openFLE: Process Delay

By default a **Drop** folder and a **Results** folder are created in the installation directory. The respective names of the folders are **Drop** and **Results**. The folder names and locations can be changed as desired. If a name other than the default is used for a folder, the complete path must be entered in the respective **Drop folder** and **Results** **folder** input fields, either manually or by use of the associated **Browse…** button (Figure 3‑4).   
  
**Note:** The examples shown in Figure 3‑3 and Figure 3‑4 are for illustration only. Your folder path must match your folder structure.

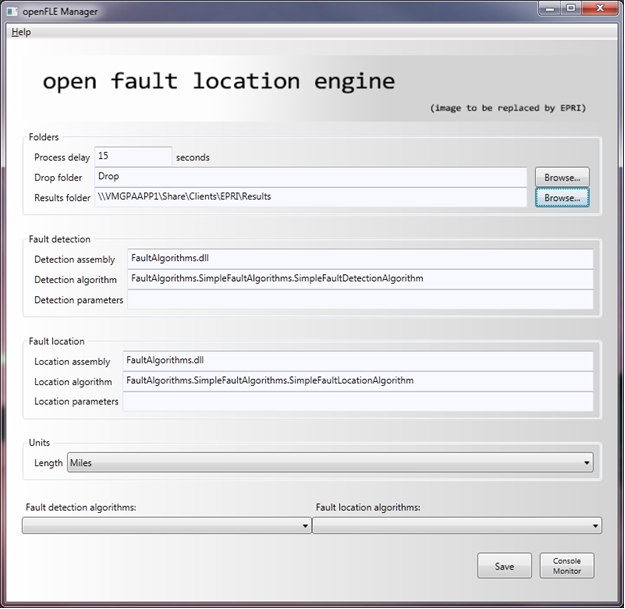


Figure ‑  
openFLE: The Manager window showing a modified Results folder path

Fault Detection

The OpenFLE is a platform where multiple detection assemblies, detection algorithms, and detection parameters can be employed. In this release, only one assembly and one algorithm are available, so the input fields **Detection assembly**, **Detection algorithm**, and **Detection parameters** should remain unchanged (Figure 3‑5).

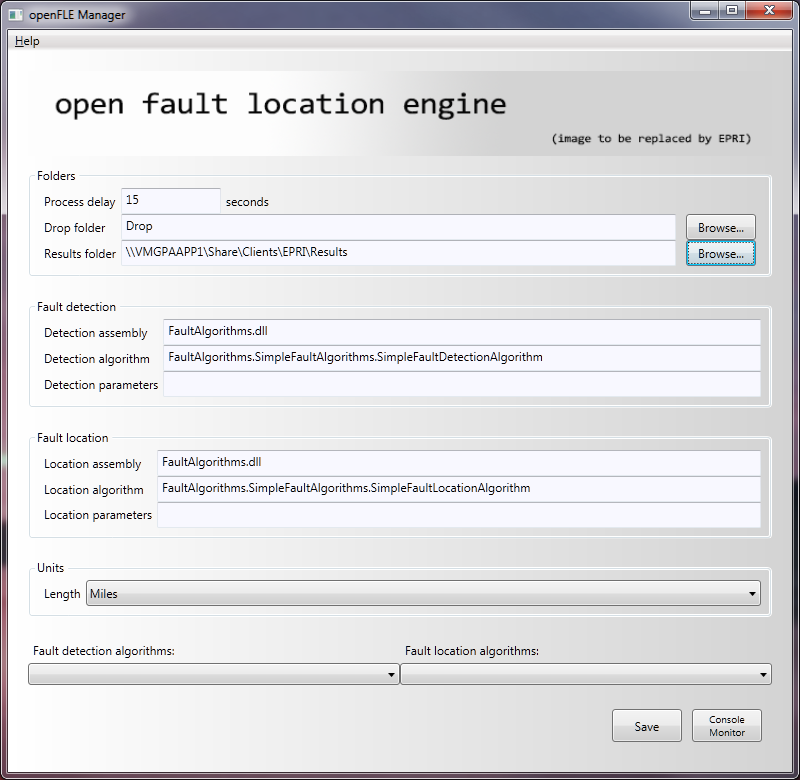


Figure ‑  
openFLE: The Manager window with default Fault detection input field contents

Fault Location

The OpenFLE is a platform where multiple location assemblies, location algorithms, and location parameters can be employed. In this release, only one assembly, and one algorithm are available, so the input fields **Location assembly**, **Location algorithm**, and **Location parameters** should remain unchanged (Figure 3‑6).

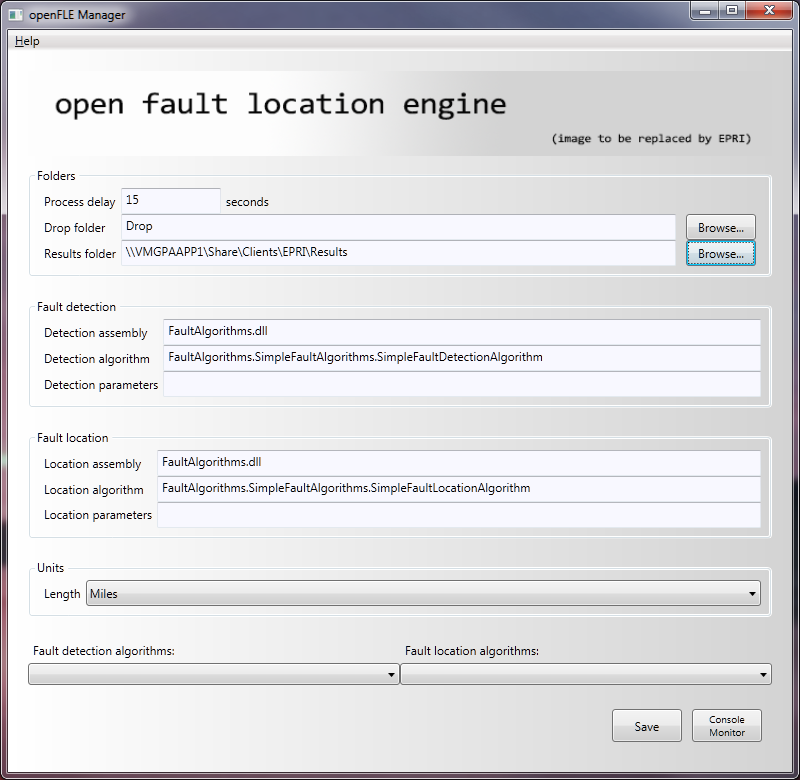


Figure ‑  
openFLE: The Manager window with default Fault location input field contents

Units

The unit of measure to describe the line lengths and the calculated distance to fault can be selected from a drop down list (Figure 3‑7). Miles are the installed default, but kilometers can be selected. If a change is made using this drop down list, the save button must be used to update the configuration information (Figure 3‑9).

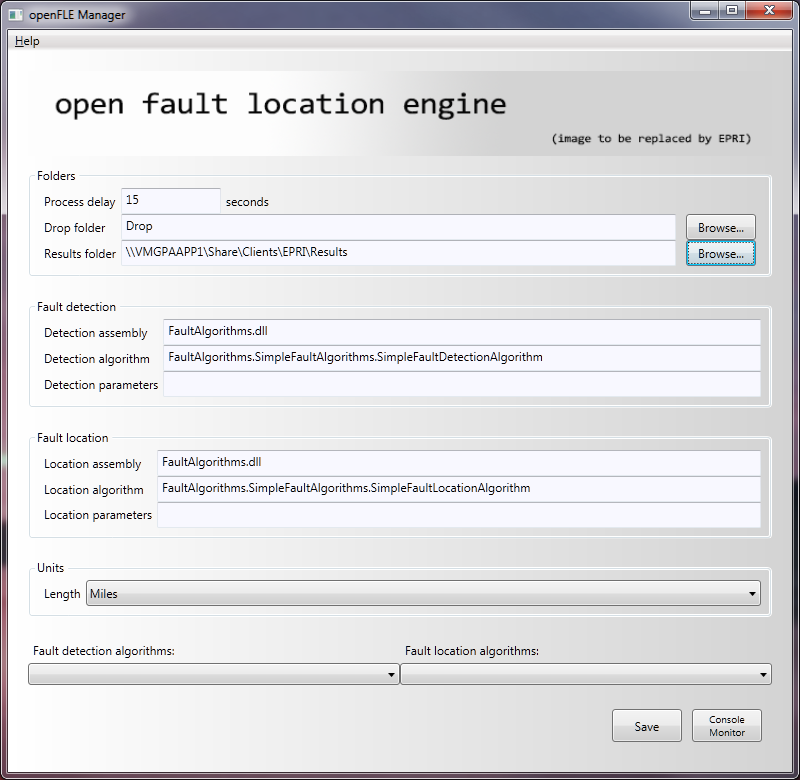


Figure ‑  
openFLE: The Manager window showing Units drop down list

Fault Detection and Fault Location Algorithm drop down selections

The drop down list selections called **Fault detection algorithms** and **Fault location algorithms** are intended to easily facilitate the selection of different options in each category (Figure 3‑8). In this project, only one selection is available in each category. The selections available can be used to reset to default, if the input fields are changed inappropriately.

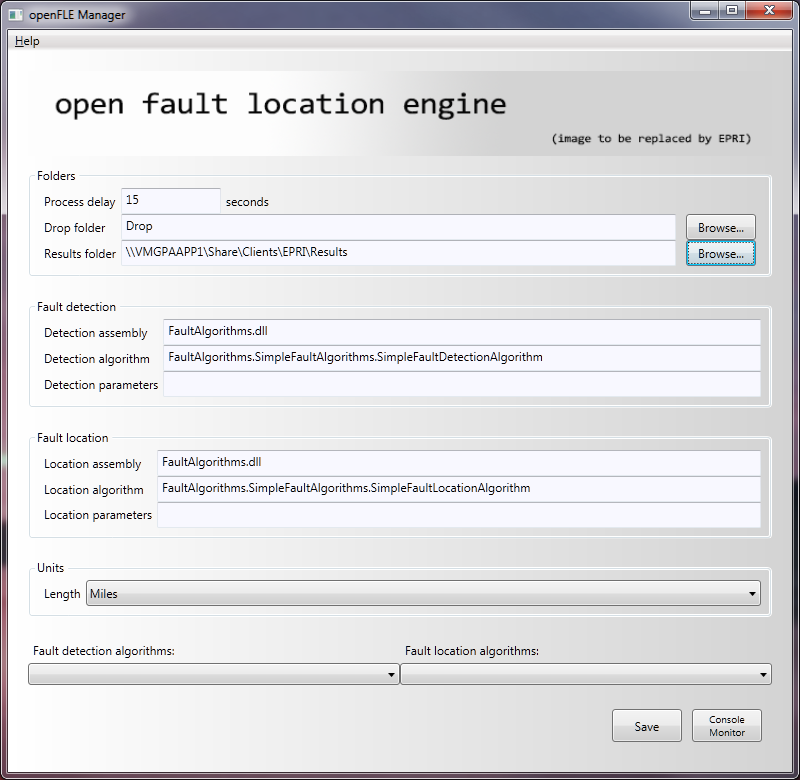


Figure ‑  
openFLE: Fault Detection and Fault Location algorithm drop down lists

Control Buttons

There are two control buttons; **Save** and **Console Monitor**. The **Save** button writes the configuration parameters as specified in the input fields, into the configuration file, which is located in the install directory. These saved settings will be used in subsequent executions of the program until new values are saved. A confirmation dialog box is displayed to indicate that the Configuration changes have been saved successfully. Click **OK** to dismiss the dialog box (Figure 3‑9).

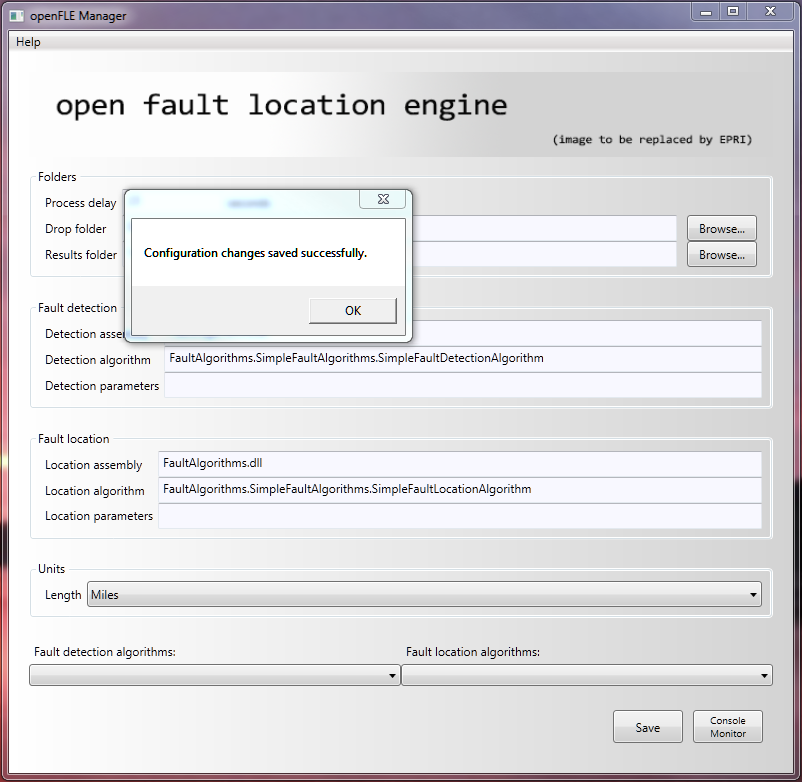


Figure ‑  
openFLE: Save Button and confirmation dialog box.

The **Console Monitor** button is used to open a window that displays commands as they are executed. When the openFLE Manager is started, the Console Monitor window is closed. The window opened by the **Console Monitor** button can be closed by clicking on the red **X** (Figure 3‑10).

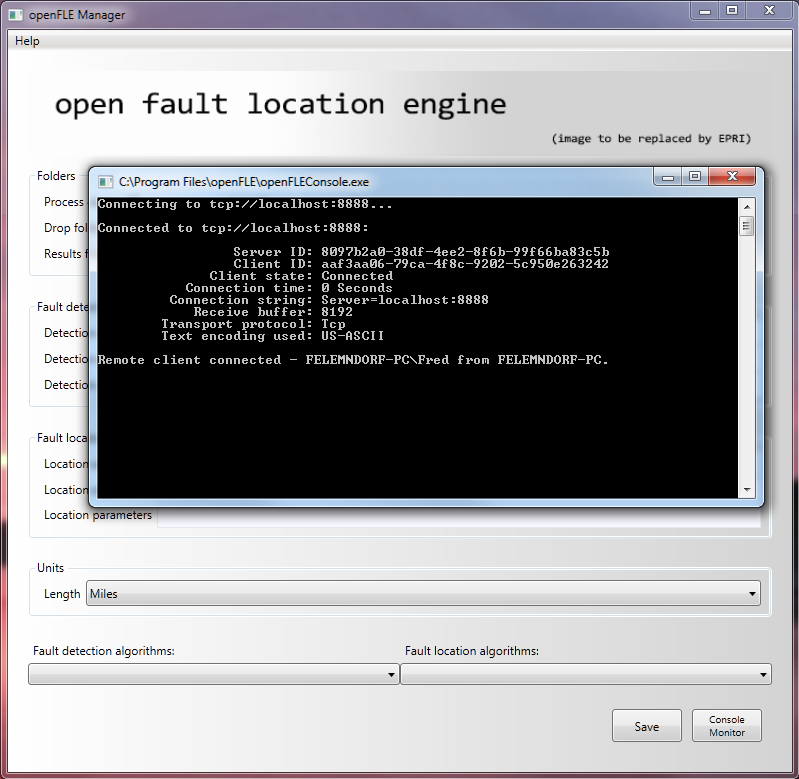


Figure ‑  
openFLE: Console Monitor Button, Console Monitor Window, Red X to close.

Closing the openFLE Manager

When no further configuration changes are desired, the openFLE Manager can be closed by clicking on the red **X** (Figure 3-13).

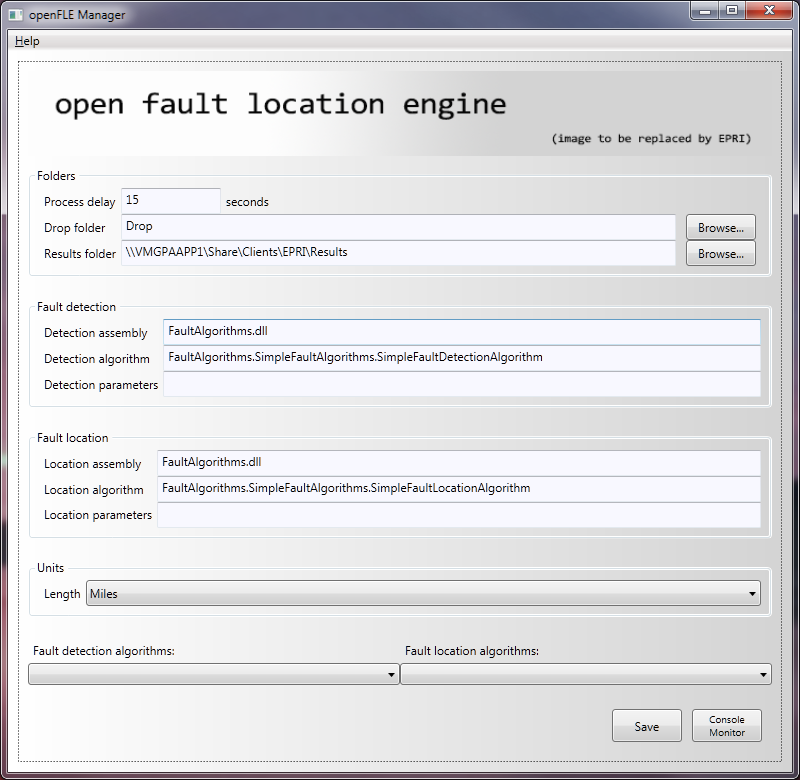


Figure ‑  
openFLE: openFLE Manager Red X to close.

Stopping the openFLE Service

The openFLE service is started automatically, and runs continuously until it is stopped manually (Figure 3‑12).

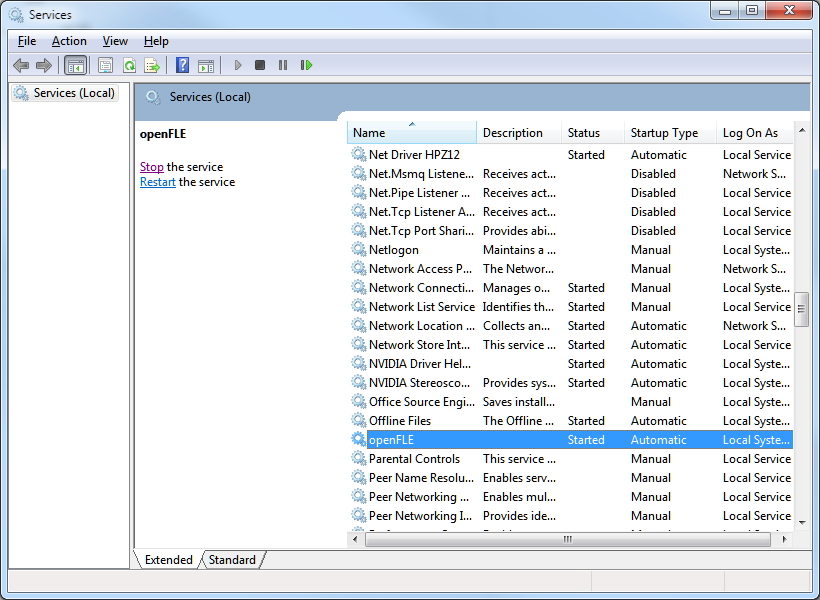


Figure ‑  
openFLE: Stopping the openFLE service

Perform Fault Location

To perform a fault location on an input event file, the input data file and the configuration file associated with the faulted line must be placed in the **Drop** folder specified in the openFLE Manager (Figure 3-5). For this project, an .xml file is made available for each input event file. Refer to Figure 3‑13 for an example of a complete input file group for an event numbered 5043. In this example, the data is contained in a simple COMTRADE group.

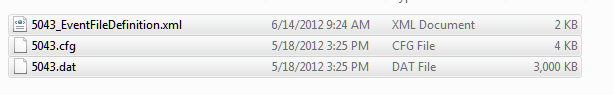


Figure ‑  
openFLE: Input COMTRADE and xml files for Event 5043

Once the complete input file group is in the **Drop** folder, the process begins automatically. When the openFLE successfully processes the input data file, all input files are moved to the **Results** folder with the calculated results files (Figure 3‑14).

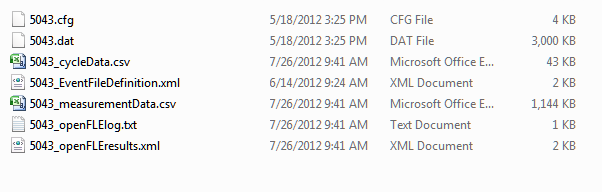
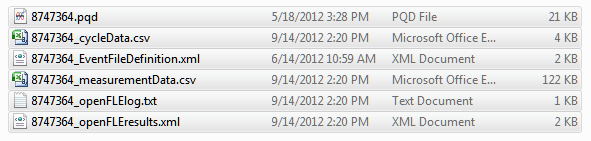


Figure ‑  
openFLE: Contents of Results folder after event 5043 processed

An example of the input files for a PQDIF input data file, and the resulting entries in the output folder are shown in Figure 3‑15.



Input files placed in **Drop** folder.



Output files in **Results** folder.

Figure ‑  
openFLE: PQDIF input file example.

###### Example of input xml file

Event File Definition - An Event File Definition (XML) File

This is the event file definition for an event called “Example.” An event file definition contains system parameters and their values.



Figure ‑1  
Example\_EventFileDefinition.xml

###### Examples of output files in results folder

* 1. Log File - A Log (TXT) File

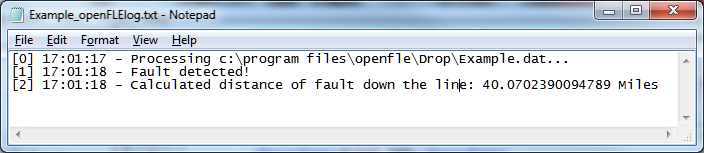


Figure ‑4  
Example\_openFLElog.txt

Cycle Data - A Portion of a Cycle Data (CSV) File

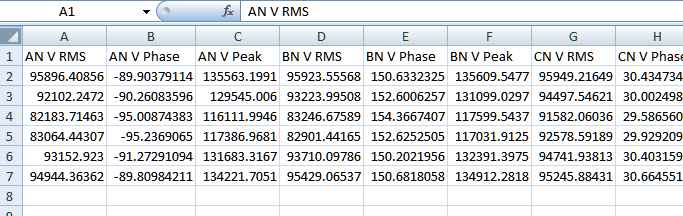


Figure B‑1  
Example\_cycleData.csv

* 1. Measurement Data - A Portion of a Measurement Data (CSV) File

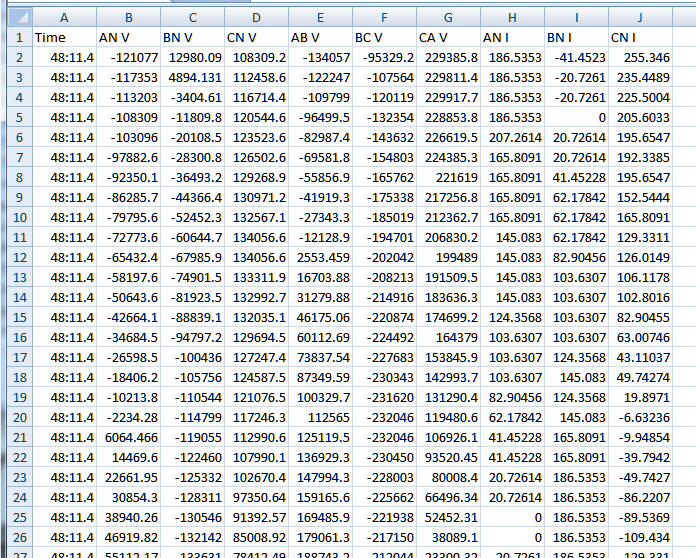


Figure ‑2  
Example\_measurementData.csv

* 1. Results File – A Results (XML) File



Figure ‑5  
Example\_openFLEresults.xml

Other Files

Other files that will be listed in the output folder are COMTRADE and PQDIF files that have been moved from the input (“Drop”) folder after processing.

|  |  |
| --- | --- |
| **WARNING:** This Document contains information classified under U.S. Export Control regulations as restricted from export outside the United States. You are under an obligation to ensure that you have a legal right to obtain access to this information and to ensure that you obtain an export license prior to any re-export of this information. Special restrictions apply to access by anyone that is not a United States citizen or a permanent United States resident. For further information regarding your obligations, please see the information contained below in the section titled “Export Control Restrictions.”  Export Control Restrictions  Access to and use of EPRI Intellectual Property is granted with the specific understanding and requirement that responsibility for ensuring full compliance with all applicable U.S. and foreign export laws and regulations is being undertaken by you and your company. This includes an obligation to ensure that any individual receiving access hereunder who is not a U.S. citizen or permanent U.S. resident is permitted access under applicable U.S. and foreign export laws and regulations. In the event you are uncertain whether you or your company may lawfully obtain access to this EPRI Intellectual Property, you acknowledge that it is your obligation to consult with your company’s legal counsel to determine whether this access is lawful. Although EPRI may make available on a case-by-case basis an informal assessment of the applicable U.S. export classification for specific EPRI Intellectual Property, you and your company acknowledge that this assessment is solely for informational purposes and not for reliance purposes. You and your company acknowledge that it is still the obligation of you and your company to make your own assessment of the applicable U.S. export classification and ensure compliance accordingly. You and your company understand and acknowledge your obligations to make a prompt report to EPRI and the appropriate authorities regarding any access to or use of EPRI Intellectual Property hereunder that may be in violation of applicable U.S. or foreign export laws or regulations. | **The Electric Power Research Institute Inc.,** (EPRI, www.epri.com) conducts research and development relating to the generation, delivery and use of electricity for the benefit of the public. An independent, nonprofit organization, EPRI brings together its scientists and engineers as well as experts from academia and industry to help address challenges in electricity, including reliability, efficiency, health, safety and the environment. EPRI also provides technology, policy and economic analyses to drive long-range research and development planning, and supports research in emerging technologies. EPRI’s members represent more than 90 percent of the electricity generated and delivered in the United States, and international participation extends to 40 countries. EPRI’s principal offices and laboratories are located in Palo Alto, Calif.; Charlotte, N.C.; Knoxville, Tenn.; and Lenox, Mass.  Together…Shaping the Future of Electricity |
| © 2012 Electric Power Research Institute (EPRI), Inc. All rights reserved. Electric Power Research Institute, EPRI, and TOGETHER…SHAPING THE FUTURE OF ELECTRICITY are registered service marks of the Electric Power Research Institute, Inc. | 1024887 |