



ACE Operation Manual



Elecsys Director

ACE Operation Manual

Product Information

Full information about other Elecsys products is available on our website at www.elecsyscorp.com

Useful Contact Information

Customer Support

Tel: +1-913-982-5672

Fax: +1-913-982-5766

Email: support@elecsysscada.com

Headquarters, Sales, Support & Manufacturing

Elecsys Corporation

846 N Mart-Way Court

Olathe, KS 66061

Tel: 913-982-5672

Fax: 913-982-5766

Email: sales@elecsysscada.com

While the Elecsys sales team may assist customers in choosing a product for a specific application, the final choice of product is entirely the responsibility of the buyer. Elecsys entire liability with respect to its products or systems is set out in the Elecsys standard terms and conditions of sale.

All example code is provided only to illustrate the use of Elecsys products. No warranty, either expressed or implied, is made regarding any example code provided by Elecsys and Elecsys shall incur no liability whatsoever arising from any use made of this code.

© 2014 Elecsys Corporation

Contents

1	Revision History	5
2	Preface	6
2.1	Scope of this Manual	6
2.2	Name Conventions	6
2.3	Disclaimer	6
3	Introduction	7
3.1	ACE Product Overview	7
3.1.1	Configurations and Templates	8
3.1.2	ACE Program File Locations	10
3.1.3	Using ACE Configuration Files	12
3.2	How To Use This Manual	12
4	Installation	14
4.1	ACE Installation Files	14
4.1.1	Install ACE	14
4.2	Install the ACE Repository Service	17
4.2.1	Starting and Stopping Repository Service	17
4.2.2	Configure Windows Firewall for Repository	19
4.3	Uninstalling ACE Software	19
5	Setting Up ACE For First Use	20
5.1	Install ACE Icon Library	20
5.2	Setting Up Offline Working Directory	21
5.3	Set Up ACE Repository Server	21
5.3.1	Connect to Repository	22
5.3.2	Create User Account	23
5.4	Connect User Workstation to ACE Repository	24
5.4.1	Connect to Repository	24
5.4.2	Select or Create User Account	26
5.5	Setting Up Templates and Configurations	27
5.5.1	Restore Existing XML Files (non-repository mode)	27
5.5.2	Restore Existing XML Files (repository mode)	28
5.5.3	Restore Backup Repository Files	29
5.5.4	Import non-XML files from old ACE	29
6	ACE Configuration Editor	31
6.1	Using the ACE Configuration Editor	31
6.1.1	Tool Bar	31
6.1.2	ACE Settings	32
6.1.3	Changing the Local Working Directory	33
6.2	Working with Configuration Files	34
6.2.1	Opening a Configuration	34

6.2.2	Creating a New Configuration.....	36
6.2.3	Saving a Configuration	38
6.2.4	Importing a Legacy Configuration	38
6.3	Editing a Configuration	40
6.3.1	Configuration Properties	41
6.3.2	Object Properties.....	42
6.3.3	Modifying Object Field Values	44
6.3.4	Edit Table	45
6.3.5	Moving, Copying and Pasting Nodes and Branches	46
7	ACE Template Editor	47
7.1	Using the ACE Template Editor.....	47
7.1.1	Tool Bar.....	47
7.1.1	Changing the Template Working Directory	47
7.2	Working with Configuration Files	48
7.2.1	Opening the Default Template.....	48
7.2.2	Opening an Existing Template.....	48
7.2.3	Saving a Template.....	49
7.2.4	Importing a Legacy Template	50
7.3	Editing a Template	51
7.3.1	Add/Delete Child Object	51
7.3.2	Edit Object Properties.....	53
7.3.3	Edit Fields and Field Properties.....	54
7.3.4	Edit Options List	56
7.3.5	Edit Table Properties	57
7.4	Resolving Template Problems.....	58
8	Using ACE in Repository Mode	59
8.1	Version History.....	59
8.2	Open File from Repository.....	59
8.3	Check In / Check Out From Repository	60
8.4	Saving Repository Files.....	61
8.5	Switch to a Different User or Repository	63
9	Deploying ACE Configurations	64
9.1	Loading Single Director Configuration	64
9.1.1	Network Upload via ACE Editor.....	64
9.1.2	Serial Upload via YMODEM	67
9.1.3	Using the ACE Uploader for Multiple Directors	68
9.2	Using HCP Configuration	69
9.2.1	Compile for HCP	69

1 Revision History

Revision	Date	Comments
Revision 1.1.3	5/15/2013	New release of ACE Operation, for XML version of ACE
Revision 1.1.4	3/27/2014	Updates for new ACE Editor with single installation menu and various changes in software release 3.0.0.

2 Preface

2.1 Scope of this Manual

This manual provides an overview of the features and applications of the Elecsys ACE software with the Director, and instructions to the first-time user to get started using the product.

2.2 Name Conventions

All numbers are in decimal unless otherwise indicated. Where a number is prefixed by '0x', the value is in hexadecimal format.

2.3 Disclaimer

The information in this manual has been carefully checked and is believed to be accurate. Elecsys Corporation assumes no responsibility for any infringements of patents or other rights of third parties that may result from its use.

Elecsys Corporation assumes no responsibility for any inaccuracies that may be contained in this document. Elecsys makes no commitment to update or keep current the information contained in this manual.

Elecsys Corporation reserves the right to make improvements to this document and/or product at any time and without notice.

3 Introduction

The Advanced Configuration Environment (ACE) is a Windows-based application used to configure the Elecsys line of Remote Data Communications SCADA products, including the Elecsys Director and HCP, and are also used as part of the OPC Appliance.

This manual provides the information that a user will need to install the ACE software, understand the structure of configuration and template files, manage local and remote configuration directories, and transfer configuration files to a device.

3.1 ACE Product Overview

The original versions of ACE operated in a stand-alone, single-user mode of operation, and used property-based template and configuration files to store the contents of each configuration. These will be referred to in this manual as “old ACE” configurations and templates.

In 2012, the program was completely rewritten and renamed to Advanced Configuration Environment (ACE). In this new release of ACE, configurations and templates are stored in an XML data structure. The new versions will sometimes be referred to in this manual as “new ACE,” or simply ACE, and refer to releases beginning with version 3.0.0. Some new features of ACE include a template editing feature and the ability to utilize a central file repository.

The repository is a file version control system that can be accessed by multiple users, and stores an archived version history of all configuration and template files. Files must be “checked out” from the repository before they can be modified, which prevents other users from accidentally editing the file at the same time. Only one user can edit and save changes to any given configuration file at a time, but other users can view the files in read-only mode..

There are several programs associated with the ACE editor software:

- ACE Configuration Editor – Primary application for viewing and modifying the properties of device configurations.
- ACE Template Editor – Utility that allows a system designer to modify the configuration objects and templates used by the ACE Configuration Editor.
- ACE Batch Uploader – Utility to upload configurations to multiple Directors in a batch process.
- ACE Repository Service – Windows service that runs in the background on a shared server computer and manages a repository of configuration directories, allowing centralized multi-user access to configurations.

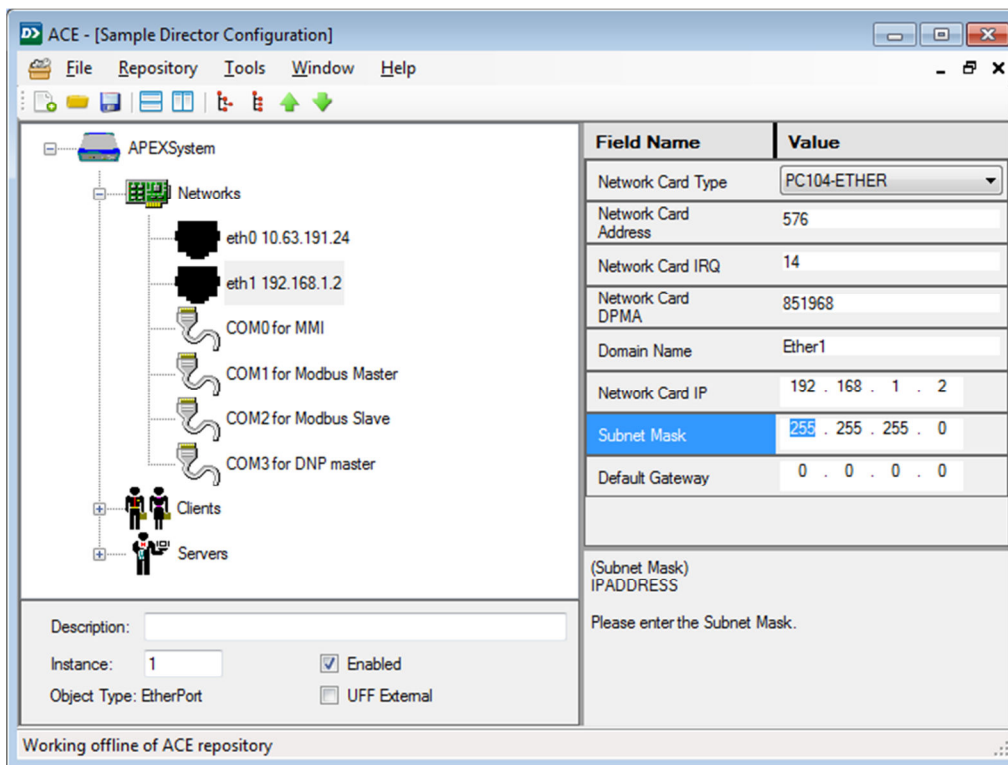
3.1.1 Configurations and Templates

An ACE configuration relies on two files: a configuration file and a template file. Both files are stored to media in XML file format. A single configuration file is used to configure a single device, typically an Elecsys Director device.

A configuration file contains a multitude of configurable properties, each of which controls a specific element or action within the device. Each configurable property is defined as a specific data type: integer, floating point number, string, etc. The configurable properties are grouped into sets called configuration objects. A single configuration file is composed of many configuration objects organized into a hierarchical structure.

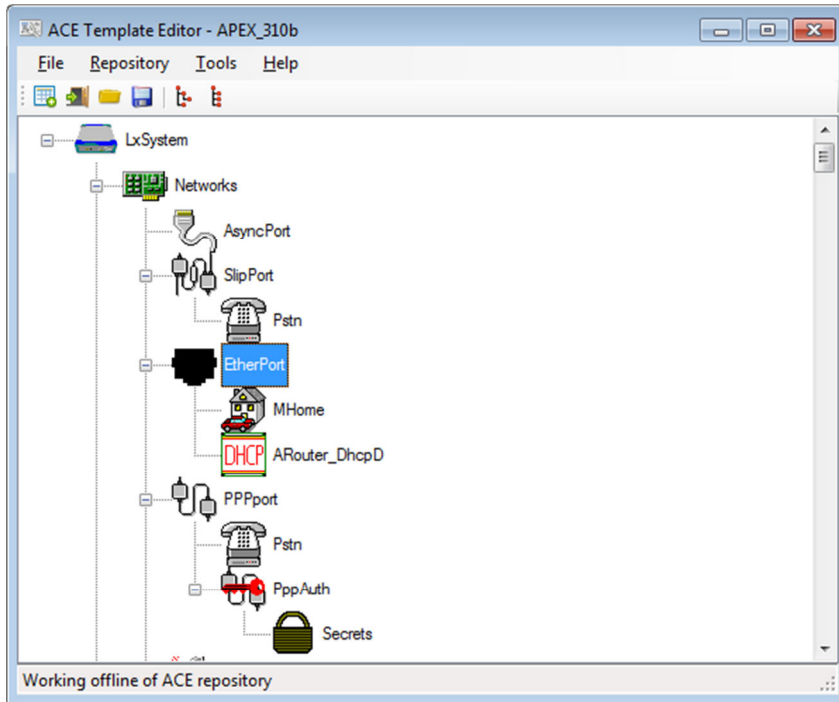
Each configuration file references a single template file. The template is typically used by many different configurations and defines the hierarchical organization of the configuration objects, as well as the structure, rules, and definition of the configurable properties within the configuration.

The ACE Configuration Editor is used to create, view, and edit ACE configuration files. The typical end user will only ever use the ACE Configuration Editor. The following screen capture illustrates the hierarchical organization of configuration objects in a typical ACE configuration file.



The ACE Template Editor is used to create, view, and edit ACE template files. The ACE Template Editor will only be used occasionally, since the structure and rules of a template rarely change.

The following screen capture illustrates the template file referenced by the configuration. This shows the relationship between the organization of the configuration file and the rules within the template. The template file defines what types of fields are stored in a configuration and what values those properties can hold, whereas the configuration files hold specific values for each of the configurable properties for an installed device.



In the previous two screen captures, the highlighted icon is the Ethernet Port configuration object (type "EtherPort"). Notice that the template defines the organization and definition of the configurable fields within the object, where the Ethernet object is one possible child of the "Networks" object. In the ACE configuration file, the highlighted node represents the configuration of a specific physical Ethernet adapter, "eth1" in this case. The ACE Configuration Editor displays the actual values assigned to each of the fields in the configuration object.

Refer to the *Director Configuration Manual* or *HCP Configuration Manual* for a list of available configuration elements, the meaning and importance of their property values, and additional information to help the system designer configure a system to function as intended.

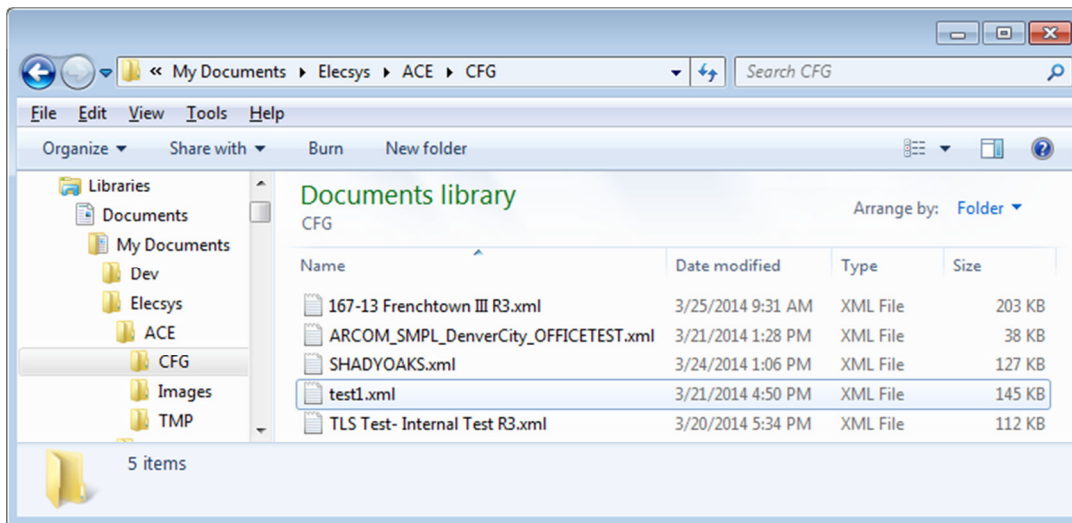
3.1.2 ACE Program File Locations

The ACE editors save files to the local working directory, and to the ACE repository if the user is working in online mode.

The local working directory holds the icons, configuration files, and template files that have been edited locally or checked out to local file storage media. When used in the offline mode of operation, the local working directory is the only location for file storage. The offline mode would be used if the configuration files are not shared among users or if the repository option is not practicable. Any synchronization or backups of the configuration files and templates used in offline mode would need to be managed by the user. Offline file storage is also useful in cases where ACE configuration files are checked out from a repository and stored on a laptop computer that is outside the reach of a corporate network; for example, a technician working at a field location.

The local working directory defaults to the user's "My Documents" folder, although its location can be changed, and includes the following directories:

- Elecsys\ACE\CFG – configuration files in local working directory
- Elecsys\ACE\Images – icon files in the local working directory
- Elecsys\ACE\TMP – template files in local working directory



The repository feature of ACE provides the ability to read and write configuration files in a shared file server. The centralized repository can more easily be maintained across an organization and backed up, rather than having multiple uncontrolled copies on different user workstations.

The ACE Repository can be installed on any Windows computer within an organization. It should be installed on a computer that is accessible by all users who wish to use ACE. The ACE Repository is installed as a Windows service and does not have a user interface. All access to the repository features is accomplished through the ACE editor tools. Backup of the repository files should be managed by the server administrator.

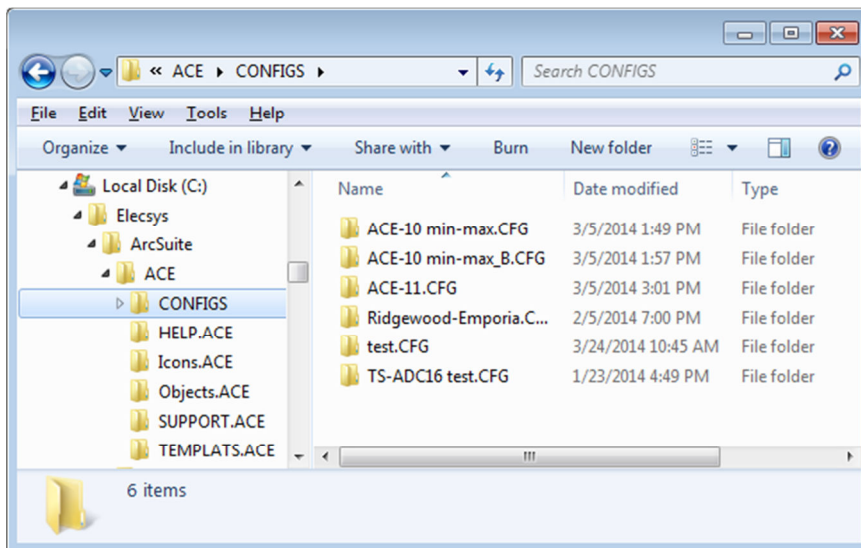
The ACE Repository files are stored on the server computer in the following locations:

- C:\ProgramData\Elecsys\AceRepository – location of repository on Windows 7
- C:\Documents and Settings\All Users\Application Data\Elecsys\AceRepository – location of repository on Windows XP

When the ACE Configuration Editor or Template Editor opens a file from the ACE repository, a cached copy of that version of the file is stored to the user's local working directory. This cached copy is useful in case the computer is disconnected from the repository. This allows a technician to work with the configurations even when disconnected from the network or remote repository, but then reconnect to the repository and check in changes.

This manual also makes occasional references to the “old ACE” (non-XML) configuration directories, which may be imported into the current XML version of ACE. The older ACE program typically installed both the program and configuration files to C:\Elecsys\ArcSuite\ACE or C:\Program Files\ArcSuite\ACE, although the default location could be changed. Within the “old ACE” file structure, the following directories were used:

- ACE\Configs – directories containing configurations and download files
- ACE\Configs\configuration_name.CFG\Download – individual binary download files
- ACE\Configs\configuration_name.CFG\UFF – combined binary download files
- ACE\Icons.ACE – icon file library
- ACE\Objects.ACE – object file library
- ACE\Templats.ACE – template file library



3.1.3 Using ACE Configuration Files

Configurations created using ACE store their property values in XML files within the ACE environment (repository and/or local working directory). In order to configure a system for operation, ACE creates one or more binary files containing the essential property values, which must be loaded into a device or used by a SCADA system.

For the Director products, the configuration's property values are compiled into one or more binary files that are downloaded to the Director. Most or all of the properties will be stored in a file with a file extension of .UFF, although some configuration elements compile their properties in a separate binary file. Once the configuration has been loaded and the device is restarted, the configuration is read from these new file(s), and the device operates according to the new configuration.

The Elecsys HCP (Host Communication Processor) requires for the binary download files for all its connected Directors, to determine their operational characteristics. The HCP expects the download files to be in the same locations as the old ACE program (Download and UFF folders in "Configs*configuration_name*.CFG"). There is an option in the ACE Configuration Editor to save files locally for the HCP using this file structure.

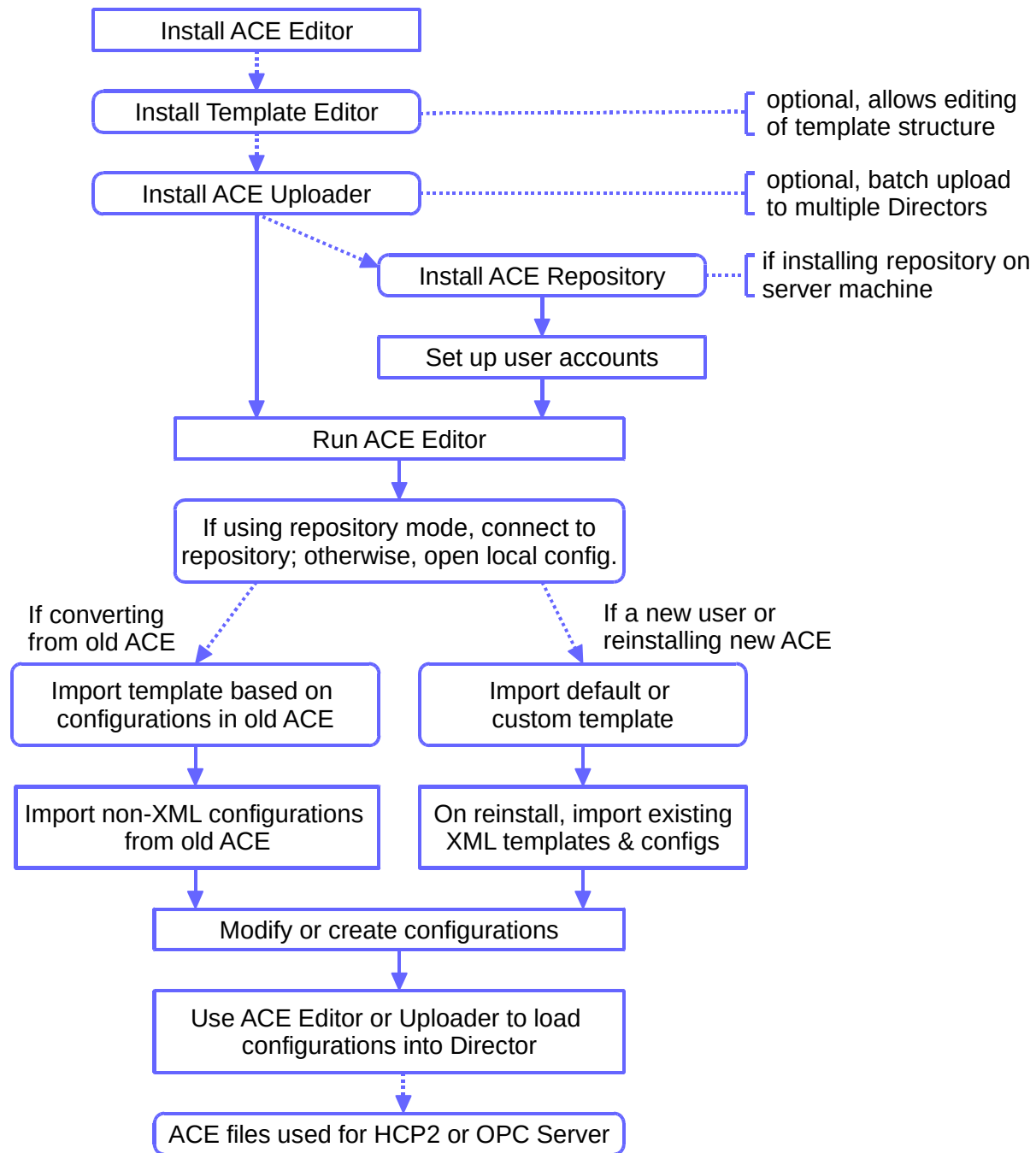
The Elecsys OPC Server requires that the Director binary UFF files be uploaded via FTP, to determine their operational characteristics.

3.2 How To Use This Manual

This manual describes the steps for installing the ACE utilities and managing configurations and templates. Depending on a user's needs and intended use of the ACE program, different sections of this manual will apply. Some use cases are described here:

- You previously used the old (non-XML) version of ACE and now need to convert configurations to the new XML version of ACE.
- Your company is using ACE in repository mode, and you need to check out, modify, and/or upload configurations to a Director.
- You need to set up an ACE repository server for first time use.
- You need to install and use ACE locally (stand-alone), not connected to a repository.
- You have already used the new XML version of ACE, but now need to reinstall the program along with its templates and configurations.

The following diagram illustrates the main steps for using ACE, with options depending on the needs of a given installation.



4 Installation

This section gives information on installing and setting up each of the ACE component programs. There is a single installation program that installs both the ACE editor tools and the repository.

4.1 ACE Installation Files

Download the ACE installation files from the following link:

<http://ftp.elecsyscorp.com/ACE/NewACE/AceInstallDisk.zip>

This Zip archive should contain two files:

- setup.exe
- AceSetup.msi

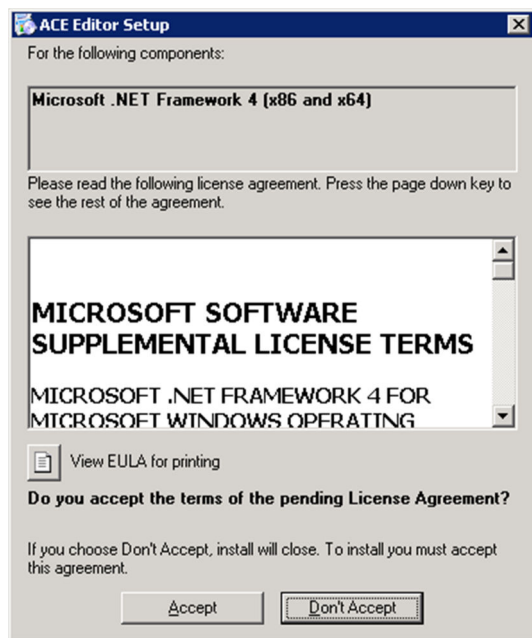
NOTE: The ACE application programs require the Microsoft .NET Framework version 4.0 to be installed on the computer. Microsoft .NET will be installed by ACE if not already present on the computer. The ACE tools should run properly on any Windows computer that is capable of running the Microsoft .NET Framework; essentially any computer with Windows XP service pack 3, or later.

After installing any of the ACE tools, it is advised that Windows Update be used to check for any Microsoft security patches for this .NET Framework.

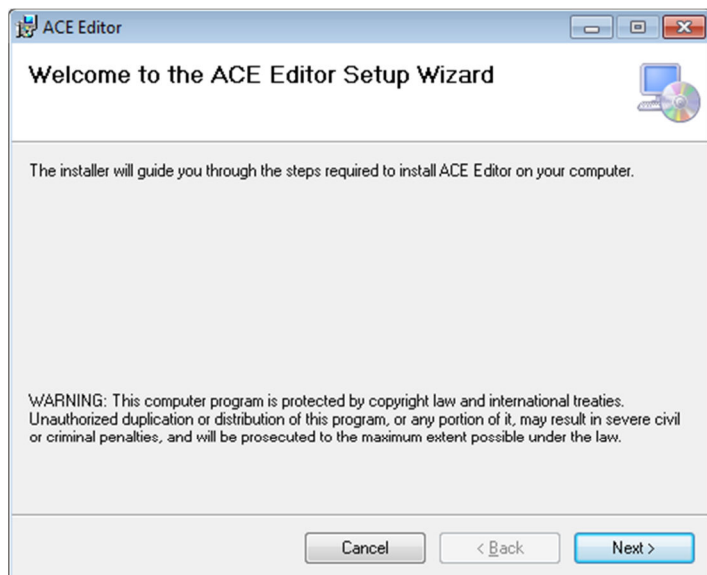
4.1.1 Install ACE

Extract the contents of the ACE installation Zip file to a temporary location and double-click the **setup.exe** program.

If the Microsoft .NET Framework is not already installed on the computer, the following prompt will be displayed. Click the **Accept** button to commence the process of downloading and installing the Microsoft .NET Framework files.



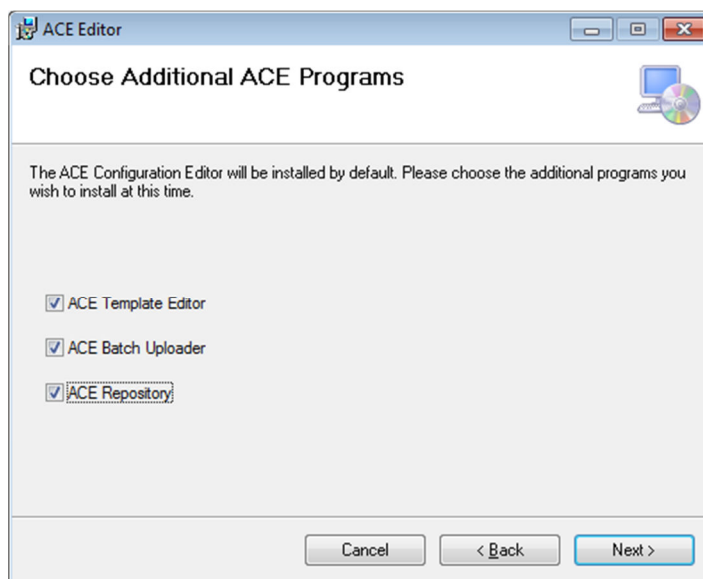
After the installer has verified that the Microsoft .NET Frame is properly installed, you will see a Welcome message. Click the **Next** button to continue.



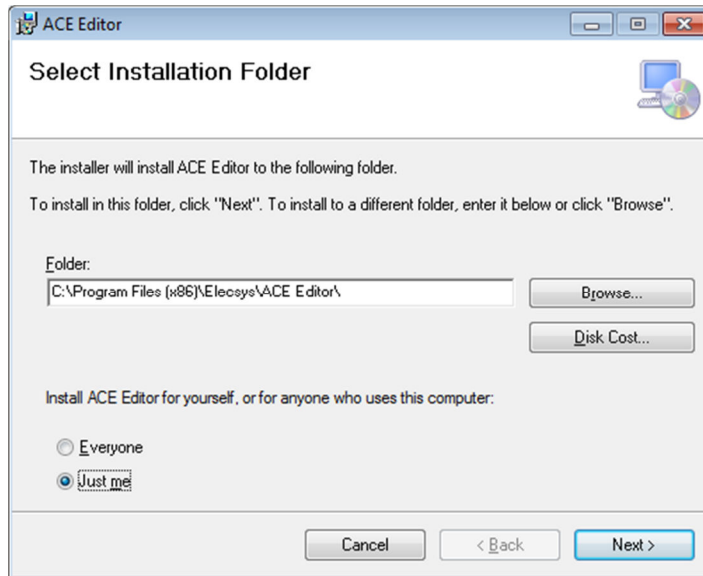
You will be prompted to indicate which optional ACE tools to install. The ACE Configuration Editor will always be installed. Check the checkbox next to the other ACE tools that you want to install, then click the **Next** button.

- Select the **ACE Template Editor** to allow modification of template files.
- Select the **ACE Batch Uploader** to allow batch upload of files from this computer to multiple Directors
- Select **ACE Repository** if you are installing on the repository server.

NOTE: You should only install the ACE Repository on one computer that can be accessed by users, not on each individual user workstation. Stand-alone installations also do not require the ACE repository to be installed.



Select the folder in which you wish to install the ACE Configuration Editor. Select the “*Everyone*” or “*Just me*” installation option to put the shortcut into the global or user-specific Windows Start menu. Then click **Next** to continue, and **Next** again to install the software.



After successful installation, click the **Close** button.

The ACE Configuration Editor, along with ACE Template Editor and ACE Uploader, if installed, will appear in the Windows Start menu in the “Elecsys” program group. The ACE repository service is listed in the Windows Services dialog. See [ACE Settings](#) on page 32 for instructions on setting up the main ACE Configuration Editor options.

4.2 Install the ACE Repository Service

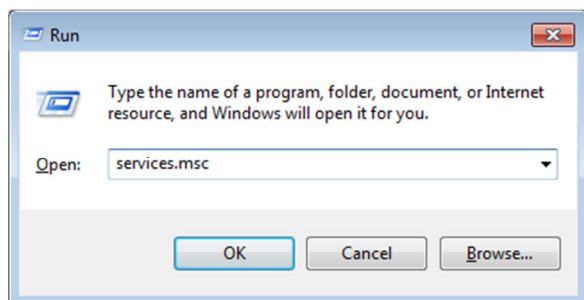
The ACE Repository service typically needs to be installed only once on a central server and then not touched again. This computer should be network-accessible by every computer hosting a copy of the ACE editor. The ACE Repository service **does not** need to be installed on every computer that has a copy of the ACE Configuration Editor or ACE Template Editor.

Ensure that the ACE Repository option is checked when running the ACE installer on the server machine.

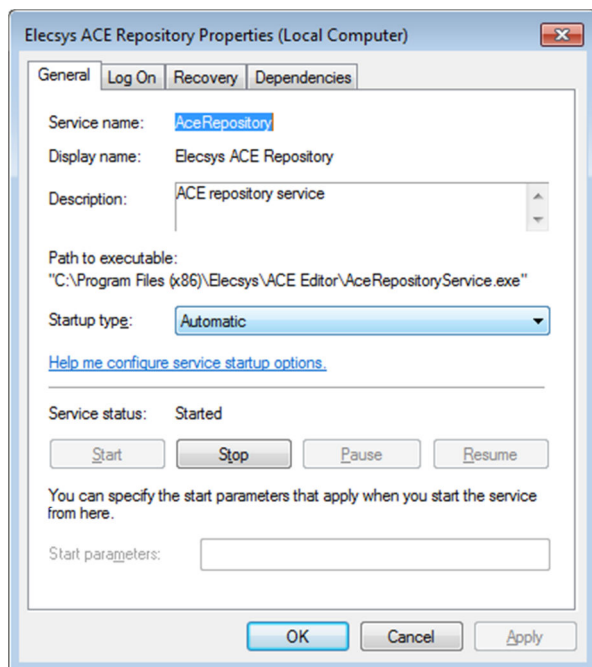
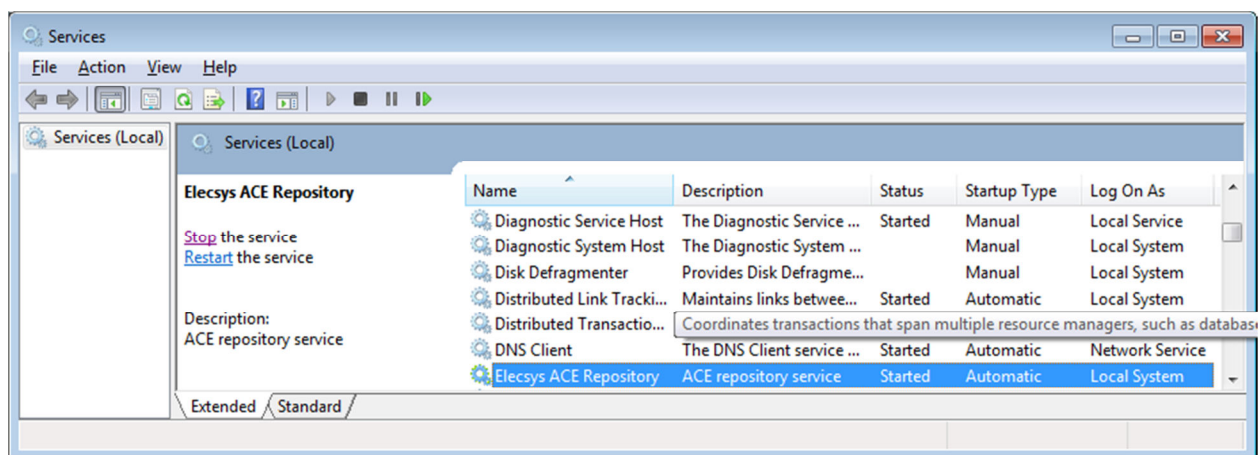
4.2.1 Starting and Stopping Repository Service

The ACE Repository is installed as a Windows service. The service starts automatically after the installation has completed, and is configured to automatically start whenever the computer boots.

If you need to start or stop the service, select the Services option from the Windows Start menu, or select Run and enter “services.msc”.



Locate the Elecsys ACE Repository service. The service may be started or stopped, and set to Automatic or Manual mode (Automatic is default).



4.2.2 Configure Windows Firewall for Repository

The ACE Configuration Editor and Template Editor communicate with the ACE repository program by using HTTP web service requests on TCP/IP port 8000. In order to access the ACE Repository from an ACE editor, it may be necessary to add an exception in the server's Firewall configuration for an inbound connection on TCP port 8000. For help on changing the firewall settings, consult a network administrator or the documentation on the firewall software in use on the server.

4.3 Uninstalling ACE Software

To uninstall ACE, open the Windows Control Panel and choose "Programs and Features" or "Add/Remove Programs", depending on the version of Windows, and uninstall the ACE Editor program.

Note that uninstalling or reinstalling the ACE Editor and ACE Repository does not remove the configurations and templates stored in the local working directories or repository.

5 Setting Up ACE For First Use

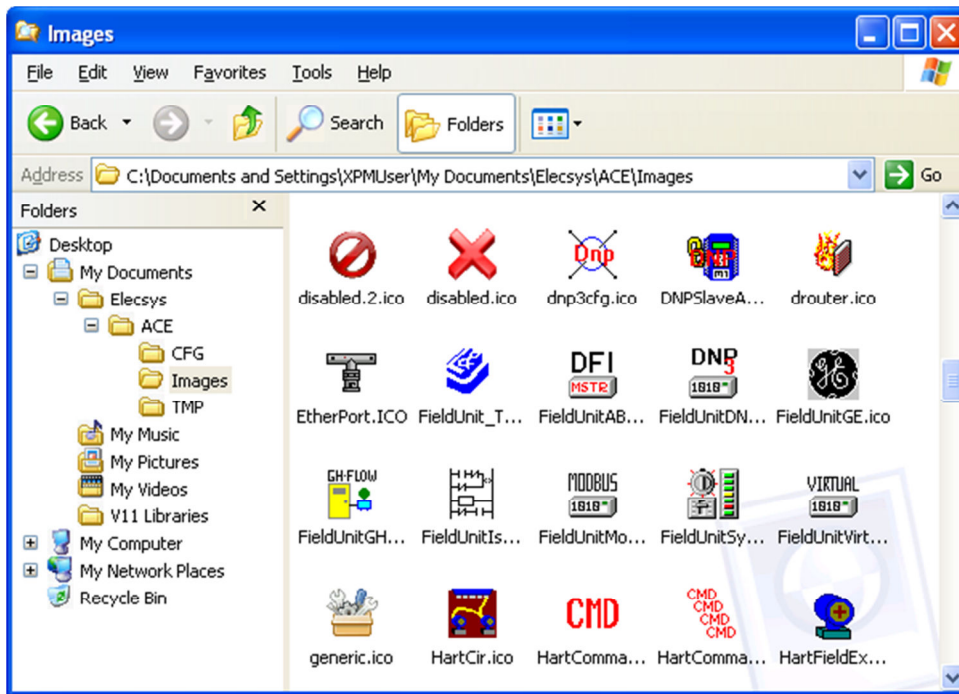
After installing ACE, it is necessary to set it up for use. This will include obtaining a library of icon files, configuring one or more repository users, and importing or copying in configurations and templates.

5.1 Install ACE Icon Library

Note: This section applies to all users.

The ACE Configuration Editor and Template Editor programs require a user-installed library of icon files. These icon files are used to present graphical images for the various configuration objects within an ACE configuration file.

Once installed, the icons are stored in the ACE local working directory on each user's workstation (such as "My Documents\Elecsys\ACE\Images"), not in the central repository.



If you have an existing ACE installation, you can manually copy the icons into the "ACE\Images" folder or add them to a Zip file that will be used during the initial ACE setup. Otherwise, the ACE program installer provides a default set of icons in the program installation directory ("Program Files (x86)\Elecsys\ACE Editor\Default\Icons.zip").

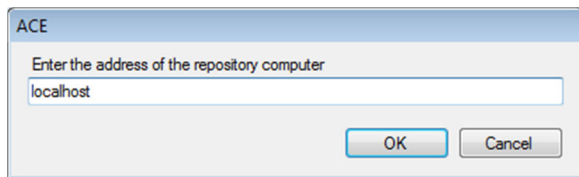
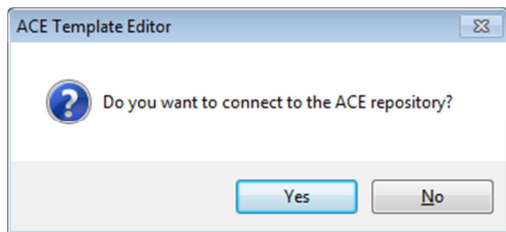
When the ACE Configuration Editor or Template Editor is run for the first time, the program will detect that the icon library has not been installed and prompt for the icon archive file. Locate an icon library ZIP file, and click the **Open** button in the prompt dialog. Both the Configuration Editor and

Template Editor also include a menu option under **Tools | Import Icon Library...** to import additional icons at any time.

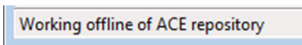
5.2 Setting Up Offline Working Directory

Note: This section applies to users who will work only in offline mode, not using repository.

When the ACE Configuration Editor or Template Editor are opened for the first time, the user will be prompted whether to connect to the ACE repository. Click the **No** or **Cancel** button to continue in offline mode.



The status bar at the bottom of the ACE Configuration Editor or ACE Template Editor window shows that you are “Working offline of ACE repository”.



To set up your local working directory for the first time, see [Setting Up Templates and Configurations](#) on page 27.

5.3 Set Up ACE Repository Server

Note: This section and its sub-sections apply only to server administrators who are setting up the ACE repository for multi-user access.

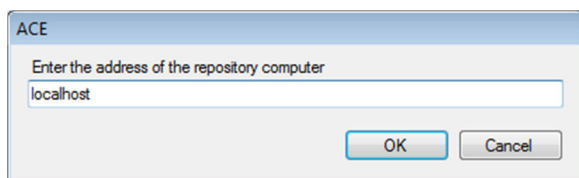
This section describes how to connect the ACE Configuration Editor and Template Editor to a repository and to set up an initial template and configuration.

5.3.1 Connect to Repository

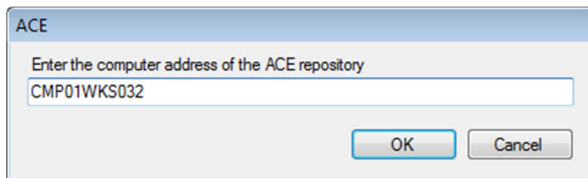
The ACE Configuration Editor or Template Editor allow the user to connect to the ACE repository to set up the repository for first-time use.

When starting the ACE Configuration Editor after its first installation and every time the editor is opened in repository mode, the user will be prompted to connect to the ACE repository. The same prompt is given after the user in offline mode enables the *Work connected to ACE repository* option in the **Tools | ACE Settings** menu.

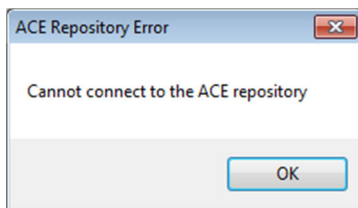
If the ACE Configuration Editor is running on the same server computer as the repository is installed, enter “localhost” or “127.0.0.1”.



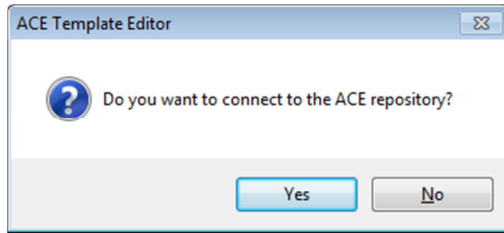
Connect to a repository on a remote computer by entering the IP address or network name of the computer on which the repository exists.



If the editor cannot communicate with the ACE Repository for any reason, you will get an error message, and the editor will switch to offline mode. Check with a network administrator for help resolving networking problems.



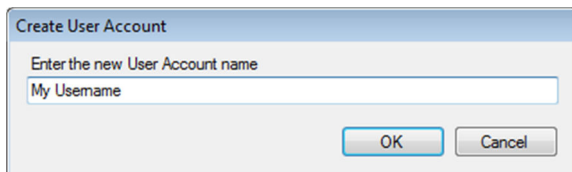
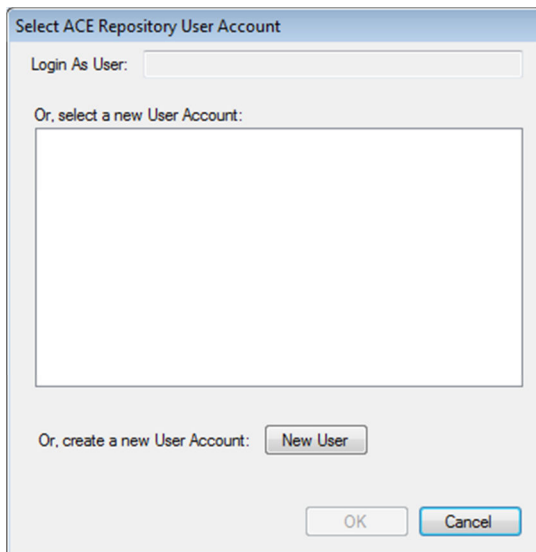
When starting the ACE Template Editor, the user will be prompted whether to connect to the ACE repository. After clicking **Yes**, enter the address of the repository as in the previous screen captures. The same prompt for repository address will be given if the user is working offline and then connects to the repository by selecting the **Repository | Connect to Repository** menu option.



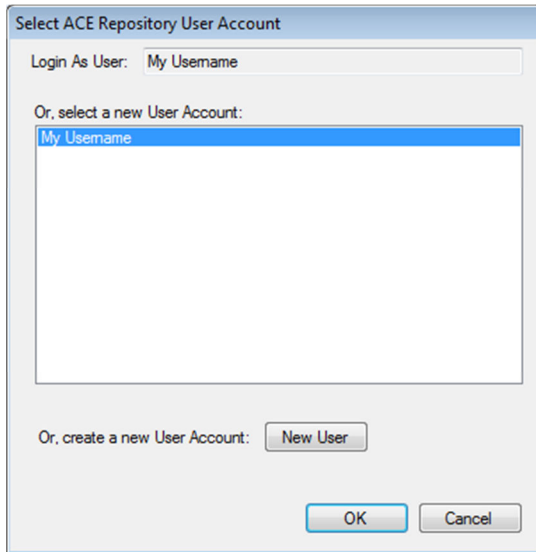
5.3.2 Create User Account

After connecting to the repository, the user will be required select a user account. Each time a configuration or template is checked out, modified, or checked in, the changes are archived in a version history according to user account.

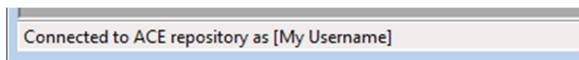
Click on the **New User** button and enter a unique user name or select an existing user account to log into the repository. Server administrators may wish to add a list of all user accounts of those who will be accessing the repository.



If you have previously created a user account, select your account name and click the **OK** button.



The status bar at the bottom of the ACE Editor or ACE Template Editor window shows that you are “Connected to ACE repository as *[your username]*”, where *[your username]* is the name of the User Account that you logged in as.



As server administrator, you will want to set up the repository to include all configuration and templates that users may wish to access. These configurations will be saved in both a local working directory and in the repository on the server computer. To set up your local working directory for the first time, see [Setting Up Templates and Configurations](#) on page 27.

5.4 Connect User Workstation to ACE Repository

Note: This section and its sub-sections apply to users who will connect to an ACE repository on a remote server.

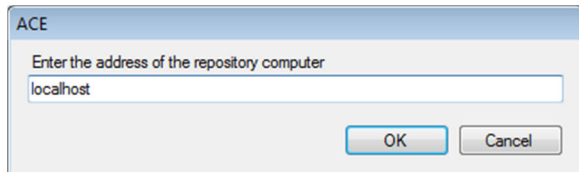
This section describes how to connect the ACE Configuration Editor and Template Editor to an existing repository and assumes that someone has previously set up the repository.

5.4.1 Connect to Repository

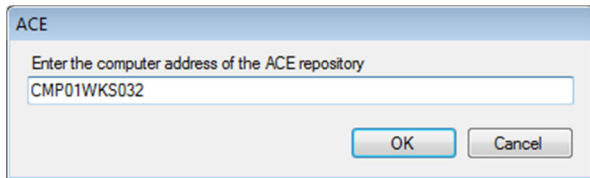
The ACE Configuration Editor or Template Editor allow the user to connect to the ACE repository to set up the repository for first-time use.

When starting the ACE Configuration Editor after its first installation and every time the editor is opened in repository mode, the user will be prompted to connect to the ACE repository. The same prompt will be given after the user in offline mode enables the *Work connected to ACE repository* option in the **Tools | ACE Settings** menu.

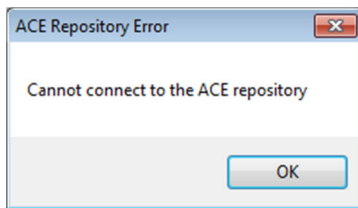
If the ACE Configuration Editor is running on the same server computer as the repository is installed, enter “localhost” or “127.0.0.1”.



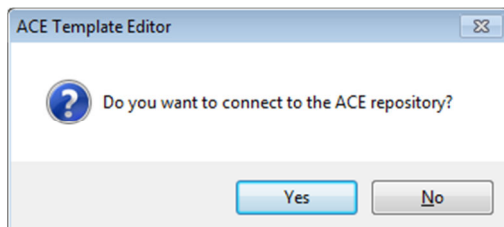
Connect to a repository on a remote computer by entering the IP address or network name of the computer on which the repository exists.



If the editor cannot communicate with the ACE Repository for any reason, you will get an error message, and the editor will switch to offline mode. Check with a network administrator for help resolving networking problems.



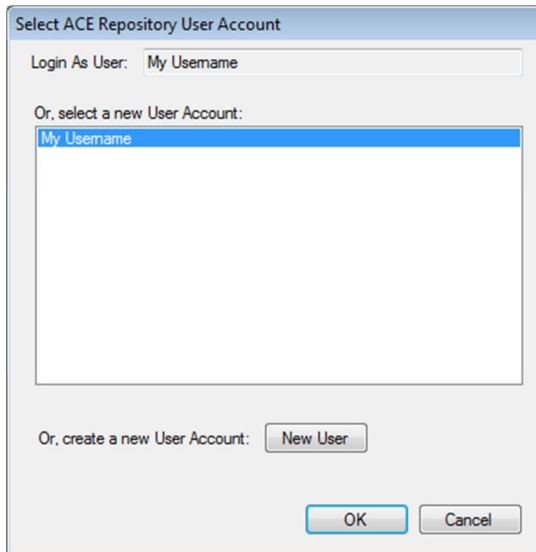
When starting the ACE Template Editor, the user will be prompted whether to connect to the ACE repository. After clicking **Yes**, enter the address of the repository as in the previous screen captures. The same prompt for repository address will be given after the user in offline mode selects the **Repository | Connect to Repository** menu option.



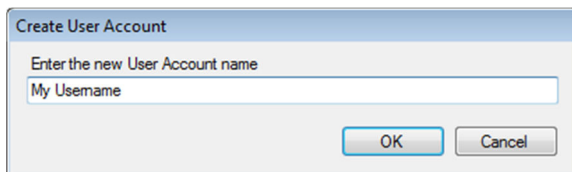
5.4.2 Select or Create User Account

After connecting to the repository, the user will be required select a user account. Each time a configuration or template is checked out, modified, or checked in, the changes are archived in a version history according to user account.

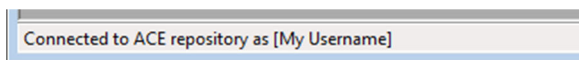
If you already have a user account, select the account name and click the **OK** button.



If you don't already have a user account, click the **New User** button and enter a unique user name.



The status bar at the bottom of the Configuration Editor or Template Editor window shows that you are “Connected to ACE repository as *[your username]*”, where *[your username]* is the name of the User Account that you logged in as.



If the repository has already been set up with configurations and templates, you may check out and modify them as needed. See [ACE Configuration Editor](#) on page 31. If you need to add configuration and templates to the repository, see [Setting Up Templates and Configurations](#) on page 27.

5.5 Setting Up Templates and Configurations

Note: This section and its sub-sections apply to users working in stand-alone or repository mode who need to import template(s) and/or configuration(s) into their ACE environment. If the working directory is already set up, see [ACE Configuration Editor](#) on page 31.

There are several different options for bringing existing templates and configurations into the ACE environment, depending on the user's situation. The following sections provide guidance for setting up the repository and/or local working directory in each of these cases.

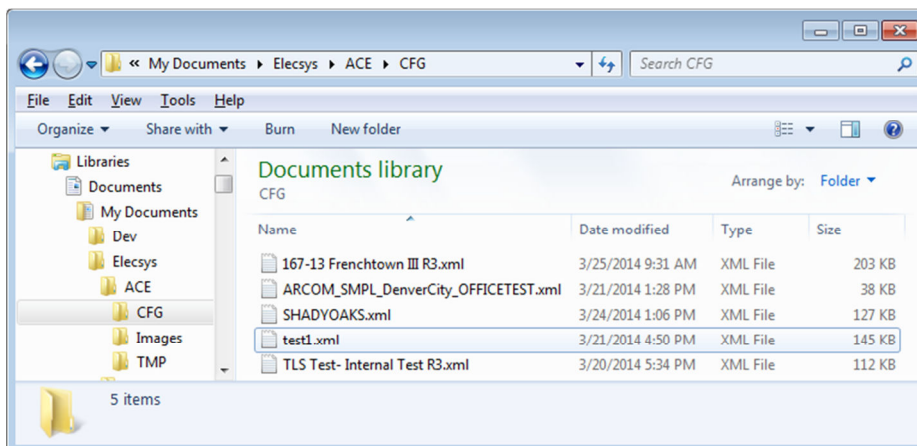
In all cases, each configuration refers to a named template that defines the options compatible with the configuration. The template must be present in the working directory before the configuration can be opened and used successfully.

5.5.1 Restore Existing XML Files (non-repository mode)

Note: This section applies to those working in non-repository mode who need to add or restore an existing XML configuration and/or template to their working directory.

If you are not working with a repository and have existing XML-based ACE configurations and templates, the files may simply be copied into your local working directory. For instance, you may have reinstalled the ACE program on a new computer and need to restore configurations from a backup drive, or you may have received a configuration from another user and need to open it to view and/or modify properties.

With the ACE programs closed, simply copy or unzip a set of configurations, templates, and/or icons into the ACE local working directory. The working directory can be changed in the ACE Configuration Editor and Template Editor, so ensure that the files are placed in the proper location.



To add configurations or templates individually to the local working directory, use the **File | Open Local File** option in the Configuration Editor or **File | Open Default Template** option in the Template Editor (or the **File | Open** option in non-repository mode of either program).

Browse to the configuration or template that you wish to add to your local working directory, and open it. Then select the **File | Save As** menu to save into your local working directory.

See the following sections for further reference:

- [Opening a Configuration](#) on page 34
- [Saving a Configuration](#) on page 38
- [Opening an Existing Template](#) on page 48
- [Saving a Template](#) on page 49

5.5.2 Restore Existing XML Files (repository mode)

Note: This section applies to those working in repository mode who need to add or restore an existing XML configuration and/or template.

If you use the repository and have reinstalled ACE on your client machine, all of the existing configurations should be stored in the repository. Simply connect to the repository and open a configuration or template. A local copy will be downloaded from the server to your local working directory.

However, if you have an existing XML-based ACE configuration or template that doesn't already exist in the repository, use the **File | Open Local File** option in the Configuration Editor or **File | Open Default Template** option in the Template Editor. Browse to the configuration or template that you wish to add to your local working directory, and open it. Then select the **File | Save As** menu to save into your local working directory. This will also add the configuration into the repository, where it can be checked in and accessed by other users.

See the following sections for further reference:

- [Opening a Configuration](#) on page 34
- [Saving a Configuration](#) on page 38
- [Opening an Existing Template](#) on page 48
- [Saving a Template](#) on page 49

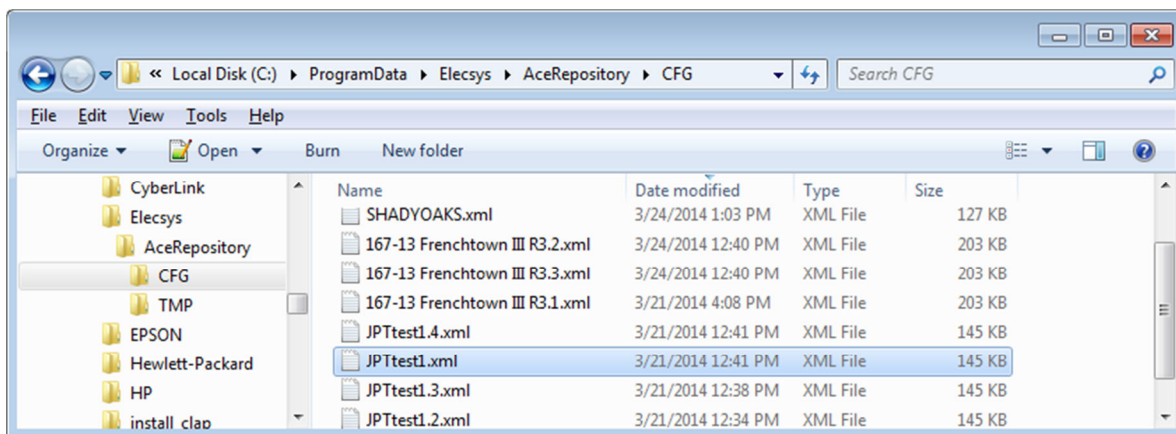
5.5.3 Restore Backup Repository Files

Note: This section applies to server administrators who are reinstalling the ACE repository and need to restore the existing XML configurations and templates.

It is recommended that the repository files on the server are regularly backed up for archival purposes. In the event that the repository is reinstalled (or installed on a different computer), the files from the original repository server can either be copied or restored from the archive.

To restore files into the repository, follow these procedures:

1. Stop the repository service (see [Starting and Stopping Repository Service](#) on page 17).
2. Remove all the files from the “AceRepository” directory location and replace them entirely with the repository files from archive (see [ACE Program File Locations](#) on page 10 for the repository location). This is important to entirely replace the directory contents, because the repository maintains a version history using a database and a collection of versioned configuration files that must be consistent.



3. Restart the repository service using the Windows Services menu.

5.5.4 Import non-XML files from old ACE

Note: This section applies to users who have old (non-XML) versions of ACE configurations that need to be brought into the new ACE environment.

The ACE installation includes a default XML template containing many of the standard Director configuration objects. However, Director customers may have legacy ACE configurations containing customized objects or properties that are not in the default template, so it is generally recommended to use existing non-XML configurations and templates to generate a project-specific template.

If you are using a default XML template:

- See [Importing a Legacy Configuration](#) on page 38, and import each of the old (non-XML) ACE configurations.
- The import operation may prompt for the name of a required template, if it doesn't already exist in the local working directory. See [Opening the Default Template](#) on page 48, and open the default XML template supplied with ACE (or supply an XML template from another location).
- Using the Save As menu option of the Template Editor, save the default template into the local working directory with whatever template name is required by the configuration.

If you are using the non-XML configurations to generate a project-specific template:

- See [Importing a Legacy Template](#) on page 50 to build a special configuration in the old ACE program containing one of all possible objects. Then use the ACE Template Editor to import this configuration as an XML template.
- See [Importing a Legacy Configuration](#) on page 38, and import each of the old ACE configurations.
- The import operation may prompt for the name of a required template, if it doesn't already exist in the local working directory. Use the Template Editor to save the template to whatever template name is required by the configuration.

When using the repository, each of the templates and configurations will need to be checked in to the repository after they are saved.

Importing a legacy configuration can have a number of problems for various reasons, including missing objects in the template, missing icons, incompatible properties in a configuration as compared with the template, etc. See [Resolving Template Problems](#) on page 58 for assistance in addressing these issues, to ensure that all legacy configuration properties are imported correctly.

6 ACE Configuration Editor

The ACE Configuration Editor allows configurations to be created, viewed, and modified. Configurations are based on templates, defined in the ACE Template Editor. The template defines which configurable objects and fields are in a configuration, and defines rules about what values can be stored in the configurable fields.

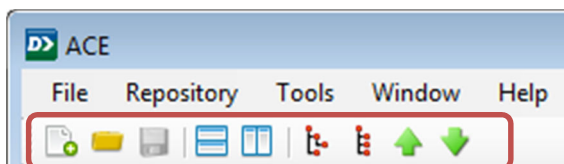
The ACE Editor may be used either in local or repository mode. The repository mode is used if there is a repository of ACE configurations, typically on a remote computer and shared with multiple users. The local mode is used on a stand-alone configuration computer. Local mode can also be used when configurations have been downloaded from a remote repository, but the configurations need to be edited locally while the repository is unavailable.

If you have existing configurations from an old version of ACE and need to import existing configurations and templates into your ACE program, or if you need to create or modify a template, see [ACE Template Editor](#) on page 47.

6.1 Using the ACE Configuration Editor

6.1.1 Tool Bar

The toolbar provides easy access to the most commonly used functions of the Configuration Editor.



New File



Open File



Save



Tile Horizontal



Tile Vertical



Expand All



Collapse All



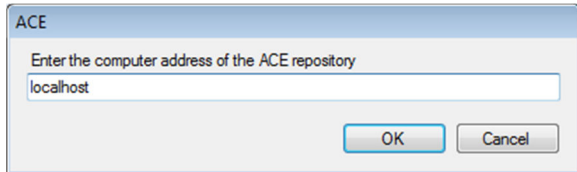
Move Node Up



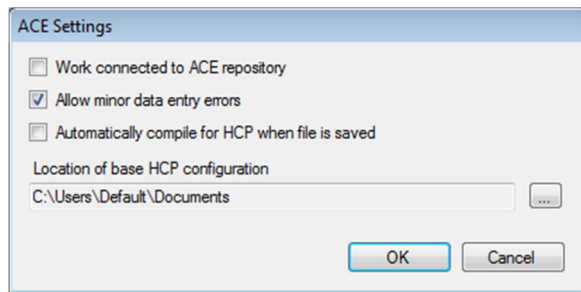
Move Node Down

6.1.2 ACE Settings

Open the ACE Configuration Editor from the Windows Start menu. Depending on the ACE Settings, you may be prompted for the repository address to connect to. If using a repository, see [Set Up ACE Repository](#) on page 21 for instructions on connecting to the repository and setting up a user account. Otherwise, click the **Cancel** button to use ACE from a local working directory.



Select the menu **Tools | ACE Settings** to set some basic properties about the ACE Editor operation.



Enable the “*Work connected to ACE repository*” checkbox to use configurations stored in a repository. You will be prompted after closing this window for the repository location. Uncheck this box to work with configurations stored in the local working directory only. This checkbox is disabled automatically if the **Cancel** button is pressed upon opening the ACE Editor, instead of entering a repository location.

Enable the “*Allow minor data entry errors*” checkbox to ignore small errors while editing the configurations. The template file contains the rules that are used to validate the entries in configurable object properties. A “minor” error would be entering a value of 11, where the template indicates the minimum and maximum property values are allowed to be between 0 and 10. In this case, it is minor because entering the value 11 results in a validation error, but it can still be encoded into the property’s data type. Certain data entry errors cannot be ignored by the ACE Editor, such as a value out of range of the field data type. An example would be to enter the value of 1000 into a UINT8 field, which is impossible since UINT8 can only encodes values from 0 up to 255.

Enable the “*Automatically compile for HCP when file is saved*” checkbox when using the ACE Configuration Editor to save Director and HCP binary files every time the configuration is saved. The HCP-NT and HCP2 are host systems which require that Director configurations be compiled into special binary files in a defined directory structure. This program setting allows the ACE Configuration Editor to automatically compile the configuration files into that structure.

When using ACE to configure an HCP system, the “*Location of base HCP configuration*” option points to the local file system directory where the ACE Editor should store the compiled binary files. See [Using HCP Configuration](#) on page 69 for more information on using ACE with HCP configurations.

Click **OK** to accept the ACE settings.

6.1.3 Changing the Local Working Directory

In the Configuration Editor, select the **Tools | Working Folder** menu option and browse to an existing ACE working directory or a new location to create a folder. The local working directory is the folder where configuration and template files are stored on the local computer, either stand-alone configurations or copies of the files contained in an ACE repository.

The local working directory defaults to the user’s My Documents, but it may be changed. Each working directory contains a settings file that determines whether to connect to the repository, what is the HCP file storage location, and other ACE options. Be aware that the configured “Working Folder” location will contain a hierarchy of folders, starting with “Elecsys\ACE” and containing the configurations, templates, and icons files.

IMPORTANT NOTE: Changing the local working directory may lead to confusion or unrecoverable loss of data. See the instructions in the following paragraphs for further information.

Also note that the Working Folder is stored in a registry entry that is used by each of the ACE programs: the ACE Configuration Editor, the ACE Template Editor, and the ACE Uploader. Care should be taken when changing this setting to avoid confusion over using different working directories.

If you open a file in one offline local working directory and then change to another offline working directory containing the same file, selecting the **File | Save** option will save the file to the original location from which the file was opened, not the current local working directory that you have just switched to. However, choosing the **File | Save As** menu option will allow you to overwrite the file with the same name in the current working directory. Care should be taken not to make conflicting changes in two places.

If you open a file in an offline local working directory, and then switch to a different working directory that is connected to the repository, the same rules apply: selecting the **File | Save** option will save to the original (offline) location, not to the repository. Selecting the **File | Save As** option will prompt for a filename to save into the repository and current working directory, rather than the original location.

Conversely, if you open a file in the repository and then select the **File | Save** option, the file will not be saved in the current location, but rather in the original working directory associated with the repository (but a copy will not be saved in the repository itself).

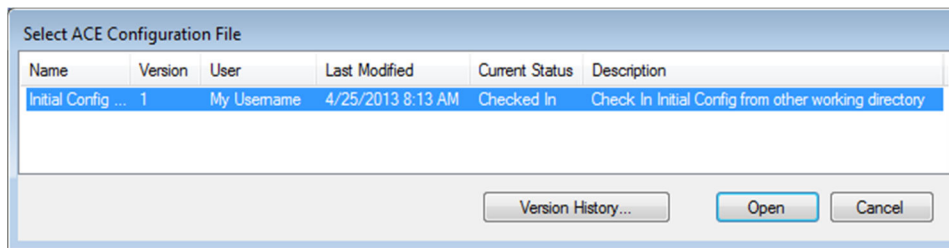
See [Saving Repository Files](#) on page 61 for important information on saving changes to repository files, and how to handle cases as described in the previous paragraph, where the local working directory becomes out of sync with the repository.

6.2 Working with Configuration Files

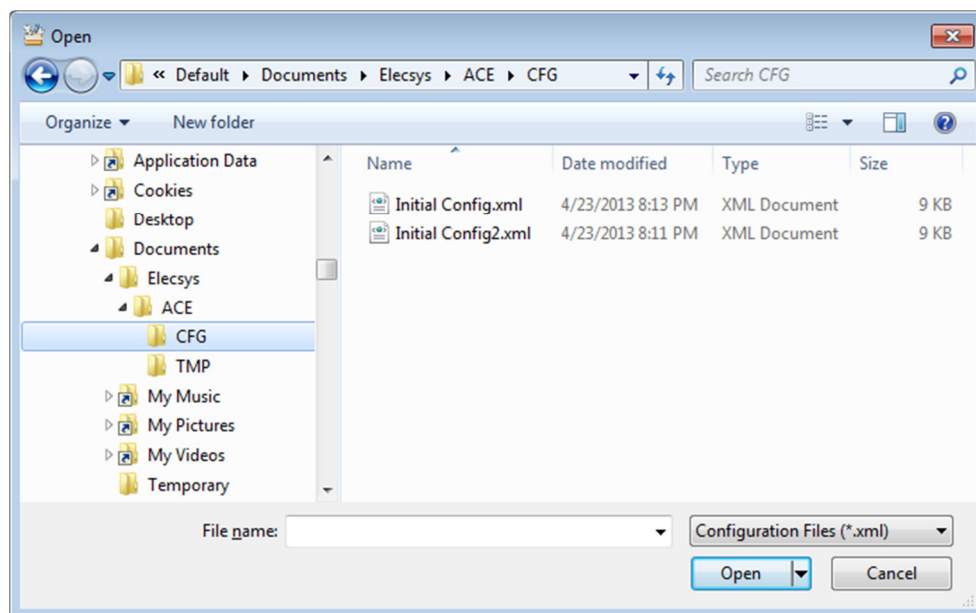
6.2.1 Opening a Configuration

Select the **File | Open** menu or click the  button on the toolbar to open an existing configuration.

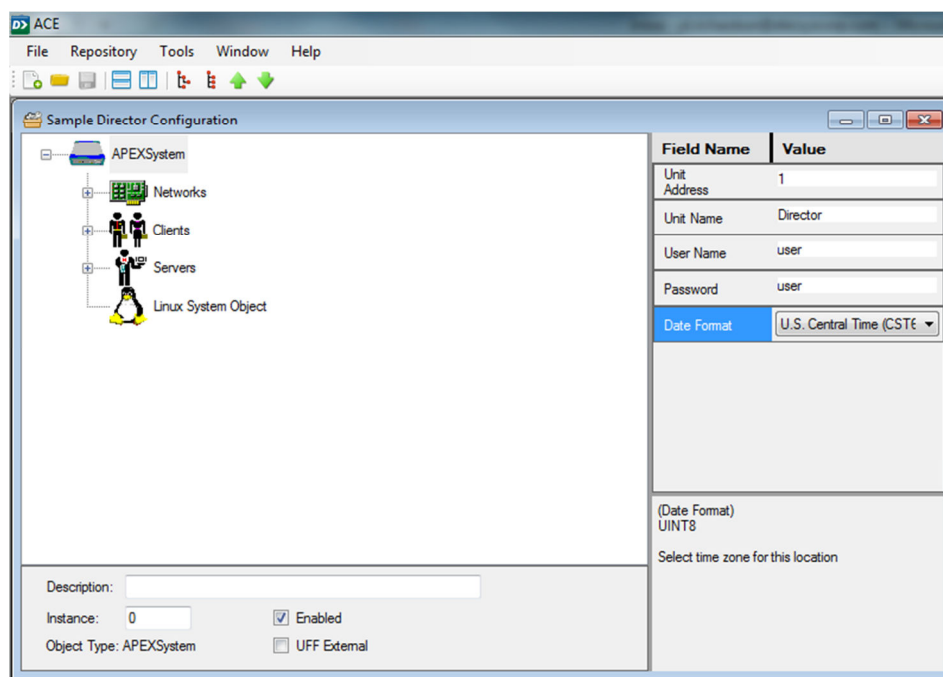
If working in the repository, you will be presented with a list of configurations available in the repository. Select the configuration to open and click the **Open** button. Optionally, click the **Version History** button to show all versions of a specific configuration. An old version of a configuration can be opened with this option.



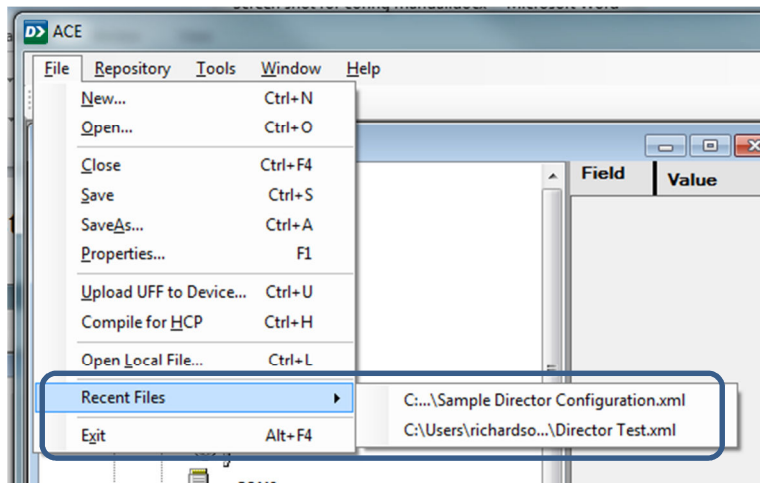
If working in the local directory, you will be presented with a file dialog. Browse to the “CFG” folder in the local working directory and choose the configuration to open.



Click the **Open** button to open the configuration. See [Editing a Configuration](#) on page 40 for instructions on editing a configuration once it's opened.



To open configurations that have been previously worked with, click **File** then **Recent Files**.

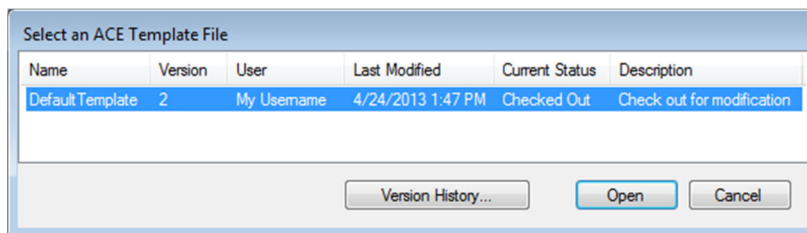


6.2.2 Creating a New Configuration

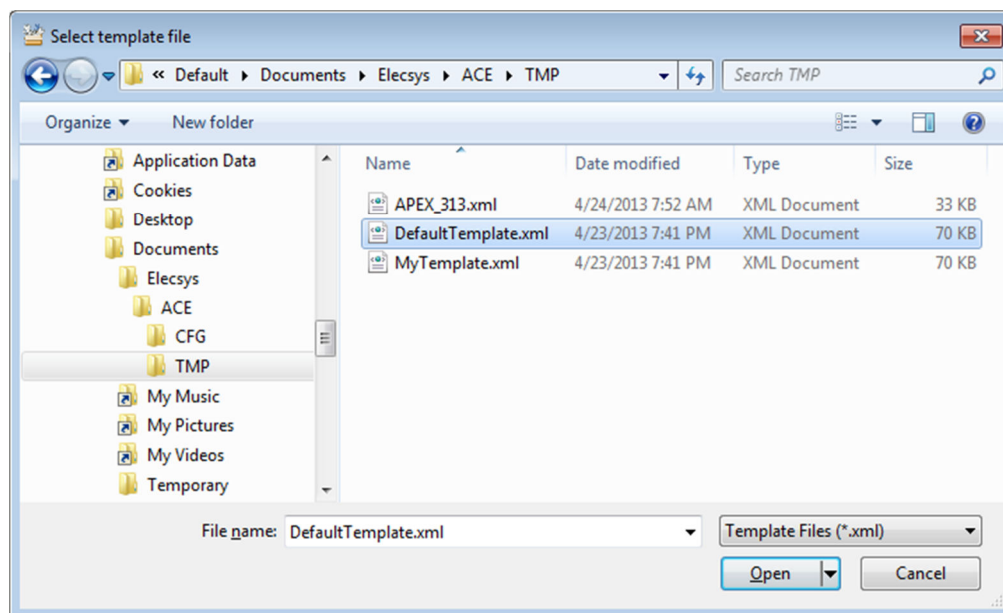
To create a new configuration, there must be at least one template file in the repository or in the local working directory, depending on the mode of operation. Template files are XML documents in a particular format recognized by the ACE Editor, to define the basic structure and capability of a configuration.

Select the **File | New** option or click the  toolbar button to create a new configuration.

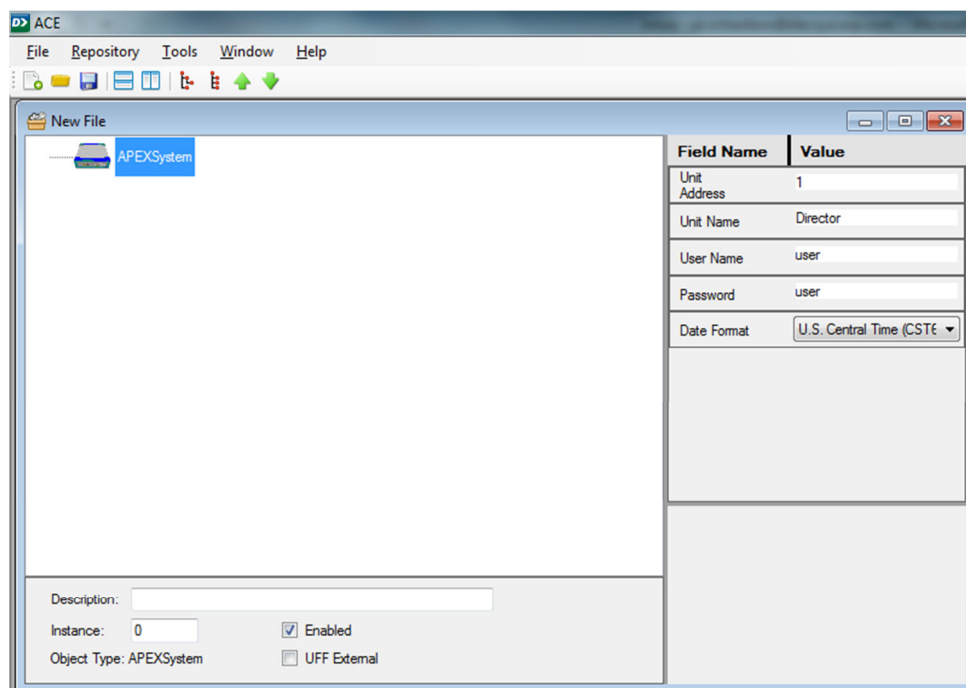
If you are connected to the repository, you will be presented with a list of available templates in the repository.



If you are working in the local working directory, you will be presented with a file dialog. Select a template file in the TMP directory. Care should be taken to choose the correct template, if more than one exists. Basing a configuration on the wrong template could result in a configuration that is not compatible with the device being configured.



After selecting the template, click the **Open** button. This creates a new configuration with a base set of objects defined in the template. After this, new objects can be added to the configuration as needed for a particular application.

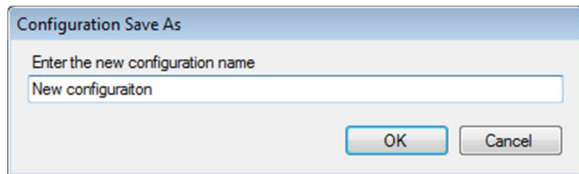


See [Editing a Configuration](#) on page 40 for instructions on editing a configuration once it has been created.

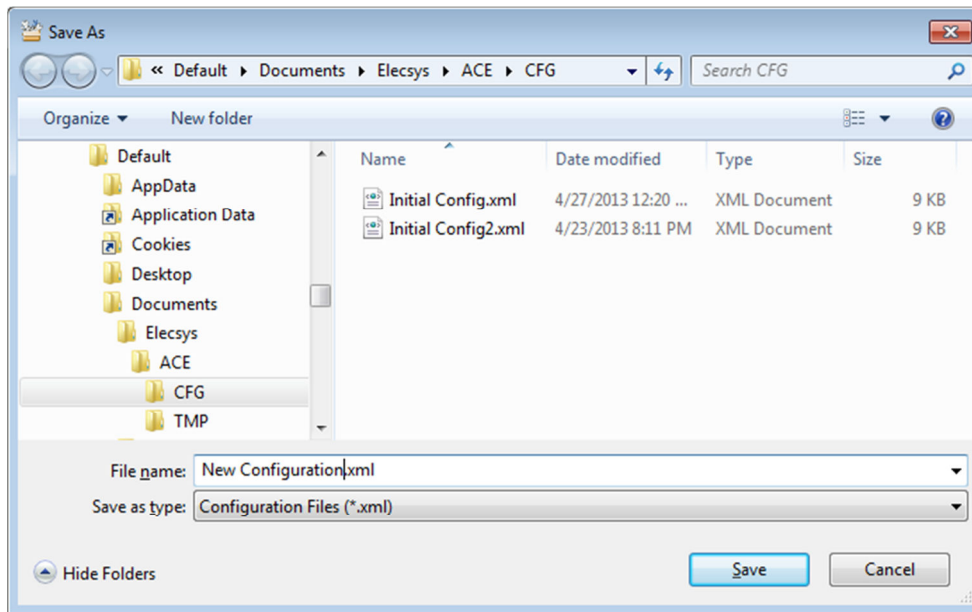
6.2.3 Saving a Configuration

With the configuration open in the ACE Editor and checked out, it can be saved by selecting the **File | Save** option. Select the **File | Save As** menu to save the configuration under a new name.

If working in the repository, enter a new name for the configuration to save in the repository.



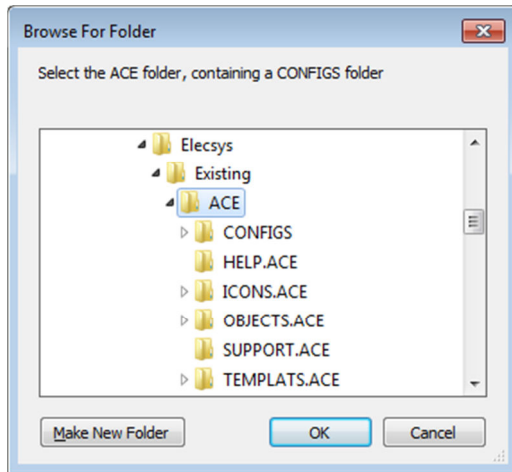
If not connected to the repository, the **Save As** menu presents a file dialog to save into the local working directory. To save the current configuration under a different name, select the "CFG" folder in the local working directory and enter a new file name.



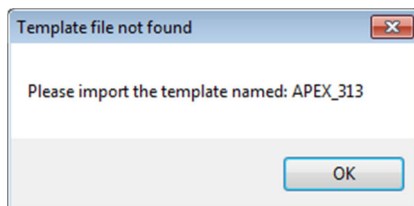
6.2.4 Importing a Legacy Configuration

In older, legacy versions of ACE (non-XML), configurations existed in a different form. In order to use those configurations in the current version of ACE, they have to be imported. Note that configurations in ACE are based on a template. In order to import legacy configurations into ACE, you must also import the template upon which the configurations were based, so that both template and configurations exist in the current (XML) ACE environment. See [Importing a Legacy Template](#) on page 50.

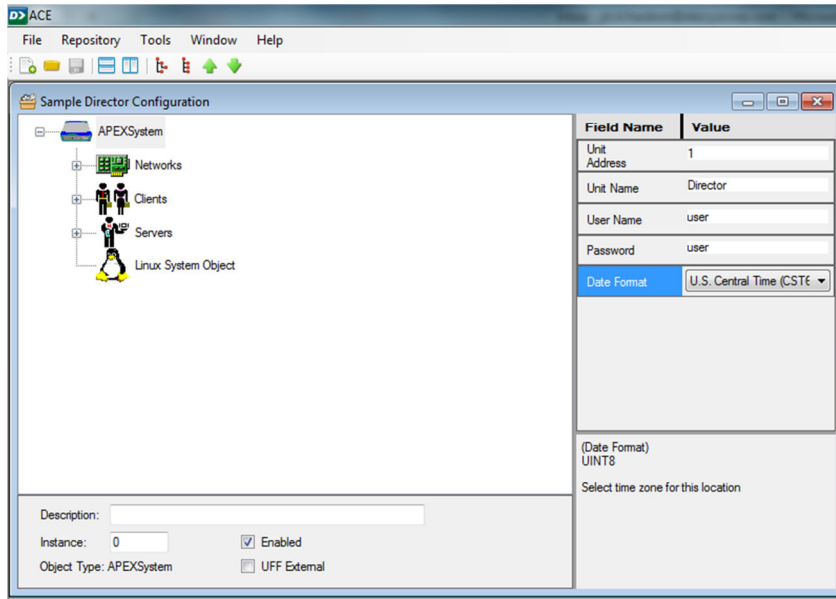
Select the **Tools | Import Legacy Configuration** menu option. A file directory dialog will be presented. Browse to the “ACE” folder of an old version of the ACE program (not the subdirectories of CONFIGS, etc.).



A list of existing configurations in the old ACE environment will be displayed. If the template doesn't exist in the current ACE environment, you will be prompted to import it first. See [Importing a Legacy Template](#) on page 50.



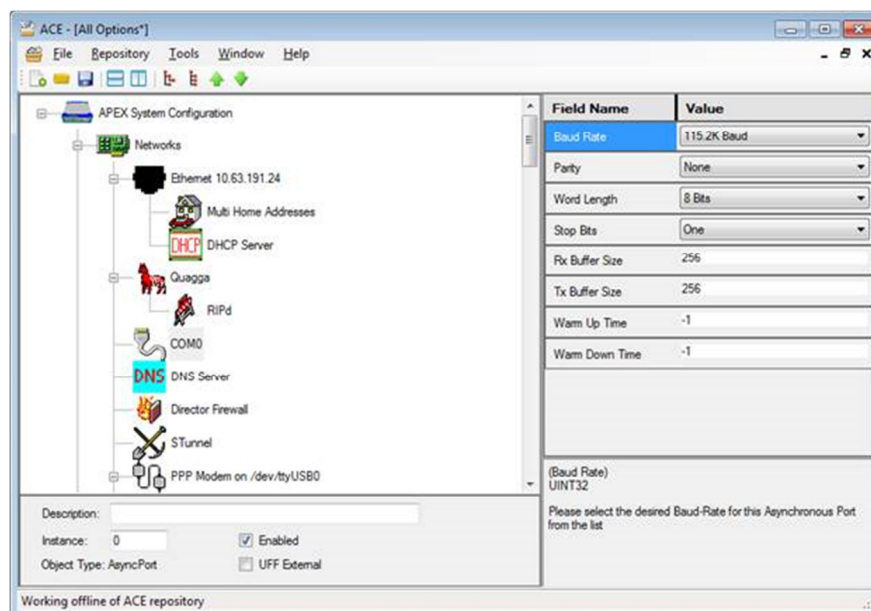
After successfully importing the configuration, see [Editing a Configuration](#) on page 40 for instructions on editing a configuration once it's opened.



6.3 Editing a Configuration

The ACE Editor allows all the operational properties of the configuration to be modified. This includes adding or deleting one or more configurable objects in the configuration hierarchy, or enabling/disabling objects. The property values that will be uploaded to a Director or used by an HCP system should be edited to satisfy the operational requirements of the system. To modify the structure of the template upon which the configuration is based (adding available child objects or changing field property definitions), see [ACE Template Editor](#) on page 47.

When a configuration is open, the main configuration window is displayed, which is divided into four quadrants. An example is shown in the following screen capture.



- The upper left quadrant contains the navigation area of the configuration. This is a hierarchical list of configuration objects and object groups, which can be expanded or contracted by clicking on the + and – icons.
- The lower left quadrant contains an optional Description of each object, and some basic properties that are common to all objects: Instance, Object Type, Enabled, and UFF External.
- The upper right quadrant contains the Object-specific properties of this instance of the configuration object. These properties can be modified according to the specific needs of the system being defined in this configuration.
- The lower right quadrant gives additional descriptive information about each property in the object, such as the data type of the field, a property description, and the range of allowed values.

6.3.1 Configuration Properties

With a configuration open in the ACE Editor, select the **File | Properties** menu option to display properties of the configuration. An example is shown below.

Base Template Name

Name of the ACE template on which this configuration is based.

Created

Date the configuration was created.

Last Modified

Date the configuration was last modified.

UFF Filename

Name of the binary file that contains all property values expressed in all objects within this configuration (excluding those objects for which the “UFF External” option is selected). The filename must not contain any embedded space characters. And the filename must end in the .uff file extension.

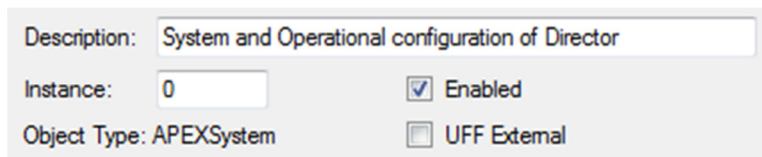
Description

Optional description of the configuration.

- Secure Copy** This option is used to store information about the Director to which this configuration will be uploaded, when using secure copy (SFTP/SCP).
Host Name (IP address) of the Director
User Name of the user account on the Director
Home Directory where the configuration will be stored. Typically on a Director, the Home Directory should be: /home/director/
- FTP** This option is used to store information about the Director to which this configuration will be uploaded, when using FTP file transfer.
Host Name (IP address) of the Director
User Name of the user account on the Director
Home Directory where the configuration will be stored. Typically on a Director, the Home Directory should be: /home/director/

6.3.2 Object Properties

The Properties section of each object appears in the lower left portion of the ACE Editor when the object is selected. The list of available properties is the same for each object. Below is an example of the properties in the highest level parent object in a Director configuration.



Description: System and Operational configuration of Director

Instance: 0 ☒ Enabled

Object Type: APEXSystem ☐ UFF External

- Description** Each object includes an optional Description field. This allows the system designer to describe each object and its purpose or function.
- Instance** Each child object under its parent is required to have an Instance number to uniquely identify it within the system configuration.
Some objects are allowed to have multiple instances, so the Instance number may have a range of values starting from zero. Other objects are only allowed a single instance under their parent, so the Instance number is required to be zero. In some cases, there are additional specific requirements for the Instance number, which are described in the Properties section for the object in the Director Configuration Manual.
- Object Type** This is a unique name of the object, used internally by the Director or HCP software.
- Enabled** Each object has an Enabled property, which allows the object and its functionality to be disabled in the system.
When an object is disabled by clearing its checkbox in the ACE Editor, all of its child objects are also disabled. In some cases, care should be given to disabling an object or its children, because it may cause errors or have negative impact on the system operation. For instance, if a Master Channel is configured to scan a Field Unit but the Field Unit has been disabled, or if the

RTDB under a Field Unit has been disabled, errors may occur in the Director when trying to run this configuration.

UFF External

This checkbox tells whether to store the object properties in a single file (.uff extension), or to save them separately, external to the .uff file.

In a very early version of the ACE program, the configuration values for each configuration object were saved in a separate file (named according to the Object Type), and all files in a configuration had to be downloaded to a Director.

This became burdensome as the number of objects increased, and a Unified File Format file was created (UFF). Now, all the individual files are stored collectively in a single configuration file with a “.uff” extension. This provides the convenience of storing and downloading typically only a single configuration file to the device.

However, in a few cases there is still a reason to keep certain objects stored in separate files, apart from the .uff file. For instance, certain files contain lists of values that are saved in Flash memory and modified through ISaGRAF or in other ways. Having a separate file for these values allows only that file to be re-written to disk when the values are changed during runtime. For those few objects that are recommended to be stored separately, the Director Configuration Manual mentions that the UFF External option should be enabled.

6.3.3 Modifying Object Field Values

Configuration objects contain unique fields, which provide the specific configurations of various features within a system. This section of this manual describes the specific fields and corresponding values of those fields. Descriptions are given for the various fields, to explain the options, how they work, and other information for the system designer to understand how the ACE configuration should be defined to achieve the intended results.



Field Name	Value
Name	DF1
Channel Type	Direct Master Flex Scan Table
Auto Start	Yes
Response Timeout	5000
Broadcast Delay	0
Interpoll Delay	100
Scan Effective Limit	99
Network Recovery	0
Scan Table	<input type="button" value="Edit Table (1)"/>

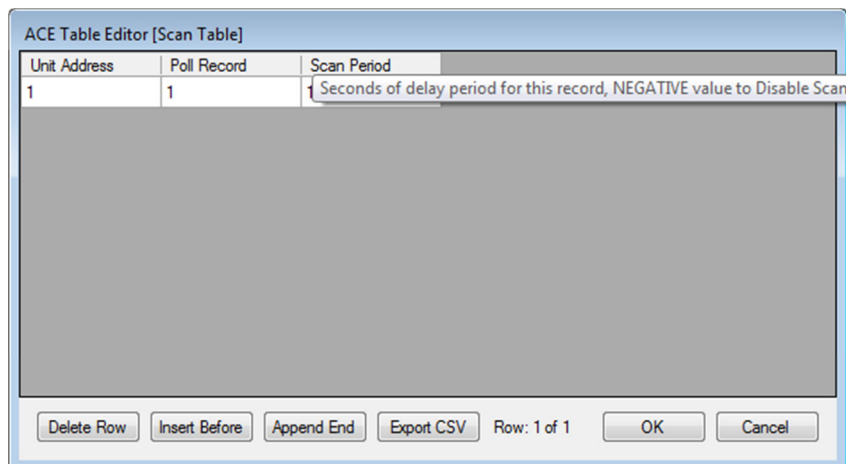
(Scan Table)
TABLE

Click the button to edit the table.

Each field has a data type (signed or unsigned integer, Boolean, text, table) and an allowed range of values (numeric range, or length of text fields). If a value is entered outside the allowed range for a field, an error will be displayed. Below is an example, showing the fields for the Master Channel object.

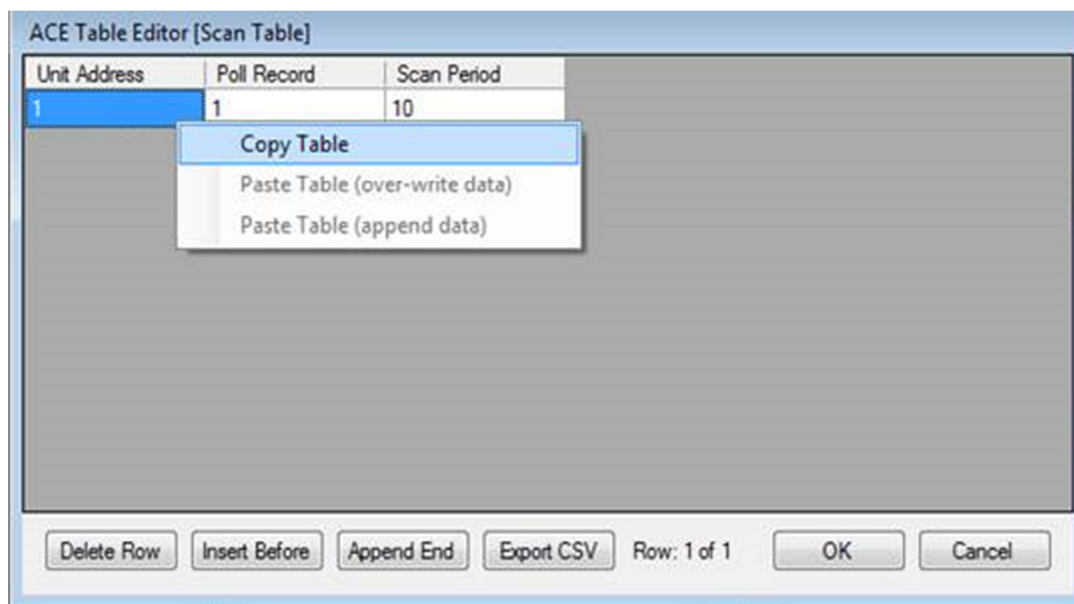
6.3.4 Edit Table

Some ACE objects contain fields that are not a single value, but an entire table of values. This will be indicated with an Edit Table button. Within the table, multiple rows and columns contain the property values for this field. One or more rows may be entered, but as with other fields, the values must be entered within the allowed range. Below is an example, showing the Scan Table of a Master Channel.



6.3.4.1 Table Editor Copy and Paste

To assist in creating large tables of values, the ACE Editor allows the entire set of values to be copied from one table and pasted into another table. Open the table editor containing the data from which to copy. Right-click anywhere on the table, and select the Copy Table menu item.



Open the table editor for the object into which you wish to paste the data. Right click on the table editor window, and select one of the two paste options.

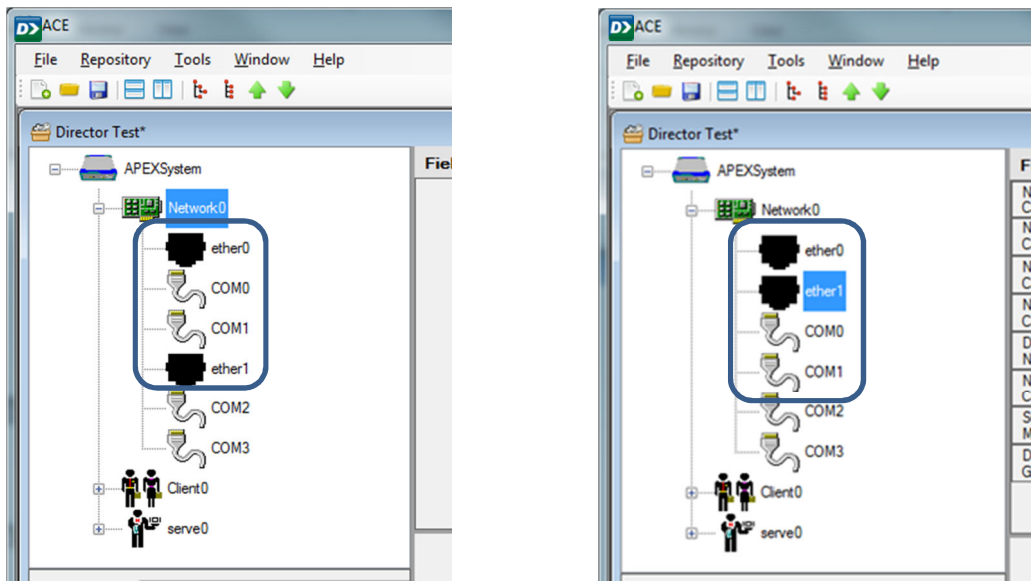
Choose the over-write data menu item if you wish to delete any current data from the table and replace it with the data from the clipboard.

Choose the append data menu item if you wish to add the clipboard data to the end of the table.

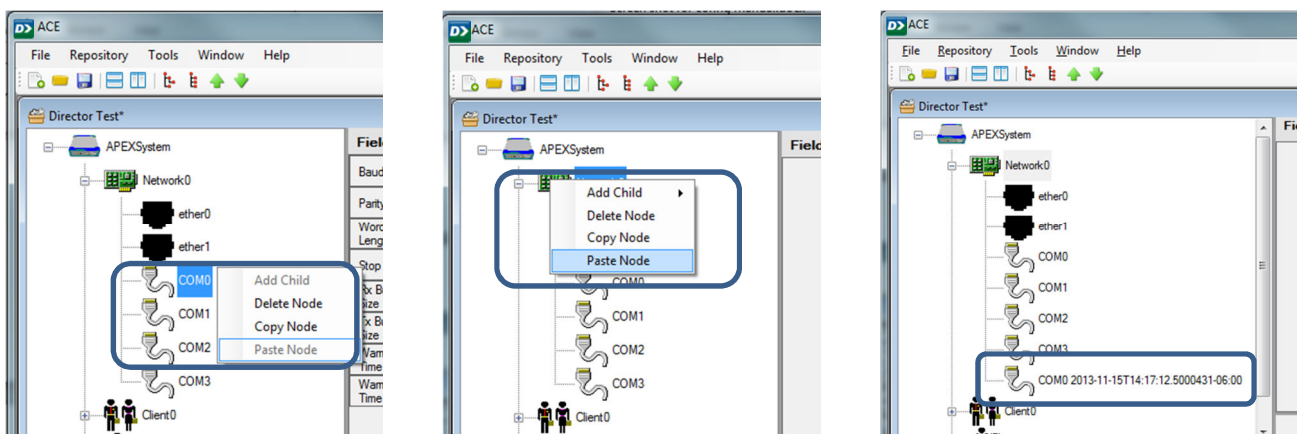
Note: This table copy/paste feature only works for transferring data between table fields of a similar type. For instance, only Scan Table data can be pasted into another Scan Table.

6.3.5 Moving, Copying and Pasting Nodes and Branches

Use the Move Node Up or Move Node Down function to reposition the node/branch if desired.



Right click on a node and select “Copy Node”. Then right click on its parent node and select “Paste Node”. The new node will be an exact duplicate of the original node/branch except a date/time stamp will be appended to every object name. (The next available ‘Instance Number’ will be applied to the major instance position).



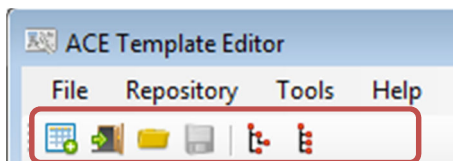
7 ACE Template Editor

The ACE Template Editor allows ACE template files to be created, viewed, and modified. In addition, the template editor allows configuration templates to be imported from older, legacy versions of ACE. The ACE template file defines what objects are available in the hierarchical structure of an ACE configuration based on that template, and occasionally changes need to be made to that structure as the capabilities of the Director software are modified.

7.1 Using the ACE Template Editor

7.1.1 Tool Bar

The toolbar provides easy access to the most commonly used functions of the Template Editor.



Open Default Template



Import Legacy Template



Open File



Save



Expand All



Collapse All

7.1.1 Changing the Template Working Directory

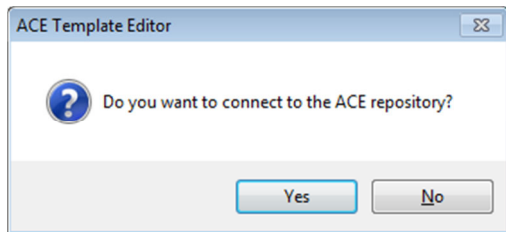
In the Template Editor, select the **Tools | Application Settings** menu option and browse to an existing ACE working directory or a new location to create a folder. The local working directory is the folder where configuration and template files are stored on the local computer, either stand-alone configurations or copies of the files contained in an ACE repository.

IMPORTANT NOTE: Changing the local working directory may lead to confusion or unrecoverable loss of data. See [Changing the Local Working Directory](#) in the Configuration Editor section, on page 33, for further information.

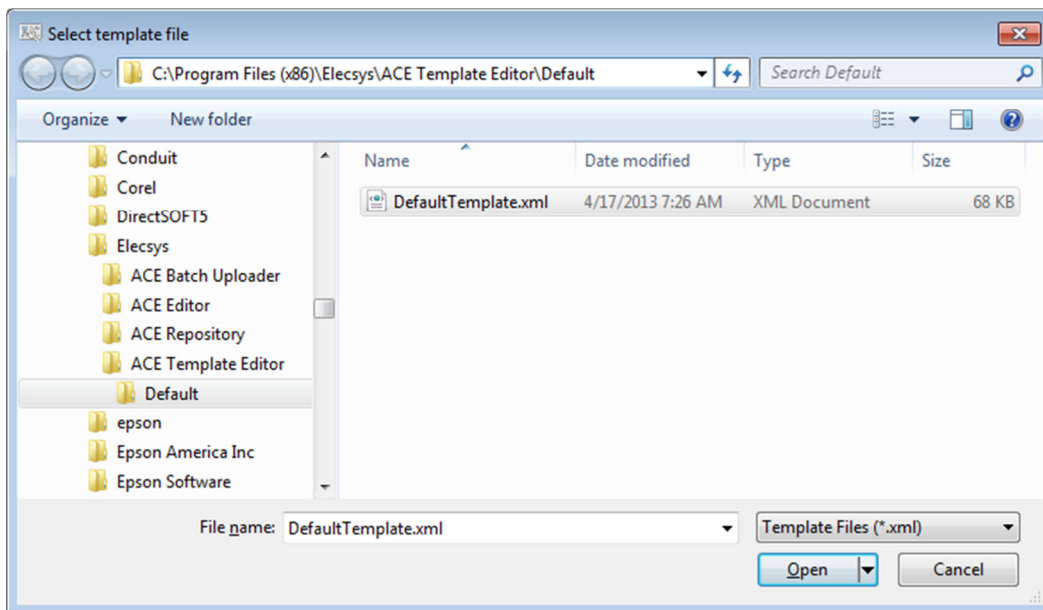
7.2 Working with Configuration Files

7.2.1 Opening the Default Template

Open the ACE Template Editor from the Windows Start menu. You will be prompted whether to connect to the repository. If using a repository, click the **Yes** button (see [Set Up ACE Repository](#) on page 21 for instructions on connecting to the repository and setting up a user account). Otherwise, click the **No** button to use ACE from a local working directory.



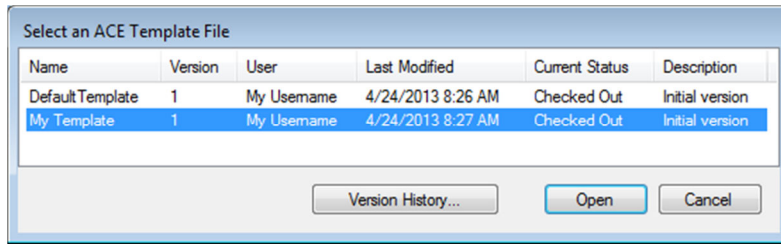
Select the **File | Open Default Template** menu option. The ACE Template Editor includes a default template that can be used as a starting point that can be modified according to the needs of an application. The Default Template is stored in the Program Files directory of the ACE Template Editor. You can also browse to another directory to open an existing template, if one exists, to either save a new copy or to import into the repository.



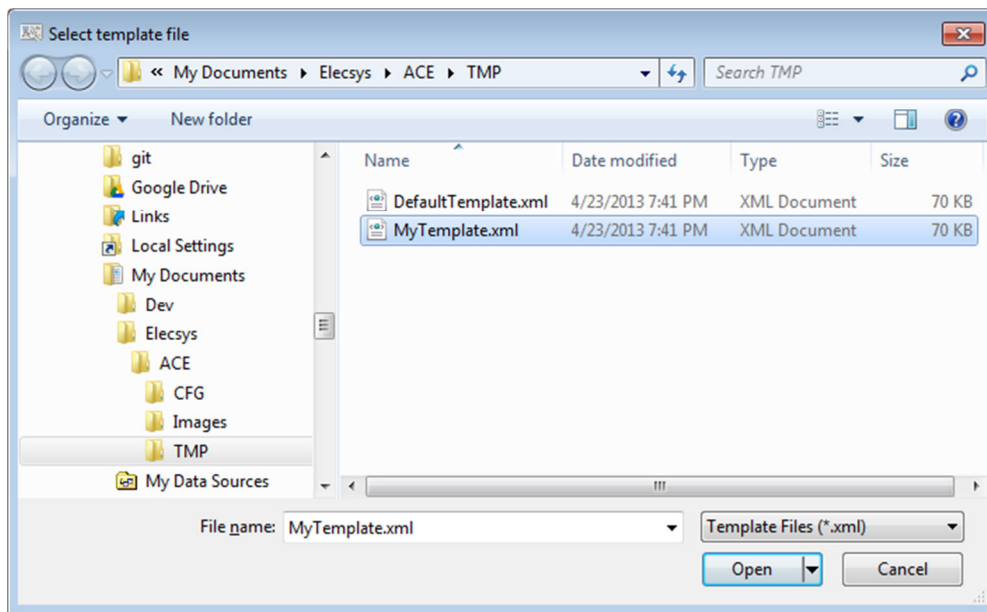
7.2.2 Opening an Existing Template

To open a template that has already been created in ACE, select the **File | Open** menu.

If you are connected to the repository, a list of templates available in the repository will be presented.



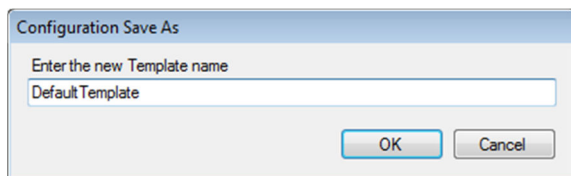
If you are not connected to the repository, a file dialog will be presented.



7.2.3 Saving a Template

With a template open in the ACE Template Editor, it can be saved under its current name by selecting the **File | Save** option. Select the **File | Save As** menu to save the template under a new name.

If connected to the Repository, enter the name of the template to save in the Repository. A copy will be automatically saved into your local working directory as well.



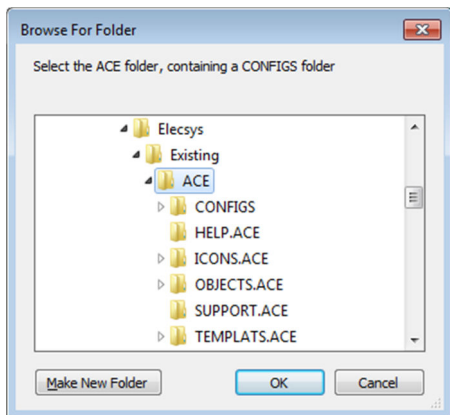
If not connected to the Repository, the Save As menu allows a copy of the template to be saved into the local working directory.

7.2.4 Importing a Legacy Template

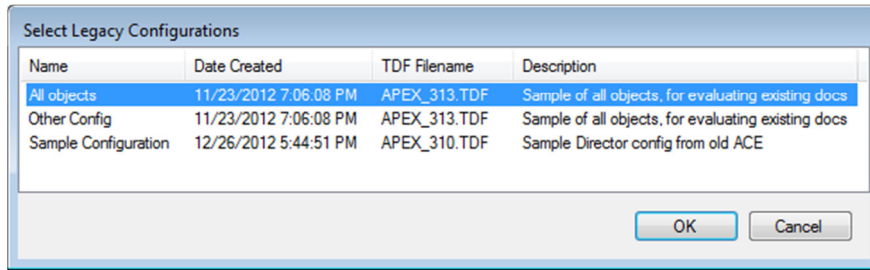
In older versions of ACE (non-XML), templates existed in a different form. In order to use those templates in the current version of ACE, they have to be imported. Note that configurations in ACE are based on a template. In order to import legacy configurations into ACE, you must also import the template upon which the configurations were based, so that both template and configurations exist in the current (XML) ACE environment.

***IMPORTANT:** When importing a legacy template, the ACE Template Editor uses the existing configurations as a guide to which objects should be included. Not all objects in the original template will be included, unless they are also in the configuration. Before importing the legacy template, it is highly recommended to create a configuration in the old style of ACE, and add at least one of every child object under every parent. This all-encompassing configuration should be the one selected for the import, so that the new template includes all possible options.*

Select the **File | Import Legacy Template** menu option. A file directory dialog will be presented. Browse to the “ACE” folder of an old version of the ACE program (not the subdirectories of TEMPLATS.ACE, etc.).



A list of existing configurations in the old ACE environment will be displayed.



This list shows the name of the configuration, the date created, and its description. It also shows the TDF Filename (legacy template definition) upon which the configurations were based. Choose the configuration that is most representative of a complete set of all possible configuration object types, one that is representative of the configurations to be imported later by the ACE configuration editor. Then click the **OK** button.

This template can be saved in the local working directory and/or repository and modified as needed.

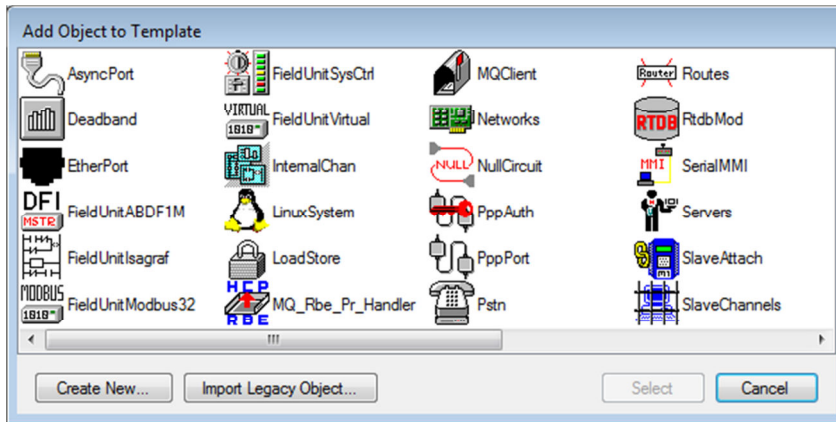
7.3 Editing a Template

The ACE Template Editor allows all the properties of the template to be modified. This defines the structure of what child objects may be added to each parent, the object properties (icon, instance numbers, and compiled file name), and all the available field and field values that make up the object. There are several levels of menus involved when editing the template objects:

7.3.1 Add/Delete Child Object

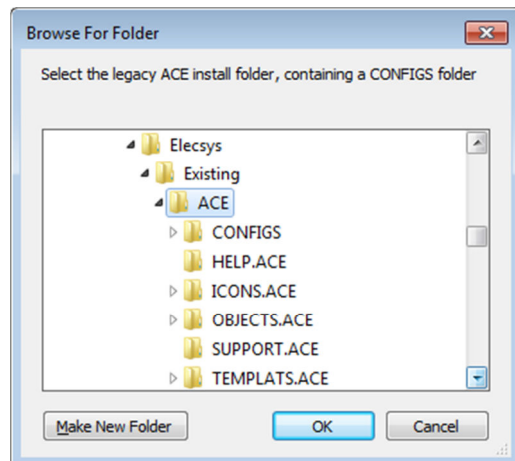
To delete a child object, right-click on the parent and select the **Delete** drop-down menu. You will be prompted to confirm the deletion. Deleting an object will automatically delete all children of the object being deleted.

To add a child object, right-click on the parent object and select the **Add Child Object** drop-down menu. A menu of objects currently available in the template is shown.



If the new object already exists in the list, click on the icon and click the **Select** button. The object will be added, along with all children objects under it as currently defined in the template.

If the object doesn't currently exist, an ACE object from an old (non-XML) version of ACE can be imported. Click the **Import Legacy Object** button and browse to the "ACE" folder of an existing installation of ACE.



Select the legacy object (.odf extension) of the object to be imported into the current ACE installation. The object and its icon will be added to the list of available objects.

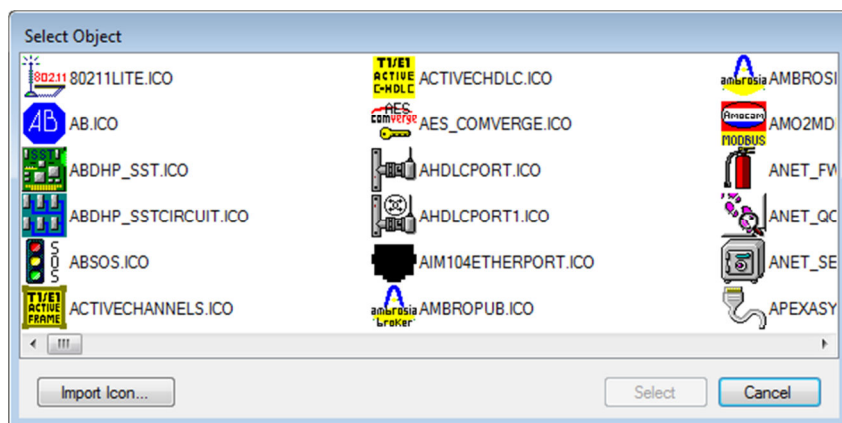
You can also create a new object in the current ACE environment by clicking on the **Create New** button. (Be aware, however, that any objects created or imported into the new ACE environment must be supported by and defined in a manner compatible with the Director or HCP software in order to be functional.) See the following sections for information on adding and modifying properties of a new object.

7.3.2 Edit Object Properties

This section describes how to set the properties of an ACE object. This and the following sections apply to either modifying an existing object or creating a new one.

In the Template Editor, right-click on an object and select the drop-down option **Edit Properties**. The object properties are illustrated using the Modbus Field Unit object.

<u>Properties</u>	<u>Function</u>
Object Type Name	Enter a unique name of this object type.
Compiled File Name	<p>Name of binary file containing properties to be downloaded to a Director.</p> <ul style="list-style-type: none"> • Use the hash symbol (“#”) to indicate a number of digits in the file name to contain the object’s instance number, for objects that can have multiple siblings under the same parent. • Use the dollar sign (“\$”) to indicate that the object will inherit the instance numbers of its parent tree of objects. • In the above example, the parent tree for the Modbus Field Unit includes the Master Channel (MastChan##) and Async or Network Circuit (Cir\$\$\$). Thus, the Field Unit object has four digits of instance number inherited from its parents, and four additional digits defined in each unique Field Unit (os\$\$\$\$###). A specific instance (123) of the Modbus Field Unit on channel 2, circuit 0 would thus have a compiled file name of “os02000123”. • An object file name may use dollar signs to inherit a parent instance number, and yet not include the hash symbol to add additional instance numbering. This could be the case if an object is required to be a sole child under a parent tree that allows multiple children.
Minimum and Maximum Instance Number	<p>Enter the range of instance numbers allowed for this object. If no instance number is allowed, enter zero (0) for both Minimum and Maximum (and the Compiled File Name doesn’t need to include the hash symbol).</p>
Select Icon	<p>Click the Select Icon to associate an icon with this object.</p> <ul style="list-style-type: none"> • If the icon is already included in the repository (or local working directory), choose it from the list of icons. • If the icon isn’t included, click the Import Icon button to import an icon file from an old version of ACE

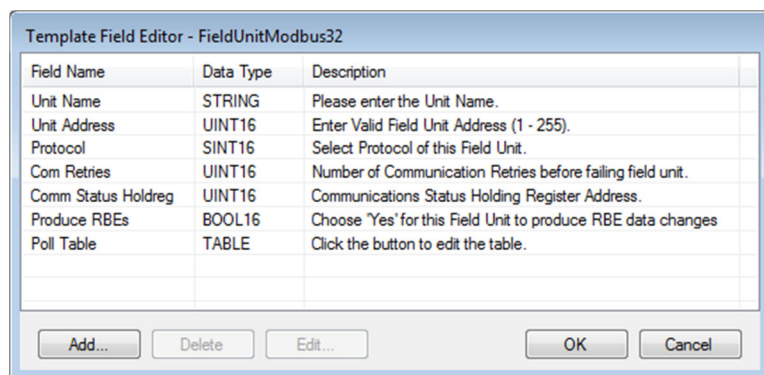


Edit Fields Click the **Edit Fields** button to add fields specific to this object. See [Edit Fields and Field Properties](#), below, for information on configuring object fields.

7.3.3 Edit Fields and Field Properties

This section describes how to edit the fields in an ACE object.

Click the **Edit Fields** button. The object fields are illustrated using the Modbus Field Unit object.



Click the **Delete** button to delete an existing field.

Click the **Add** or **Edit** buttons to add a new field or edit the existing field properties. The following table explains the properties of template fields.

The image shows a 'Field Properties' dialog box. It contains the following fields and controls:

- Field Name:** A text box containing 'Protocol'.
- Data Type:** A dropdown menu showing 'SINT16'.
- Has options:** A text box showing '2' and an 'Edit Options List...' button.
- Default Value:** A text box showing '12'.
- Description:** A text box showing 'Select Protocol of this Field Unit.'
- Color:** A text box (empty).
- Minimum:** A text box showing '0'.
- Maximum:** A text box showing '2147483647'.
- Field is read-only:** An unchecked checkbox.
- Buttons:** 'OK' and 'Cancel' buttons at the bottom.

Properties

Field is read-only

If this checkbox is enabled, the field cannot be edited in the ACE Editor, and will always have the value specified by Default Value.

Field Name

Enter a unique name of this field (unique within this object only).

Data Type

- Select the data type for data to be entered in this object field. Note, this is very important to configure the correct Data Type that matches the configuration structure expected in the Director software.
 - *BOOL8 – Boolean value, occupies 8 bits in .uff configuration file*
 - *FLAG8 – Same as UINT8 data type, 8 bits*
 - *UBYTE – Unsigned byte value, 8 bits. Same as UINT8 data type.*
 - *UCHAR – Single character, 8 bits. Same as UINT8 data type.*
 - *UINT8 – Unsigned integer, 8 bits (0 to 255)*
 - *SBYTE – Signed byte value, 8 bits*
 - *SINT8 – Signed integer, 8 bits (-128 to +127)*
 - *BOOL16 – Boolean value, occupies 16 bits in .uff configuration file*
 - *FLAG16- Same as UINT16 data type, 16 bits*
 - *SINT16 – Signed integer, 16 bits (-32,768 to 32,767)*
 - *UINT16 – Unsigned integer, 16 bits (0 to 65,535)*
 - *SINT32 – Signed integer, 32 bits (-2,147,483,648 to 2,147,483,647)*
 - *UINT32 – Unsigned integer, 32 bits (0 to 4,294,967,295)*
 - *STRING – ASCII string (Minimum and Maximum fields determine allowed length)*
 - *IPADDRESS – allows four-octet IP address to be entered*
 - *UINT16LIST – provides a drop-down list of options, where each option is defined with a text description and a 16-bit unsigned integer value. The integer values (not the description) are entered into the .uff configuration file.*
 - *TABLE – Provide a multi-column table of values. (See [Edit Table Properties.](#))*
 - *RAWTABLE – Provides a multi-column table of values.*
 - *COMMENT – A field is provided to describe an element in the ACE configuration.*

Edit Options List

For fields where the selection should be a drop-down list instead of data entry, click the **Edit Options List** button. This allows the list of options to be defined (see [Edit Options List](#)).

Default Value

Enter the default value to be included for this field when the object is first created. The default value should be between the Minimum and Maximum values, and for

Description

drop-down fields it should equal one of the allowed “Values” in the list of options.

- For fields with a option list, set the Default Value in the Options List window.

Enter an optional description for this field, to guide the configuration user with information on the purpose or use of the field.

Color

Enter an optional color for the name of the field. If the color is omitted, the field name is displayed in black in the ACE Editor.

- Colors may include names such as: red, blue, green, purple, orange, yellow, gray, brown, tan, indigo, etc. An error flag will appear to the right of the field if a non-supported color name is entered.

Minimum, Maximum

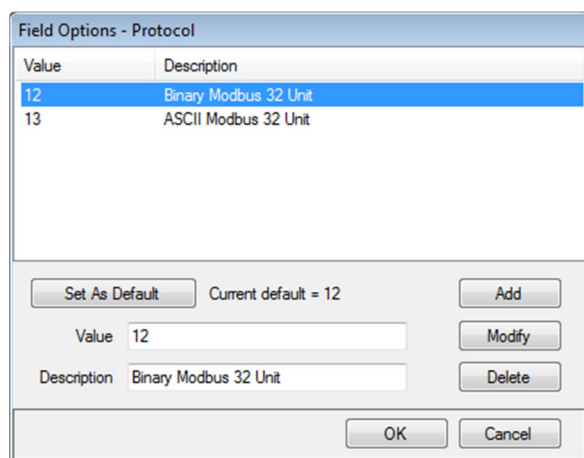
Enter the minimum and maximum allowed values for the field.

- For numeric data types, this range should not exceed the possible range of values as determined by the Data Type. However, the range of minimum and maximum values can be made smaller, to prohibit the user from entering values that aren’t allowed in the Director software.
- For the String data type, the Minimum and Maximum fields determine the length of the string allowed to be entered. Minimum should be 0 or greater. Maximum should be the number of allowed characters plus one, to allow for a null-terminated string.

7.3.4 Edit Options List

This section describes how to modify the Options List for a field using a drop-down list in an ACE object.

In the Field Properties window for a field requiring a drop-down list (see [Edit Fields and Field Properties](#)), click the **Edit Options List** button. This brings up a list of available options. The option fields are illustrated using the “Protocol” field of the Modbus Field Unit object.

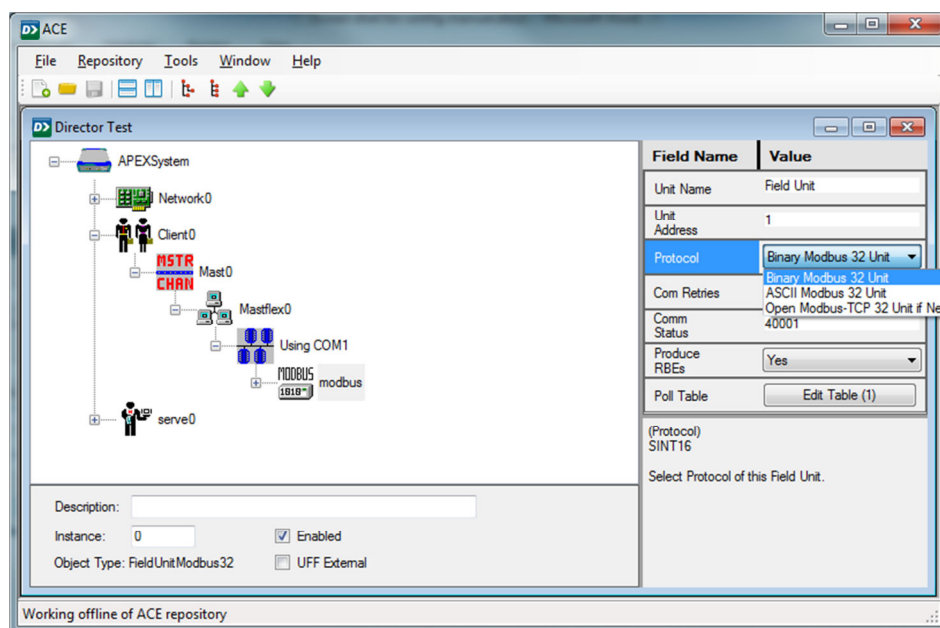


Click the **Delete** button to delete an existing option.

Click the **Add** button to add a new option to the existing field property options.

Properties	Function
Value	Enter the value associated with the option.
Description	Enter a description of the option that will be displayed in the drop-down list.
Set as Default	Select one of the options that you want to be the default when the object is first created, and click the Set as Default button. This value will be copied into the Field Properties window.

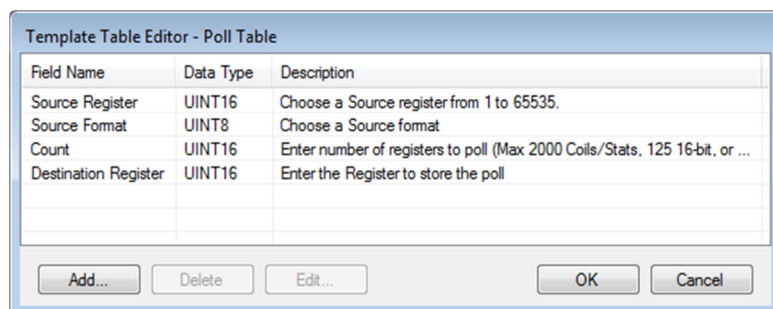
In the ACE Editor, the Modbus Field Unit “Protocol” field appears as shown below. The selection list shows the Description of each field option. The associated value (12 or 13) is not shown in the list, but is stored in the .uff configuration file based on selection made in the user configuration.



7.3.5 Edit Table Properties

This section describes how to modify the Table definition for a field using the TABLE or RAWTABLE data type.

In the Field Properties window for a Table field (see [Edit Fields and Field Properties](#)), click the **Edit Table** button. This brings up a Table Editor. The Table fields are illustrated using the “Poll Table” field of the Modbus Field Unit object.



Click the **Delete** button to delete an existing option.

Click the **Add** or **Edit** button to add a new option to the existing list of fields in the Table. The fields in table are the same as the fields in the top level of the ACE object (see [Edit Fields and Field Properties](#) for instructions on modifying field properties).

7.4 Resolving Template Problems

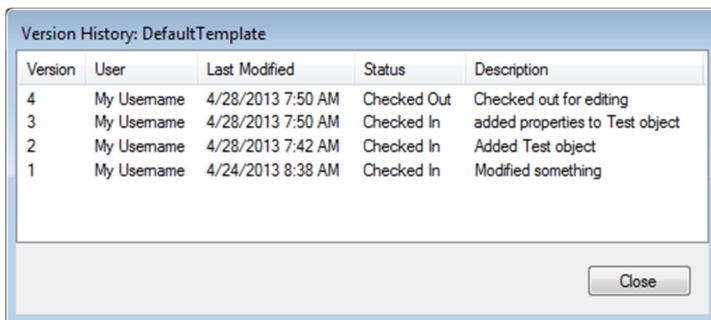
[TBD]

8 Using ACE in Repository Mode

The ACE programs include a Repository menu for working with configurations and templates in the ACE repository. See [Set Up ACE Repository Server](#) on page 21 for more information on the repository installation.

8.1 Version History

While working in repository mode, there is a Version History that shows the history of changes made to the file in the repository, as well as its current status (“Checked In” or “Checked Out”). An example is shown below.



Version	User	Last Modified	Status	Description
4	My Username	4/28/2013 7:50 AM	Checked Out	Checked out for editing
3	My Username	4/28/2013 7:50 AM	Checked In	added properties to Test object
2	My Username	4/28/2013 7:42 AM	Checked In	Added Test object
1	My Username	4/24/2013 8:38 AM	Checked In	Modified something

To show the version history, connect to the repository and select the **File | Open** menu. Select a configuration or template, and click the **Version History** button. After a configuration or template is open in the editor, select the **Repository | Version History** menu to show the history of the current file.

8.2 Open File from Repository

When the Configuration Editor or Template Editor are connected to the repository, the **File | Open** menu will provide only a list of configurations or templates that exist in the repository. If the file is not checked out to the current user, the file will be opened from the repository in Read-Only mode. The Open menu shows whether the file is currently “Checked Out”, and if so, which user has the file checked out.

If a user opens a file that is checked out to another user, the file may be viewed but not modified, unless the check-out is overridden. It is recommended to contact the other user and ensure that they finish their changes and check the file back into the repository so that other users can access the latest copy of the file. When a user checks in their latest changes, you should close the file and re-open it, to ensure that you are opening the latest “Checked In” version, before making any changes.

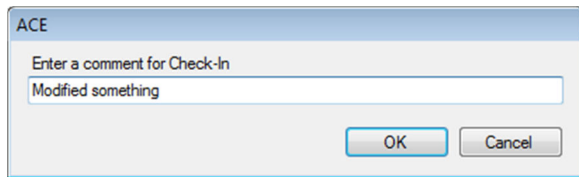
8.3 Check In / Check Out From Repository

If a configuration or template is not already checked out to another user, you can open the file from repository and select the menu option **Repository | Check Out**. This will essentially lock the configuration for your use, and allow you to save changes to the configuration. Only one user can have any given configuration checked out.

NOTE: When checking out a file from the repository, any existing file in the local working directory with that name will be overwritten without notification. If you have local changes that are later than the latest repository copy, see [Saving Repository Files](#) on page 61 for special rules on checking in those changes, to avoid losing your work.

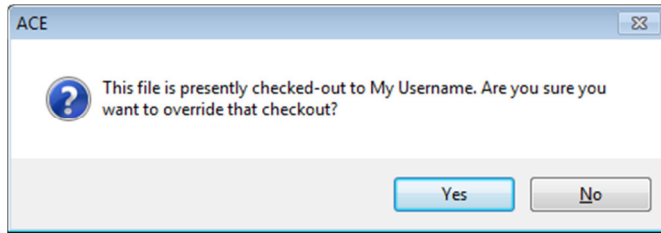
Select the **Repository | Undo Check-out** option if you have checked out a configuration or template and subsequently wish to revert back to the repository version of the file. This will cause any local changes to be lost, whether or not they have been saved to the local working directory, and will free the file to be modified by other users.

After modifying a configuration or template, select **Repository | Check In** to create a new version in the repository. Enter a comment that describes the changes that were made. The comment is entered into the Version History of this configuration. After checking in, the title bar of the configuration in the editor window indicates that this configuration file is “READ-ONLY”.



Important Note: If changes have been made to a configuration while disconnected from the repository, special care needs to be taken while reconnecting to the repository to avoid losing the offline changes. See [Saving Repository Files](#) on page 61 for more details on saving changes in repository mode.

If another user has a configuration or template checked out, the repository doesn't allow other users to check out at the same time. However, this can be overridden by selecting the **Repository | Override Check-out** menu option.



However, be aware that this may lead to a conflict, because if another person has checked out a configuration and made changes, one or the other set of changes may be lost if both people check in their version of the file independently.

The override check-out feature is really intended to be used only as a last resort. An example would be that a user has checked out a configuration and made some changes. That user becomes unavailable for some reason, leaving the configuration checked out. Another user can take control of the configuration by using override check-out to revert any changes made by the first user.

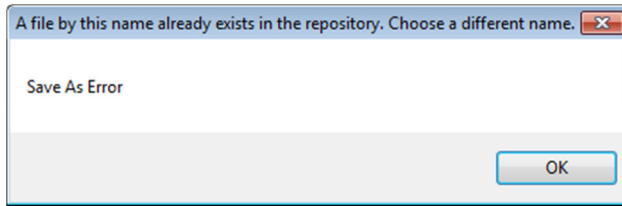
8.4 Saving Repository Files

After a user has checked out a configuration or template, they may make changes to the local copy of the file. A user may also open a local configuration file or a default (or other) local template file outside the repository. A third scenario involves opening a file and making changes while disconnected from the repository, then connecting to the repository to save changes.

NOTE: There are some important rules that govern where and how the configuration or template file is saved, depending on the circumstances.

While connected to repository:

- **Save** – Changes to repository files are saved to local working directory. Files opened from another local directory will be saved in their original location. A copy of the file is also saved on the repository server (but not checked in). If the file didn't already exist in the repository, it is added to the repository collection with version number 1 and is automatically checked out to the current user.
- **Save As** – User is prompted for a new file name, and the file is saved both in the local working directory and the repository. If the file didn't already exist in the repository, it is added to the repository collection, given a version 1, and is automatically marked as checked out to the current user. If the file does already exist in the repository, an error message warns that this operation is not allowed.



While disconnected from the repository, changes are only saved locally, not in the repository:

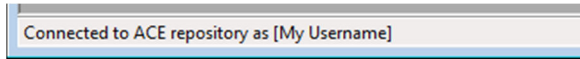
- **Save** – Changes to repository files are saved to the local working directory. Files opened from another local directory will be saved in their original location. Note that the latest repository copy and the local working copy are now out of sync.
- **Save As** – User is prompted for a new file name and a directory location in which to save the configuration.

If changes have been made offline, **special care should be taken** when checking into the repository to avoid losing your work. Please pay special attention to the following instructions:

- **Check Out** – Check out a configuration or template from the repository to the local working directory.
- **Disconnect** – For instance, a technician may need to disconnect from the network and/or the repository in order to visit a field location and configure a Director.
- **Make Local Changes** – If the technician needs to make changes while disconnected from the repository, those changes are saved only in the local working directory.
- **Open Locally Changed Files** – Before reconnecting to the repository, open the Configuration Editor and/or Template Editor in non-repository mode, and open all files for which changes were made offline.
Or, connect to the repository and use **File | Open Local File** (in Configuration Editor) or **File | Open Default Template** and browse to the changed template (in Template Editor). In either case, do not open the file from the repository (i.e., from the Open menu showing the list of file version numbers).
- **Save and Check In Changes** – With the locally-modified file(s) open, connect to the repository and select **File | Save**. This will save a copy to both the local file location and to the repository. This ensures that the latest copy in the repository is overwritten with your local changes. Then the file can be checked in to the repository as normal.

8.5 Switch to a Different User or Repository

While connected to the repository, the status bar at the bottom of the Configuration Editor or Template Editor window shows that you are “Connected to ACE repository as *[your username]*”, where *[your username]* is the name of the User Account that you logged in as.



Closing and re-opening the ACE programs will prompt to log into the repository, and a new user account can be chosen at that time.

The Configuration Editor also allows the user to switch to a different user account or repository in two steps: Select the **Tools | ACE Settings** menu, and disable the “*Work connect to ACE repository*” checkbox, and click **OK**. Then return to the ACE Settings menu and re-enable the checkbox. This will prompt again for the repository location and user account.

9 Deploying ACE Configurations

In order to use the configurations created in the ACE Editor, they must be deployed. A configuration is deployed to a Director by uploading the binary files (*.uff, etc.) and restarting the Director. This section describes several methods of configuration deployment.

9.1 Loading Single Director Configuration

The Director may be configured either via a network connection or using the local Serial MMI serial port (typically on the COM1 port). The following two sections describe both procedures for storing the configurations to a single Director.

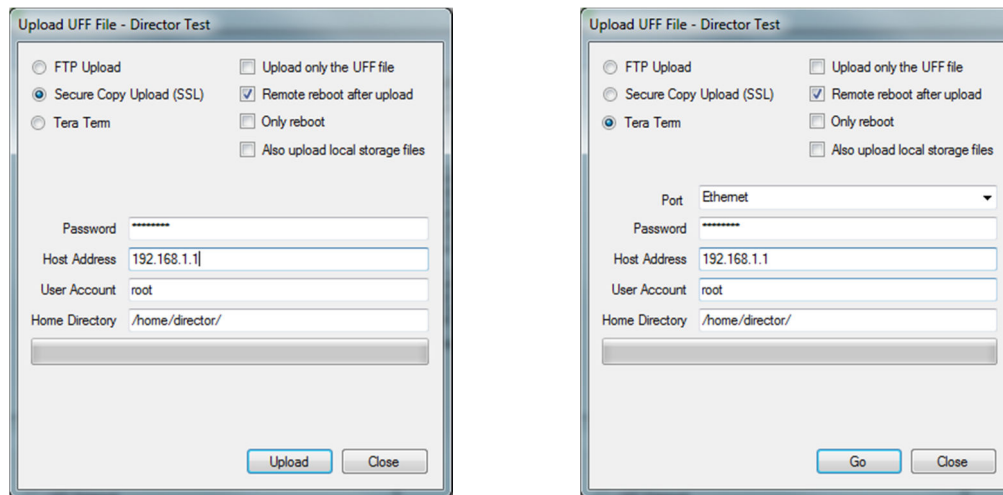
9.1.1 Network Upload via ACE Editor

In order to upload the Director configuration across a network, you need to know the Director's network address.

In the ACE Editor, select **File | Upload UFF to Device**. Enter the Host Address to the IP address of the Director. Enter the User Account under which the configuration should be uploaded, typically 'root', and the password for that user account. Enter the Home Directory into which the configuration should be uploaded.

After the configuration is uploaded, ACE can also upload a file called 'reset', which will cause the Director to automatically reboot, thus implementing the new configuration settings. Select the "Remote reboot after upload" checkbox to cause this reboot.

Typically, the Director uses secure copy (SFTP/SCP) to transfer files. However, some Directors may also support standard (unsecure) FTP. Select the “Secure Copy Upload (SSL)” option, or “FTP Upload” if you know that the Director supports it.

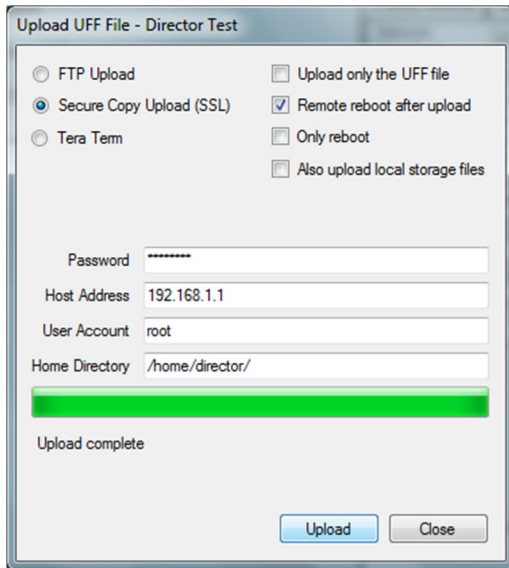


The “Only reboot” option sends a signal to the Director to reboot. No UFF or other files will be uploaded.

The “Also upload local storage files” option will send local storage files to one Director unit along with the UFF file.

Tera Term is a terminal server client that uses the serial connection to upload the UFF file, other files, and/or reboot the Director. Tera Term can also access the Director through the Ethernet connection which will launch a terminal window where a user can diagnose the Director.

Once these options have been configured, click the **Upload** button to transfer the configuration to the Director. If successful, the completed status will be displayed. Otherwise, an error condition may be indicated.



Below is a screen capture showing an example of the Home Directory after files are downloaded, and before the Director automatically reboots. The files newly downloaded by the ACE Editor are “Apex.uff” and “reset”.

```

10.11.30.90 - PuTTY
root@Director root#
root@Director root# cd /home/director/
root@Director director# ls -l
-rw-r--r-- 1 root root 329 Apr 23 12:13 Apex.uff
-rw-r--r-- 1 root root 0 Dec 31 1969 Director-Z.32
-rw-r--r-- 1 root root 0 Dec 31 1969 Director.nam
lrwxrwxrwx 1 root root 33 Dec 31 1969 director.uff.run -> /usr/director/config/director.uff
lrwxrwxrwx 1 root root 28 Dec 31 1969 ls15000000.run -> /usr/director/bin/ls15000000
lrwxrwxrwx 1 root root 28 Dec 31 1969 ls15000000.run -> /usr/director/bin/ls15000000
-rw-r--r-- 1 root root 329 Apr 23 12:14 reset
root@Director director#

```

After a reboot, the “reset” file is removed, and the “Apex.uff” file is moved to the /usr/director/config directory. A shortcut is created in the /home/director location, which points to the current configuration file.

```

10.11.30.90 - PuTTY
root@Director director# cd /home/director/
root@Director director# ls -l
lrwxrwxrwx 1 root root 29 Apr 23 15:49 Apex.uff.run -> /usr/director/config/Apex.uff
-rw-r--r-- 1 root root 0 Dec 31 1969 Director-Z.32
-rw-r--r-- 1 root root 0 Apr 23 15:49 Director.nam
lrwxrwxrwx 1 root root 28 Apr 23 15:49 ls15000000.run -> /usr/director/bin/ls15000000
lrwxrwxrwx 1 root root 28 Apr 23 15:49 ls15000000.run -> /usr/director/bin/ls15000000
root@Director director#

```

9.1.2 Serial Upload via YMODEM

A Director may also be configured using its Serial MMI port. To do this, the Director must be configured to use Serial MMI (typically on COM1). The *Director Getting Started Guide* describes the use of the PuTTY program for establishing serial console access to the Director; however, as of this writing, PuTTY does not support the YMODEM file transfer protocol, so it cannot be used for this procedure.

To transfer a file to the Director, you will need a serial terminal program that supports the YMODEM file transfer program, such as Tera Term (available free via the Internet), or HyperTerminal (only included through Windows Vista, not Windows 7), or another serial transfer program. The installation of these programs is not described in this manual. Tera Term is installed with the ACE software in the ACE directory. For easier access to Tera Term, a shortcut may be placed on the desktop of the operators computer.

In the ACE Editor, open the configuration for this Director. Select the **File | Upload UFF to Device** menu option. This causes the ACE Editor to create the .uff file in the "ACE" folder of the local working directory. Then close the menu.

Using the serial terminal program, connect to the SerialMMI port of the Director at the defined baud rate (typically 115,200, and log into the Director using the User account and password defined in its ACE system configuration object. Enter '2' for the Directory Services menu option.

```
===== Main Menu =====  
ElecSys(V:5.6.2012-05-21-1300) Tue Apr 23 22:42:44 2013  
Director : 1 @ 10.11.30.90  
-----  
1) System      Services  
2) Directory   Services  
3) Diagnostics Services  
  
99 Log Off  
  
Make selection: 2
```

```

===== Directory Services =====
ElecSys(V:5.6.2012-05-21-1300) Tue Apr 23 22:43:06 2013
Director : 1 @ 10.11.30.90
-----
1) Executables          2) Current Configs
3) Download Directory   4) Put File to Remote
5) Get File From Remote 6) Rename File
7) Delete File in Remote 8) Delete IsaGraf Files
9) View Zombie.log      10) IsaGraf App Info
11) Restore Previous UFF 12) View UFF Processing Details
13) View Load-Store File 14) View kernel or message logs
15) MQTT XML Information

```

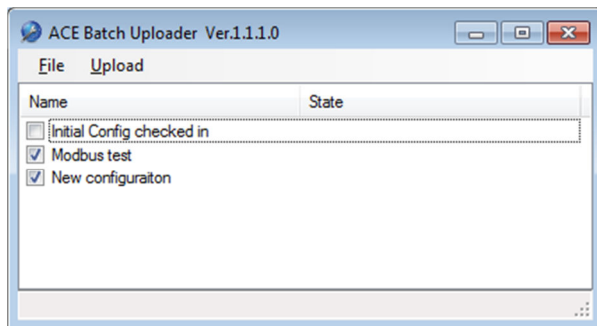
Enter '4' for the 'Put File to Remote' option. Then immediately initiate a YMODEM (or YMODEM-G) file transfer, and browse to the .uff file created in the local working directory above. The file transfer must be started within 40 seconds or else the operation will timeout. Enter option '4' again to try again if you miss it the first time.

After downloading the ACE configuration file, you can reboot the Director by power cycling it, or by returning to the Main Menu and enter '1' for System Services and '1' again for Unit Restart.

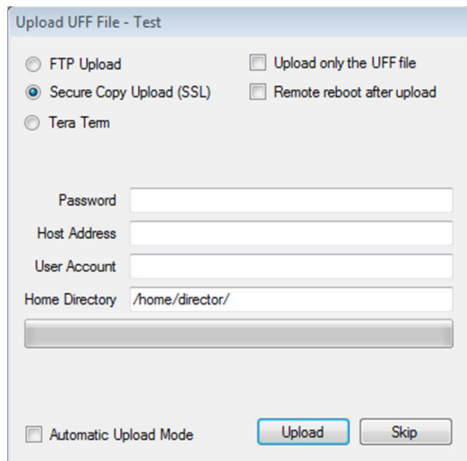
9.1.3 Using the ACE Uploader for Multiple Directors

If you have many configurations to upload to Directors all at once, you can use the ACE Batch Uploader. Run the program from the Windows Start menu, and connect to the repository if you are using the repository.

Select the checkboxes for the configurations to upload from the list of available configurations.



Select the **Upload | Upload UFF Files** menu to begin the configuration upload. For each configuration, enter the connection details of the Director. This requires knowledge of its current network IP address and login credentials.



Enter the Host Address to the IP address of the Director. Enter the User Account under which the configuration should be uploaded, typically 'root', and the password for that user account. Enter the Home Directory into which the configuration should be uploaded. Select the "Remote reboot after upload" checkbox to cause this reboot.

After the information has been entered, click the **Upload** button. If successful, the ACE Batch Uploader will move to the next selected configuration, and the process is repeated. Click the **Skip** button to skip a Director if you can't gain access to it or need to skip for some other reason.

If the check box Automatic Upload Mode is not checked, then the user must repeatedly click the **Upload** and **Skip** buttons for each configuration. Check this check box and click the Upload button once, and the program will automatically process all remaining selected configurations.

9.2 Using HCP Configuration

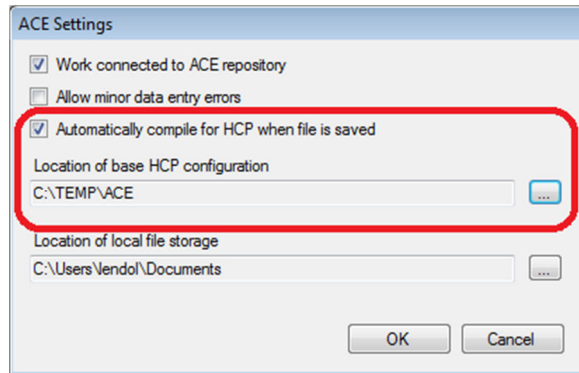
The Elecsys HCP software products require that the configuration files be compiled into a special organization of binary files and directory folders. These files and folders must reside on the computer that hosts the HCP application. Since the ACE Configuration Editor will compile these binary files and folders, the ACE Configuration Editor must also be installed on this HCP computer.

Compiling of the configuration files into a format usable by the HCP can be accomplished in either of two methods: automatic during file save, or manually.

9.2.1 Compile for HCP

Automatic compilation is enabled by setting the appropriate application settings in the ACE Configuration Editor. Refer to [ACE Program Settings](#) for details. The following screen capture shows the two settings that control this automatic compile process. Select the option *Automatically compile for HCP when file is saved*. And use the button to choose the base directory to indicate where to

store the compiled binary files. With these settings enabled, the ACE Configuration Editor will automatically generate the necessary folders and files whenever a configuration file is saved.



To manually generate the compiled binary files needed by the HCP, select the **File | Compile for HCP** menu item. Note that this menu item is only visible when a configuration file is open. This menu item will not modify the configuration, or save the configuration to the repository. This menu item will only compile the configuration into binary files usable by the Elecsys HCP products.