Important user information

Read this document and the documents listed in the additional resources section about installation, configuration, and operation of this equipment before you install, configure, operate, or maintain this product. Users are required to familiarize themselves with installation and wiring instructions in addition to requirements of all applicable codes, laws, and standards.

Activities including installation, adjustments, putting into service, use, assembly, disassembly, and maintenance are required to be carried out by suitably trained personnel in accordance with applicable code of practice. If this equipment is used in a manner not specified by the manufacturer, the protection provided by the equipment may be impaired.

In no event will Rockwell Automation, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Rockwell Automation, Inc. with respect to use of information, circuits, equipment, or software described in this manual.

Reproduction of the contents of this manual, in whole or in part, without written permission of Rockwell Automation, Inc., is prohibited.

Throughout this manual, when necessary, we use notes to make you aware of safety considerations.

---

**WARNING:** Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.

---

**ATTENTION:** Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss. Attentions help you identify a hazard, avoid a hazard, and recognize the consequence.

---

**Important:** Identifies information that is critical for successful application and understanding of the product.

---

Labels may also be on or inside the equipment to provide specific precautions.

---

**SHOCK HAZARD:** Labels may be on or inside the equipment, for example, a drive or motor, to alert people that dangerous voltage may be present.

---

**BURN HAZARD:** Labels may be on or inside the equipment, for example, a drive or motor, to alert people that surfaces may reach dangerous temperatures.

---

**ARC FLASH HAZARD:** Labels may be on or inside the equipment, for example, a motor control center, to alert people to potential Arc Flash. Arc Flash will cause severe injury or death. Wear proper Personal Protective Equipment (PPE). Follow ALL Regulatory requirements for safe work practices and for Personal Protective Equipment (PPE).
# Table of contents

## Chapter 1

### Application Code Manager

- Application Code Manager Overview............................................................... 7
- Design process........................................................................................................ 8
- Library objects..................................................................................................... 12
- Templates............................................................................................................. 13
- Schedules.............................................................................................................. 13
- Design Automation Concept................................................................................ 13
- Create a central ACM database........................................................................... 14
- Upgrading a central ACM database.................................................................... 15
- Initial configuration of Application Code Manager............................................. 16
- Upgrade the Application Code Manager application......................................... 17
- Navigate the Application Code Manager user interface.................................... 19

## Chapter 2

### Connect to an ACM database

- Connect to an ACM database............................................................................... 23
  - Connection Properties settings.......................................................................... 24
  - Advanced Properties settings........................................................................... 25

## Chapter 3

### Project Development

- Project Development............................................................................................ 31
  - Project commands.............................................................................................. 31
  - Create a new project......................................................................................... 32
  - Create a new project from an existing project.................................................. 33
  - Add a Historian object...................................................................................... 34
  - Add a Historian sub-object................................................................................ 34
  - Historian Point Type parameters...................................................................... 35
  - Generate a Historian object.............................................................................. 37
  - Add an HMI object............................................................................................ 38
  - Add a Display sub-object.................................................................................. 38
  - HMI Display parameters.................................................................................... 39
  - Generate an HMI object.................................................................................... 40
  - Add an Alarm Group sub-object....................................................................... 41
  - FTAE Alarm Group parameters........................................................................ 42
  - Find an object.................................................................................................... 42
  - Update a project library.................................................................................... 43
  - Extract attached files....................................................................................... 44
  - Indications......................................................................................................... 45
# Table of contents

## Chapter 4

**Registered Libraries**
- Registered Libraries ................................................................. 47
  - Register an ACM library object ............................................... 48
  - Reconstitute an ACD file ......................................................... 49
  - View registered library usage .................................................. 50
  - Share libraries, templates, and schedules ................................. 51
  - Local library and template file location .................................... 51

## Chapter 5

**Configure controllers**
- Configure controllers ................................................................. 53
  - Generate a controller file .......................................................... 54
  - Add a new controller ................................................................. 55
  - Move or copy a controller to another project ............................ 56
  - Select Destination Project dialog box ....................................... 57
  - Add a controller from an ACD or L5X file ............................... 57
  - Update using an ACD or L5X file .............................................. 58
  - Detach from ACD/L5X files ..................................................... 60
  - Contribute instances ............................................................... 60
  - Merge controllers ................................................................. 60
  - Add a hardware module .......................................................... 61
  - Delete a hardware module ...................................................... 62
  - Use Copy and Paste Special .................................................... 62
  - Add a new instance ............................................................... 63
  - Add a new software object to a controller .............................. 64
  - Generate a partial program .................................................... 64
  - Generate a partial routine ...................................................... 65
  - Object Configuration Wizard .................................................. 66
  - Auto Create Linked Objects ................................................... 67

## Chapter 6

**Use ACM Tools**
- Use ACM Tools ........................................................................... 69
  - Import Export Manager .......................................................... 69
    - Import a schedule ............................................................... 71
    - Import Export Manager Import tab settings ......................... 72
    - Export a schedule ............................................................. 73
    - Import Export Manager Export tab settings ....................... 74
    - Compare a project to a saved schedule ............................... 75
    - Import Export Manager Compare Tab settings .................... 75
    - Create a new schedule template ........................................ 76
  - Database Manager .................................................................. 79
    - Create an ACM database ................................................... 79
    - Upgrade an ACM database ................................................ 80
    - Backup an ACM database .................................................. 81
<table>
<thead>
<tr>
<th>Table of contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>Restore an ACM database ....................................................................................</td>
</tr>
<tr>
<td>Delete an ACM database .....................................................................................</td>
</tr>
<tr>
<td>Database Manager settings ................................................................................</td>
</tr>
<tr>
<td>Log Debug Information .......................................................................................</td>
</tr>
<tr>
<td>Log File Viewer ...................................................................................................</td>
</tr>
<tr>
<td>ACM Default Settings ..........................................................................................</td>
</tr>
<tr>
<td>Add ACM library content to an existing ACD project ........................................</td>
</tr>
</tbody>
</table>

**Chapter 7**

Reports .................................................................................................................. 89
- Report command reference ................................................................................ 90
- Generate a report ............................................................................................... 91

**Chapter 8**

ACM Console ......................................................................................................... 93
- Open the ACM Console ....................................................................................... 94
- List all commands .............................................................................................. 94
- Generate a limited list of commands ............................................................... 94
- Generate detailed command information ....................................................... 95
- Console scripts ................................................................................................. 95
  - Create an Application Code Manager Console script .................................. 96
  - Run an Application Code Manager Console script ...................................... 96
  - Extended scripts ......................................................................................... 97

Legal Notices ....................................................................................................... 103
Application Code Manager

Application Code Manager Overview

Studio 5000® Application Code Manager is a tool that enables more efficient project development with libraries of reusable code. Application Code Manager creates modular objects with customizable configuration parameters using the reusable content. Application Code Manager can also create the associated visualization, historical and alarming elements for a project.

Use Application Code Manager to:

- Create databases
- Register libraries
- Create projects
- Manage objects

Application Code Manager includes an additional command-line tool, the ACM Console. Use the ACM Console to perform the following actions:

- Edit Parameters
- Export All Projects
- Export Libraries by Attribute
- Generate Controller (as an L5X or ACD file)
- Import Project
- Publish Library

Before starting a project, become familiar with the basic concepts used in Application Code Manager; the design process, the different library objects, the available templates, the use of schedules, and the design outputs from Application Code Manager.

Activation

Activating Application Code Manager provides access to all Application Code Manager features.

When not activated Application Code Manager has the following limitations:
• Application Code Manager can only connect to a local database instance.
• Application Code Manager can only support a single controller per project. Attempting to import projects with multiple controllers will result in an error.

See also

Design process on page 8
Library objects on page 12
Templates on page 13
Schedules on page 13
Initial configuration of Application Code Manager on page 16

Design process

The Application Code Manager (ACM) design process introduces a modular, object-based approach to the creation of ACD controller code, FactoryTalk® View SE/ME display content, FactoryTalk Historian Tag and FactoryTalk Alarms and Events (FTAE) import configuration.

The Studio 5000 ACM design process separates function and configuration into two separate layers of data, and divides the design process into two distinct workflows, library management and project execution.

The design process involves a suite of applications:

• The Studio 5000 Logix Designer® application
• The Library Designer
• The Library Object Manager application
• The Application Code Manager application
• FactoryTalk View Studio

Library management workflow: Studio 5000 Logix Designer

The library management workflow begins when a specific instance of ACD controller code is created in the Logix Designer application. The specific instance is a single project containing a single controller. The project includes a logical structure containing these Logix objects:

• Controller Tags
• Tasks
• Motion Groups
• Add-On Instructions
• Data Types
• Trends
• I/O Configurations

Each Logix object has an internal hierarchy of elements. Example: a task may contain one or more programs, each of which may contain one or more routines.

Every project has one controller. There may be one, many, or no instances of any type of Logix object in the project when the specific instance is created. This single instance of controller code is saved to an ACD file.

Traditionally, controller code was designed and configured for a specific project. In the library management workflow, content is not designed for a specific project, but to provide a widely applicable set of functions known as project components that are then used to create library objects. Each library object is an independent functional entity that can be easily configured to meet a wide range of applications and can be used in many projects.

**Library Designer**

Use the Library Designer to assign the project, the controller, and any of the Logix objects to one or many library objects. Each library object defines a set of functions, capabilities, and connections. Example: those that support function of the valve, motor, and controller modules. Rather than being tied to one application, library objects can be configured to meet the needs of multiple applications. The Library Designer allows the publishing of a library directly into an ACM database. Options include the ability to specify the location where the library will be published in the ACM database, and the ability to specify the status of the library, either Published or Pending.

Custom properties called "Decorations" can be added to a library object using the Library Designer. Decorations include parameters, sub-objects, functions, substitutions, and external references. Decoration lets the library object be configured when it is implemented in a project in the ACM application.

Logix objects can be restricted to a single library object or assigned to multiple library objects, each with a different set of decorations. A library object can contain a single Logix object, or a Logix object can be added as an element of a more complex library object. Example: a P_Alarm Add-On Instruction can be assigned to a valve library object and can also be an element of a Motor or Pump library object.

Each ACD file can support multiple projects, controller libraries, and library objects. The ACD is not required to contain a project or controller library. While
Decoration is stored as part of the ACD file, it is treated as a separate layer of information from the base controller code and does not affect code execution.

Decoration controls how the library object is instantiated, including configurations such as naming, tag values, conditional inclusion, and connections to other library objects. One or many distinct instances of a library object can be instantiated within an ACM project and each instance can be separately configured. Using Library Designer each Logix object can be published directly to the ACM database or to a file in HSL4 format.

**Library Object Manager**

The library management workflow continues by opening the decorated ACD file in the Library Object Manager application. The Library Object Manager application can be used to publish each library object, either directly to the ACM database or to a file in HSL4 format. HSL4 files can be distributed individually or as part of a repository.

Use Library Object Manager to add HMI displays (FactoryTalk View SE/ME), FactoryTalk Alarms and Events configurations, and Historian (FactoryTalk Historian SE) components to the library object. This can only be done after the library object has been published from the ACD file to a folder or ACM database. The features added in the Library Object Manager application are saved to the individual HSL4 file or database entry for the library object and are not saved to the original ACD file.

Each library object file saved from the Library Object Manager application is classified within a four-level hierarchy:

**Solution -&gt; Library Type -&gt; Category -&gt; Catalog Number**

Example: an analog input module might be classified as:

- **Solution: (RA-LIB) ACM**
  Solution will, in most cases, name the library object repository for the library object.

- **Library Type: Modules**
  Library Type is a general classification for the library object based on its function, such as module, control module or design pattern.

- **Category: Analog**
  Category is a more specific classification for the library object, based on its function.

- **Catalog Number: 1734-IE2C/C**
  The specific identifier for the library object.
Each library object file must have a distinct version number per solution. Just as the same Logix object can be used to create one or many library objects within the Library Designer, the same library object can be used to create one or many distinct library object files (versions) within the Library Object Manager application.

Library objects can be quickly distributed, then registered into and configured for multiple Projects in multiple locations. Library objects are available to any project that requires the functionality the library object provides.

Using Library Object Manager new library objects can be created and distributed rapidly to meet the needs of specific applications.

FactoryTalk View Studio

Use FactoryTalk View Studio to create Site Edition (SE) and Machine Edition (ME) graphic displays. When the graphic displays are exported to XML they are called “Symbol objects.” The XML files can be imported into the Library Object Manager application and the graphic displays added as non-Logix content to library objects.

The Project Execution Workflow: Application Code Manager

In the Application Code Manager application, library objects become the building blocks used to rapidly create and deploy projects.

Execution is simply a matter of registering, adding, and configuring the library objects. Projects can be completed without requiring high-end programming support.

In the project execution workflow, library objects are selected in the ACM application and then the library object parameters are configured to meet the requirements of the current application. The workflow is complete when the Project to ACD controller code is created.

During the project, new library objects can be created using Library Designer, library objects from previous projects can be reused, or library objects can be shared from other databases.

After the project is complete, it can be used to create new library objects so that future projects can use the solutions developed.

See also

Application Code Manager Overview on page 7
Project Development on page 31
Library objects

A library object is the class definition of an object, it can contain links to other libraries. A library object is instantiated. When instantiating a library, all the linked libraries can be instantiated to new objects, or can continue to link to an existing instantiated object in the library.

Individual library object files (HSL4) are XML formatted and registered in the ACM database. A library object typically defines parameters, subclasses, user interface contents, and portions of controller code (example: Logix) and HMI code (example: FactoryTalk View SE/ME).

Library objects contain controller code, as well as decoration. Decoration is a set of custom properties applied to a library object using the Library Designer. Decoration can be inherited from a library object that is higher in scope such as the Controller and Project object. Decoration that is applied to a library object is inherited by, or available to, all elements that are contained within the library object. Decoration can also be applied directly to an element, overriding inheritance from the library object and from library objects of higher scope.

ACM provides the option to include Project Data during the controller code generation. If this option is selected all instances with their parameter values, as well as all libraries (zipped) will be included as part of the Controller’s Custom Properties.

Every element created and included by ACM will also have project data custom properties which includes information about the instance and the library that owns or created the element.

See also

Application Code Manager Overview on page 7

Registered Libraries on page 47
Templates

A template defines the static content and format of design output (example: a FactoryTalk View display). A template is not a class definition. A template is not instantiated. Templates have a variety of formats (example: xml, csv, docx, xlsx) and are stored in the ACM program folder or an individual user folder.

Before a template can be used the initial configuration of ACM must be completed, the local library must be registered, and the template installed.

See also

Application Code Manager Overview on page 7
Initial configuration of Application Code Manager on page 16
Local library and template file location on page 51
Register an ACM library object on page 48

Schedules

Use schedules to display or edit project data, typically parameter values.

Read-only schedules called "views" are temporarily generated for certain ACM reports (example: the I/O Schedule report).

The Import Export Manager tool can be used to export a schedule to a Microsoft® Excel® spreadsheet. Exporting a schedule to a spreadsheet can be useful for:

- Bulk additions, duplication, and changes
- Transferring project contents
- Snapshots
- Backups
- Version comparison

Import the schedule spreadsheet to ACM using the Import Export Manager tool.

See also

Application Code Manager Overview on page 7
Import Export Manager on page 69

Design Automation Concept

The project design outputs are generated automatically by ACM. The objects (instances) and parameter values, stored in the ACM database, are combined with various templates to create the following design outputs:

- Logix
Create a central ACM database

When planning an ACM deployment if there are multiple people collaborating on projects make sure to select a computer to use as the ACM database server that can be accessed by all users in the project. This can be a standard computer that does not belong to a particular user or a project computer. The computer must always be turned on and available. Microsoft SQL Server 2012 is the only software required on the shared computer.

Note: The ability to connect to a remote database is only available if the product has a Standard activation license and is not available in Lite mode.

Install SQL Server 2012 via the ACM installation media. Select only SQL Server 2012 when presented with the selection of install options.

When configuring the SQL Server note the following considerations when supporting multiple user connections to the database:

1. **Add Users**: Normally ACM uses Windows Authentication to connect to the ACM database. When the ACM database is located in a remote computer, local users must be created using the SQL Server Management Studio. Create one user account that is shared by all the project collaborators or add an individual user for each collaborator. Only assign users to the ACM database after the ACM database is created.

If the SQL user is not a sysadmin, creation of the database for the first time is a multi-step process:

a. **Create users in SQL**.
   - **Minimum Server Roles** should be:
     - dbcreator
     - public
   - **Permissions on Securables** should be:
     - Connect SQL
     - Control Server
• View any database (optional, but if not, the database needs to be mapped to this user in SQL)

b. Launch the database manager in ACM and create the database using the new user.

Creation of this database must be done from ACM and cannot be done manually in Microsoft SQL Server.

c. In order for the user to connect and update the database, add the newly created database to the User Mapping in SQL and set the memberships on the database to:

• db_owner
• db_datareader
• db_datawriter
• public

d. In order for the user to back-up the database, add the newly created database to the User Mapping in SQL and set the memberships on the database to:

• db_backupoperator

2. Record Database Connection Information: Record the SQL Server computer name and/or computer IP address, the SQL Server instance name, the SQL Server authentication (username and password), and the ACM database name. This information is required by ACM users attempting to connect to the central ACM database.

See also

Initial configuration of Application Code Manager on page 16

Connect to an ACM database on page 23

Upgrading a central ACM database

After installing a new version of ACM, the database may sometimes require upgrading due to the adding of new tables or fields. You will be notified to upgrade the database if applicable.

The failure of the upgrade process could be related to you having insufficient permissions of the central server to execute the upgrade commands. The upgrade process could also fail due to the requirement of additional prerequisite software needed on the SQL server host machine.

If upgrading ACM from v2.xx to v3.xx then this error is likely due to the x64 versions of the following components not being installed:

• Microsoft System CLR Types for SQL Server 2012 (x64)
• Microsoft SQL Server 2012 Management Objects (x64)

It is recommended that when installing a new version of ACM on a client machine on which the same version of ACM is also installed on the SQL server host machine to ensure all prerequisite software is also installed.

See also

Create a central ACM database on page 14

Connect to an ACM database on page 23

Initial configuration of Application Code Manager

Before using Application Code Manager, initial configuration must be completed. In this procedure, a database is created, the database connection method is specified, and the default ACM libraries are registered.

Tip: If the SQL Server installation option was selected, this procedure was completed automatically during installation.

To complete the initial configuration of Application Code Manager

1. Open Application Code Manager. On the main menu, click Tools > Database Manager.

2. In Server Name, select the computer name and SQL Server instance from the drop down list or type a computer name and SQL Server instance in the following format:

   <Computer Name> \ <SQL Server Instance>

3. In Log on to the server, type the User name and Password to use to authenticate the connection to the SQL Server, then click Connect.

   Status updates to Connected.

4. In Specify the database, type a unique name for the ACM database.

5. In Actions, select Create database and then click Execute Task.

   Once the database is created an Action completed successfully message displays. Click OK.

6. Click Close to close Database Manager.
Tip:
The Application Code Manager title bar displays the application icon and application name followed by the computer name, SQL Server instance, and database name formatted as:

<Computer Name> \ <SQL Server Instance>.<DataBase Name>

If Application Code Manager is not able to connect to a database the title bar displays (Not Connected).

1. In Application Code Manager, right click Registered Libraries and then select Register.

The Open dialog box displays the default library file location.

C:\User\Public\Public Documents\Rockwell Automation\Studio 5000\Libraries\Application Code Manager

2. Open the (RA-LIB) ACM folder. Select all the objects in the folder and then click Open.

3. The Libraries Registration window opens and displays the status of the library registration process.

Review any errors or warnings that were encountered.

Tip: The results of the registration process are also saved to a log file. Click Show Log File to view the contents of the Log file.

4. Click Finish to close the Libraries Registration window.

The registered libraries display Application Code Manager under Registered Libraries.

See also

Create a central ACM database on page 14

Database Manager settings on page 84

Upgrade the Application Code Manager application

When upgrading the Application Code Manager application, consider updating the libraries used in the ACM projects for a particular ACM database and obsoleting the previous version libraries.

Important: Always back up the current ACM database before performing an upgrade.

Prerequisites

1. Back up database using the Database Manger.
2. Export all projects using the **Import Export Manager**.

3. Install the newer version of Application Code Manager.

**To upgrade the Application Code Manager application**

1. Open Application Code Manager. A message appears with a prompt to upgrade the database. Click **OK**. The **Database Manager** displays.

2. Select **Upgrade database** in the **Actions - Tasks** area and then click **Execute Task**.

   Click Close to close **Database Manager**.

3. Right click **Registered Libraries** and then select **Register**.

   The **Open** dialog box displays the default library file location.

   C:\User\Public\Documents\Studio 5000\Libraries\Application Code Manager

4. Open each folder. Select all the objects in the folder and then click **Open**.

5. The **Libraries Registration** window opens and displays the status of the library registration process.

   Review any errors or warnings that were encountered.

   **Tip:** Results of the registration process are also saved to a log file. Click **Show Log File** to view the contents of the Log file.

6. Click **Finish** to close the **Libraries Registration** window.

   The registered libraries display in Application Code Manager under **Registered Libraries**.

7. Import previously exported projects using the **Import Export Manager**.
Tip: When importing projects, a compatible library must be registered in the ACM Database. Library compatibility rules:

- The catalog number of the registered library must match the catalog number in the project schedule, and a library with a revision greater than or equal to the revision in the schedule must be registered.
- If a library with the same revision is registered, the library with the same revision is used.
- If a library with the same revision is not registered and a library with a greater revision is registered, the newest library is used.

See also

Import Export Manager on page 69

Navigate the Application Code Manager user interface

Application Code Manager is composed of a menu bar, a toolbar, and several different panes that navigate through different objects. The objects (instances) contained in a project are arranged in a hierarchy. There are three ways to view the hierarchy, which are accessed by clicking one of the three panes (System View, Controller Preview, and Class View). Hide or float the panes so that they can be arranged according to preference.

This table provides an overview of the user interface panes.
<table>
<thead>
<tr>
<th>Pane</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>System View</td>
<td>A pane that contains a tree control that enables navigation through the different components of an ACM project. The System View pane displays branches for the objects contained in the project including FactoryTalk Historian Server objects, FactoryTalk View SE/ME Server objects, Libraries used in the project, and Controller objects.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Historian branch</strong>&lt;br&gt;The Historian branch contains all Historian ScanClass objects in the Project.&lt;br&gt;The Historian objects are organized in a 3-level hierarchy:&lt;br&gt;<strong>Historian</strong>&lt;br&gt;  --&gt; <strong>ScanClass</strong>&lt;br&gt;    ----&gt; <strong>Object</strong>&lt;br&gt;The Historian Scan Class definitions are Sub-objects in the FT_Historian object (instance).</td>
</tr>
<tr>
<td></td>
<td>- <strong>HMI branch</strong>&lt;br&gt;The HMI branch contains all HMI objects in the Project.&lt;br&gt;HMI objects are organized in a 3-level hierarchy:&lt;br&gt;<strong>HMI</strong>&lt;br&gt;  --&gt; <strong>Displays</strong>&lt;br&gt;    ----&gt; <strong>Object</strong>&lt;br&gt;<strong>HMI</strong>&lt;br&gt;  --&gt; <strong>Colors</strong>&lt;br&gt;    ----&gt; <strong>Object</strong>&lt;br&gt;and&lt;br&gt;<strong>HMI</strong>&lt;br&gt;  --&gt; <strong>FTAE</strong>&lt;br&gt;    ----&gt; <strong>Object</strong></td>
</tr>
<tr>
<td></td>
<td>- <strong>Used Libraries branch</strong>&lt;br&gt;The Used Libraries branch displays the libraries that are used in the project.&lt;br&gt;Libraries are organized in a 4-level hierarchy:&lt;br&gt;<strong>Solution</strong>&lt;br&gt;  --&gt; <strong>Library Type</strong>&lt;br&gt;    ----&gt; <strong>Library Category</strong>&lt;br&gt;      ----&gt; <strong>Library Catalog Number (Library Version)</strong></td>
</tr>
<tr>
<td>Controller Preview</td>
<td>A pane that contains a tree control that enables navigation through the different controllers. The Controller Preview pane displays all the project data organized into the following folders:</td>
</tr>
<tr>
<td></td>
<td>- Controllers&lt;br&gt;- Controller-specific data&lt;br&gt;- Logix-specific objects&lt;br&gt;- Task/Programs&lt;br&gt;- Control Modules&lt;br&gt;Controller Preview displays Logix content similar to its final state after it is generated. Use this view to add objects, as well as generate code.</td>
</tr>
<tr>
<td>Class View</td>
<td>A pane that contains a tree control that enables navigation through objects grouped by controller. Objects are organized by the Library object catalog numbers and show the instances below them. Use this view to add, copy, or delete any object instances as well as generate code and reports.</td>
</tr>
<tr>
<td>Object Identifiers</td>
<td>Displays the identification information for the currently selected object. Includes:&lt;br&gt; - <strong>Name</strong>. The name of the object in the database&lt;br&gt; - <strong>Description</strong>. Object category or user specified label for the object.&lt;br&gt; - <strong>Catalog Number</strong>. The specific identifier assigned to the object in the library.&lt;br&gt; - <strong>Solution</strong>. The name of the library object repository for the library object</td>
</tr>
</tbody>
</table>
## Object Parameters

This pane displays the properties tab for the object currently selected in either the System, Controller, or Class view. When applicable, a toolbar is present in the parameter pane that provides the ability to:

- Sort parameters alphabetically
- Sort parameters into groups
- Show only visible parameters
- Show all (visible and hidden) parameters
- Open additional properties

The parameter name shows in the column on the left and the parameter value shows in the column on the right. Change parameter values by typing a new value in the right column. A description of the selected parameter displays at the bottom of the parameters tab.

Some objects have additional tabs for sub-object parameters, (example: analog input of a 1756-IF16 or Attachments for libraries). Sub-object parameters show on an additional tab labeled with the sub-object display name.

Each row in the sub-object parameters tab represents a sub-object. By default, sub-objects sort alphabetically by name. Sort sub-objects in groups by clicking on a column header.

The sub-object name shows in the **Name** column. Additional columns display the sub-object parameters (example: Channel). Change the sub-object name and the sub-object parameter values by typing a new value below the column header.

The sub-object name can be configured as read-only. When the sub-object name is read-only, the value is shown dimmed.

If an object or sub-object parameter value is changed click **Apply changes** to save the changes to the ACM database.

The version number of the Application Code Manager software displays in lower left corner of the object Parameter tab.

## Registered Libraries

A pane that contains a tree view display of all libraries in the connected ACM database. These libraries can be added to an ACM project.

The libraries are organized in a 4-level hierarchy with statistical information in parenthesis:

- **Solution** (Number of objects)
- **--> Library Type** (Number of objects)
- **--> Library Category** (Number of objects)
- **------> Library Catalog Number** (Library Version)

### See also

- [Create a new project](#) on page 32
- [Add a Historian object](#) on page 34
- [Add a new HMI object](#) on page 38
- [Update a project library](#) on page 43
Connect to an ACM database

To use Application Code Manager it must be connected to an ACM database. During the initial configuration of Application Code Manager the default database was configured. Use this procedure to open the Connection properties sheet and update the database connection configuration if the original ACM database server becomes unavailable, if the authentication method needs to be modified, or if a different database is to be used.

To connect to an ACM database

1. Open Application Code Manager. On the main menu, click **File > Connect**.

2. In **Data source** confirm that the database type is **Microsoft SQL Server (SqlClient)**.

3. In **Server Name**, select the computer name and SQL Server instance from the drop down list or type a computer name and SQL Server instance in the following format:

   `<Computer Name> \ <SQL Server Instance>`

   **Tip:** If the database is located on the same computer as Application Code Manager, type **localhost**.

4. In **Log on to the server**, either:

   - Choose **Use Windows Authentication** to allow SQL Server log on using the logged on user account credentials.
   - Choose **Use SQL Server Authentication** to allow SQL Server log on using SQL Server authentication.

   If this method is selected then type the **User name** and **Password** to use to authenticate the connection to the SQL Server. If the SQL database is created by the ACM installer then the user name is "sa" and the default password is "ApplicationAdm1n".
5. In **Connect to a database**, determine whether to connect to a database on the server or to attach a database file.

- Choose **Select or enter a database name** to select a database name from a pull-down list or enter a database name. The database must exist on the database server.
- Choose **Attach a database file** and then type a database file name or click Browse to navigate to the file.

Then, in **Logical name** type the name of the database within the file that Application Code Manager should connect to.

6. (optional) Click **Advanced** to configure settings related to connection protocols, resiliency, content, initialization, pooling, replication, and additional security controls. Applying the recommended settings as defined on the help page for the Advanced Properties will improve performance, especially for network connections.

7. Click **Test Connection** to verify the connection settings specified. If the connection succeeds, click **OK** to close **Connection Properties**.

**See also**
- Connection Properties settings on page 24
- Advanced Properties settings on page 25
- Initial configuration of Application Code Manager on page 16

**Connection Properties settings**

**How do I open the Connection Properties dialog box?**

- On the main menu, click **File > Connect**.

The **Connection Properties** dialog box defines the settings for how Application Code Manager connects with the SQL Server database. Use this dialog box to set the data source, the type of user authentication, to specify a different database file, or to modify advanced security settings.

This table describes the settings in the **Connection Properties** dialog box.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data source:</td>
<td>Database type. Always select Microsoft SQL Server (SqlClient)</td>
</tr>
</tbody>
</table>
## Connect to an ACM database

### Setting | Description
--- | ---
Server name: | Selects a computer name and SQL Server instance from the list or type a computer name and SQL Server instance in the following format:
<Computer Name> \ <SQL Server Instance>

### Log on to the server

| Setting | Description |
--- | ---
Use Windows Authentication | Allows SQL Server log on using Windows authentication. When selected the logged on Windows user account credential will be sent to SQL Server to authenticate the session. |
Use SQL Server Authentication | Allows SQL Server log on using SQL Server authentication. When selected the user name and password must be provided for authentication by the SQL Server.  
- **User name**. The SQL Server user name, "sa" by default.  
- **Password**. The SQL Server password associated with the user name specified, “ApplicationAdm1n” by default.  
- **Save my password**. When selected, saves the SQL Server password specified so that it can be used in the next session. |

### Connect to a database

| Setting | Description |
--- | ---
Select or enter a database name: | Select a database name from the list or enter a database name. |
Attach a database file: | When selected specify the identifiers for the database file. SQL Server database files have two names, the operating system file name used to locate the database in the file system and the logical file name used to identify the database within SQL Server transactions.  
- Type a database file name or use the **Browse** button to use the **Open** dialog to locate the database file by clicking through the file system.  
- In **Logical name**, type the logical name of the database. |
Advanced | Select to configure advanced database properties. |
Test Connection | Tests the connection to the database.  
If a "Test connection succeeded." message is not returned, check that the following settings are correct:  
- Computer name  
- SQL Server authentication  
- Network access (remote SQL Server) |

### See also

- **Initial configuration of Application Code Manager** on page 16
- **Advanced Properties settings** on page 25
- **Connect to an ACM database** on page 23
- **Create an ACM database** on page 79

### Advanced Properties settings

How do I open the Advanced Properties dialog box?

1. On the main menu, click **File >Connect** to open **Connection Properties**.
2. Click **Advanced**.

The **Advanced Properties** dialog box provides a means of changing how the connection between ACM and the SQL Server passes information.
This table describes the settings in the **Advanced Properties** dialog box. The dialog box is divided into functional areas.

**Note:** Applying the recommended settings will improve ACM performance especially for network connections.

<table>
<thead>
<tr>
<th>Area</th>
<th>Setting</th>
<th>Possible Values</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advanced</td>
<td>MultipleActiveResultSets</td>
<td>True, False (default)</td>
<td>When True, multiple result sets can be returned and read from one connection.</td>
</tr>
<tr>
<td>Network Library</td>
<td>blank (required if local)</td>
<td>Named Pipes (DBNMPNTW)</td>
<td>The network library used to establish a connection to an instance of SQL Server. Do not use when the SQL Server is resident on the local host computer, value should be blank.</td>
</tr>
<tr>
<td></td>
<td>TCP/IP (DBMSSOCN)</td>
<td>Shared Memory (DBMSLPCN)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>VIA (DBMSGNET)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Packet Size</td>
<td></td>
<td>8000 (recommended)</td>
<td>Size in bytes of the network packets used to communicate with an instance of SQL Server. PacketSize may be a value in the range of 512 and 32767 bytes.</td>
</tr>
<tr>
<td>Transaction Binding</td>
<td>Implicit Unbind (default)</td>
<td>Explicit Unbind</td>
<td>Indicates the binding behavior of connection to the System.Transactions namespace.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>When set to <strong>Implicit Unbind</strong>, the connection detaches from the transaction when it ends, switching back to autocommit mode.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>When set to <strong>Explicit Unbind</strong> the connection remains attached to the transaction until the transaction is closed. The connection will fail if the associated transaction is not active or does not match the current transaction.</td>
</tr>
<tr>
<td>Type System Version</td>
<td>Latest (default)</td>
<td>SQL Server 2012, SQL Server 2008, SQL Server 2005</td>
<td>Indicates which server type system the provider will expose through the DataReader.</td>
</tr>
<tr>
<td>Connection Resiliency</td>
<td>ConnectRetryCount</td>
<td>2 (recommended)</td>
<td>Number of attempts to restore a connection. The number of reconnections attempted after identifying that there was a connection failure. This must be an integer between 0 and 255. Set to 0 to disable reconnecting on idle connection failures.</td>
</tr>
<tr>
<td></td>
<td>ConnectRetryInterval</td>
<td>5 (recommended)</td>
<td>Delay between attempts to restore connection. The amount of time (in seconds) between each reconnection attempt after identifying that there was a connection failure. This must be an integer between 1 and 60.</td>
</tr>
<tr>
<td>Context</td>
<td>Application Name</td>
<td>.Net SqlClient Data Provider</td>
<td>The name of the application.</td>
</tr>
<tr>
<td></td>
<td>Workstation ID</td>
<td></td>
<td>The name of the workstation connecting to SQL Server.</td>
</tr>
<tr>
<td>Initialization</td>
<td>ApplicationIntent</td>
<td>ReadWrite (default), ReadOnly</td>
<td>Declares the application workload type when connecting to a server.</td>
</tr>
</tbody>
</table>
### Connect to an ACM database

#### Chapter 2

<table>
<thead>
<tr>
<th>Area</th>
<th>Setting</th>
<th>Possible Values</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connect Timeout</td>
<td></td>
<td>30 (recommended)</td>
<td>The length of time in seconds to wait for a connection to the server before terminating the attempt and generating an error. A value of 0 indicates no limit, and should be avoided in a ConnectionString because an attempt to connect waits indefinitely.</td>
</tr>
<tr>
<td>Current Language</td>
<td></td>
<td></td>
<td>The SQL Server Language record name.</td>
</tr>
<tr>
<td>Pooling</td>
<td>Enlist</td>
<td>True (default)</td>
<td>When True sessions in a Component Services environment should automatically be enlisted in a global transaction where required.</td>
</tr>
<tr>
<td></td>
<td>Load Balance Timeout</td>
<td>0 (default)</td>
<td>The minimum amount of time (in seconds) for this connection to live in the pool before being destroyed. When a connection is returned to the pool, its creation time is compared with the current time, and the connection is destroyed if that time span (in seconds) exceeds the value specified by <strong>Load Balance Timeout</strong>. A value of zero (0) causes pooled connections to have the maximum connection timeout.</td>
</tr>
<tr>
<td></td>
<td>Max Pool Size</td>
<td>1000 (recommended)</td>
<td>The maximum number of connections allowed in the pool. Valid values are greater than or equal to 1. Values that are less than <strong>Min Pool Size</strong> generate an error.</td>
</tr>
<tr>
<td></td>
<td>Min Pool Size</td>
<td>0 (default)</td>
<td>The minimum number of connections allowed in the pool. Valid values are greater than or equal to 0. Zero (0) in this field means no minimum connections are initially opened. Values that are greater than <strong>Max Pool Size</strong> generate an error.</td>
</tr>
<tr>
<td></td>
<td>PoolBlockingPeriod</td>
<td>Auto</td>
<td>Defines the blocking period behavior for a connection pool. When connection pooling is enabled and a timeout error or other login error occurs, an exception will be thrown and subsequent connection attempts will fail for the next five seconds, the “blocking period”. If the application attempts to connect within the blocking period, the first exception will be thrown again. Subsequent failures after a blocking period ends will result in a new blocking period that is twice as long as the previous blocking period, up to a maximum of one minute.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AlwaysBlock</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>NeverBlock (recommended)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pooling</td>
<td>True (recommended)</td>
<td>When True, the connection object is drawn from the appropriate pool, or if necessary, is created and added to the appropriate pool. Any newly created connection is added to the pool when closed by the application. In the next attempt to open the same connection, that connection will be drawn from the pool. Connections are considered the same if they have the same connection string. Different connections have different connection strings.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>Replication</td>
<td>Replication</td>
<td>False (default)</td>
<td>Used by SQL Server in replication. Set to True if replication is supported using the connection.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>True</td>
<td></td>
</tr>
<tr>
<td>Security</td>
<td>Authentication</td>
<td>NotSpecified (default)</td>
<td>Specifies the method of authenticating with SQL Server.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SqlPassword</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ActiveDirectoryPassword</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ActiveDirectoryIntegrated</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Column Encryption Setting</td>
<td>Enabled</td>
<td>Default column encryption setting for all the commands on the connection.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Disabled (default)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Encrypt</td>
<td>True</td>
<td>When True, SQL Server uses SSL encryption for all data sent between the client and server if the server has a certificate installed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>False (default)</td>
<td></td>
</tr>
</tbody>
</table>
### Chapter 2  
Connect to an ACM database

<table>
<thead>
<tr>
<th>Area</th>
<th>Setting</th>
<th>Possible Values</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integrated Security</td>
<td>True, False (default)</td>
<td></td>
<td>Whether the connection is to be a secure connection or not. When False, User ID and Password are specified in the connection. When True, the current Windows account credentials are used for authentication.</td>
</tr>
<tr>
<td>Password</td>
<td>****************</td>
<td></td>
<td>Indicates the password to be used when connecting to the data source.</td>
</tr>
<tr>
<td>Persist Security Info</td>
<td>True, False (default)</td>
<td></td>
<td>When False, security-sensitive information, such as the password, is not returned as part of the connection if the connection is open or has ever been in an open state.</td>
</tr>
<tr>
<td>TrustServerCertificate</td>
<td>True (recommended), False</td>
<td></td>
<td>When True (and Encrypt is set to True), SQL Server uses SSL encryption for all data sent between the client and server without validating the server certificate. If TrustServerCertificate is set to True and Encrypt is set to False, the channel is not encrypted.</td>
</tr>
<tr>
<td>User ID</td>
<td>sa</td>
<td></td>
<td>Indicates the user ID to be used when connecting to the data source.</td>
</tr>
</tbody>
</table>

### Source

<table>
<thead>
<tr>
<th>Area</th>
<th>Setting</th>
<th>Possible Values</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AttachDbFilename</td>
<td></td>
<td></td>
<td>The name of the primary file, including the full path name, of an attachable database.</td>
</tr>
<tr>
<td>Context Connection</td>
<td>True, False (default)</td>
<td></td>
<td>When True, indicates the connection should be from the SQL Server context. Available only when running in the SQL Server process.</td>
</tr>
<tr>
<td>Data Source</td>
<td>localhost\SQLACM (default)</td>
<td></td>
<td>Indicates the name of the data source to connect to.</td>
</tr>
<tr>
<td>Failover Partner</td>
<td></td>
<td></td>
<td>The name or network address of the instance of SQL Server that acts as a failover partner.</td>
</tr>
<tr>
<td>Initial Catalog</td>
<td>Initial Database Name</td>
<td></td>
<td>The name of the initial catalog or database in the data source.</td>
</tr>
<tr>
<td>MultiSubnetFailover</td>
<td>True, False (default)</td>
<td></td>
<td>If your application is connecting to a high-availability, disaster recovery (AlwaysOn) availability group (AG) on different subnets, setting this value to True configures SqlConnection to provide faster detection of and connection to the (currently) active server. When set to True, the application is required to retrieve all IP addresses for a particular DNS entry and attempt to connect with the first one in the list. If the connection is not established within 0.5 seconds, the application will try to connect to all others IP addresses in parallel. When the first IP address answers, the application will establish the connection with the respondent IP address. If MultiSubnetFailover is set to True, this setting is ignored. If Failover Partner is specified, this setting is ignored. The default setting is False if Authentication is set to either Active Directory Password or Active Directory Integrated, otherwise the default setting is True.</td>
</tr>
<tr>
<td>TransparentNetworkIPResolution</td>
<td>True (default), False</td>
<td></td>
<td>If your application connects to different networks, setting this value to True configures SqlConnection to provide transparent connection resolution to the currently active server, independently of the network IP topology. When set to True, the application is required to retrieve all IP addresses for a particular DNS entry and attempt to connect with the first one in the list. If the connection is not established within 0.5 seconds, the application will try to establish connection with all others IP addresses in parallel. When the first IP address answers, the application will establish connection with the respondent IP address.</td>
</tr>
<tr>
<td>User Instance</td>
<td>True, False (default)</td>
<td></td>
<td>Indicates whether the connection will be re-directed to connect to an instance of SQL Server running under the user's account.</td>
</tr>
</tbody>
</table>

**See also**

[Connection Properties settings on page 24](#)
Project Development

Application Code Manager organizes components into projects. There are two methods of creating projects:

- Use the Object Configuration Wizard
- Use an existing project

Once a project is created, add objects for your application, such as:

- Content from the ACM library
- Historian objects
- HMI objects

See also

Create a new project on page 32
Create a new project from an existing project on page 33

Project commands

In the System View pane, right-click the Project branch to view the project commands. Different branches have different commands available.

This table describes each project command.

<table>
<thead>
<tr>
<th>Command</th>
<th>Branch</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>View &gt; Project History</td>
<td>Project (top)</td>
<td>Displays a report showing the Project History.</td>
</tr>
<tr>
<td>Export</td>
<td>Project (top), Used Libraries, Library Module</td>
<td>Starts the Import Export Manager.</td>
</tr>
<tr>
<td>Import</td>
<td>Project (top)</td>
<td>Starts the Import Export Manager.</td>
</tr>
<tr>
<td>Delete</td>
<td>Project (top), Historian object, Displays object, FTAE object</td>
<td>Deletes the selected object.</td>
</tr>
<tr>
<td>Rename</td>
<td>Project (top), Historian object, Displays object, FTAE object</td>
<td>Renames the selected object.</td>
</tr>
<tr>
<td>Refresh</td>
<td>Project (top), Historian, ScanClass, Historian object, HMI, Displays, Displays object, FTAE, FTAE object</td>
<td>Refreshes the tree view.</td>
</tr>
<tr>
<td>Update All Child Objects</td>
<td>Historian, HMI</td>
<td>Opens the Update Used Libraries window.</td>
</tr>
<tr>
<td>Add</td>
<td>ScanClass, Colors, Displays, FTAE</td>
<td>Adds a new object (instance) to the selected Category.</td>
</tr>
</tbody>
</table>
Create a new project

Create a new project when the new project is not similar to existing projects.

To create a new project

1. Click File point to New and then click Project. The Object Configuration Wizard is displayed.

2. On the Select a library page click the + symbol to expand the library category and display the libraries registered in the connected ACM database.

3. Click a library to select it and then click Next.

4. If the library contains a library link, click the Linked Libraries tab.

5. Click the link and choose how to instantiate the linked library, select either:

<table>
<thead>
<tr>
<th>Command</th>
<th>Branch</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Paste Special</td>
<td>ScanClass, Colors, Displays, FTAE</td>
<td>Displays PasteSpecial window where child objects and reference values can be included or excluded.</td>
</tr>
<tr>
<td>Paste</td>
<td>ScanClass, Colors, Displays, FTAE</td>
<td>Pastes a copied object in the selected location.</td>
</tr>
<tr>
<td>Copy</td>
<td>Historian object, Displays object</td>
<td>Copies the selected object.</td>
</tr>
<tr>
<td>Update</td>
<td>Used Libraries, Library Module</td>
<td>Displays the Update Used Library dialog.</td>
</tr>
<tr>
<td>Generate Historian</td>
<td>Historian object</td>
<td>Generates a copy of the selected Historian object from ACM. The Historian object is saved to an external folder.</td>
</tr>
<tr>
<td>Generate Displays</td>
<td>Displays Object</td>
<td>Generates a copy of the selected Displays object from ACM. The Display object is saved to an external folder.</td>
</tr>
<tr>
<td>Generate FTAE</td>
<td>FTAE Object</td>
<td>Generates a copy of the selected FTAE object from ACM. The FTAE object is saved to an external folder.</td>
</tr>
</tbody>
</table>

See also

Create a new project on page 32

Add ACM library content to an ACD project on page 87

Add a Historian object on page 34

Add an HMI object on page 38
a. **Create New Instance.** To create a new instance of the library the link is attached to.

b. **Link to Existing Instance.** To link to a library instance of the target library that has already been created.

6. Enter a unique name in the Name field. Enter a description in the Description field and click Finish.

The new project appears highlighted in the **System View** pane.

**See also**

- Initial configuration of Application Code Manager on page 16
- Create a new project from an existing project on page 33
- Add a Historian object on page 34
- Add an HMI object on page 38

---

**Create a new project from an existing project**

Use an existing project in the ACM database as the basis for a new project if the projects use the same components and libraries.

If the project is in a different ACM database, such as one from another installation, you must connect to that database, export the project, and then import it to the current database.

**To create a new project from an existing project**

1. Click **File** point to **New** and then click **Project from Existing Project**. The **New Project Wizard** is displayed.

2. On the **Create a new Project from an existing Project** page in From Project type a Project name or select the project name from the list.

3. In **New Name**, type a name for the new project.

4. (optional) In **New Description**, type a description of the new project.

5. Click **Finish**.

The new project appears highlighted in the **System View** pane.

**See also**

- Connect to an ACM database on page 23
- Add a Historian object on page 34
Add a Historian object

Add a Historian object to the project to support integration of Historian data.

To add a Historian object

1. In the System View pane, expand Historian, right click ScanClass and then click Add.

   The Object Configuration Wizard appears.

2. Under Select a library in the Solution column, click the + symbol to expand a library category and display the Historian libraries registered in the connected ACM database.

3. Click a Historian library, the row highlights to indicate that it is selected. Click Next.

4. In Name, type a unique name for the Historian object and then click Finish.

   The Historian object is added to the System View pane as a child object of ScanClass.

See also

Add a Historian sub-object on page 34

Historian Point Type parameters on page 35

Generate a Historian object on page 37

Update a project library on page 43

Add a Historian sub-object

Historian objects have sub-objects that represent the point types configured. A FactoryTalk Historian point is the basic building block for controlling data flow to and from the FactoryTalk Historian SE server. A single point is configured for each measurement value that needs to be archived.

To add a Historian sub-object

1. In the System View pane, expand Historian, expand ScanClass and then click the Historian object to which the sub-object will be added.

2. In the object parameters pane, select the Point Type tab.

3. Right-click in the white space below the objects shown in the tab. Click Add New.
Repeat this step to add all of the points that require configuration.

4. Click the appropriate column and modify the parameters as needed.

5. Click **Apply Changes** to save the parameter configuration.

**Group sub-objects**

Group sub-objects into categories based on the data columns.

- Double-clicking on a column heading updates the display in the parameters tab to group items with the same value together.
- Right-click anywhere in the parameters tab and select **Reset Grouping** to return to the default display.

**See also**

- [Add a Historian object](#) on page 34
- [Historian Point Type parameters](#) on page 35
- [Generate a Historian object](#) on page 37

### Historian Point Type parameters

The **Point Type** tab displays a columnar grid of the parameters defined for the points of the Historian object.

This table describes the parameters in the **Point Type** tab of the Historian object.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the point. Double-click to configure a unique name for the point.</td>
</tr>
<tr>
<td>ScanClassNo</td>
<td>The scan class determines the frequency at which input points are scanned for new values.</td>
</tr>
<tr>
<td>ExcDev</td>
<td>Exception Deviation. Specifies in engineering units how much a point's value must change before the interface considers it a significant value, and sends it to the server. As a general rule, you should set the exception slightly smaller than the precision of the instrument system.</td>
</tr>
<tr>
<td>ExcDevPercent</td>
<td>Specifies the exception deviation as a percentage of span instead of in engineering units. For digital, string and Blob tags, ExcDev and ExcDevPercent are ignored and display by applications as zero.</td>
</tr>
<tr>
<td>ExcMax</td>
<td>Exception Maximum. Specifies a limit on how long the interface can go without reporting a value to the Historian server. After the ExcMax time period elapses, the interface sends the next new value to the server, regardless of whether the new value is different from the last reported value.</td>
</tr>
<tr>
<td>ExcMin</td>
<td>Exception Minimum. Specifies a limit on how frequently the interface can report values to the server. Example: For the interface to wait a full ten minutes before reporting a new value to the server, set the ExcMin attribute to ten minutes. ExcMin is typically set to zero.</td>
</tr>
<tr>
<td>CompDev</td>
<td>Compression Deviation. Specifies in engineering units how much a value may differ from the previous value before it is considered to be a significant value. In most environments, set CompDev to the precision of the data source or hardware (instrument). Initially, set it to a lower value so that important data is not lost. After collecting data for a while, go back and check the data for your most important tags, and then adjust CompDev to a higher value, if necessary. Setting the CompDev attribute value too low causes too little data compression, and wastes space in the archive. Setting the value too high causes loss of useful data. For most flows, pressures, and levels, use a deviation specification of 1% or 2% of span. For temperatures, the deviation should usually be 1 or 2 degrees.</td>
</tr>
</tbody>
</table>
### Parameter Description

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CompDevPercent</td>
<td>Specifies the compression deviation as a percentage of span instead of in engineering units.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> For non-numeric tags, CompDev and CompDevPercent are ignored. They will be displayed by applications as zero.</td>
</tr>
<tr>
<td>CompMax</td>
<td>Compression Maximum. A point is archived if the elapsed time since the previous time the point was saved is greater than the maximum time. The recommended maximum time specification is one work shift (example: 8 hours). Duplicate values will be archived if the elapsed time exceeds CompMax. In most environments, set CompMax to the same value for all points in the system.</td>
</tr>
<tr>
<td>CompMin</td>
<td>Compression Minimum. A point is archived if the elapsed time since the previous time the point was saved is greater than or equal to the minimum time, and the value has changed by more than the deviation. For data points associated with interfaces that send exception reports, set CompMin to 0.</td>
</tr>
<tr>
<td>Compressing</td>
<td>Determines whether compression is turned on or off for the point. Compression should be turned on for all real-time points in the system. Set compression OFF for laboratory and manually entered tags so every value is recorded in the archive. To turn compression on, select the Compressing parameter for most points. With compression off, every value sent is saved in the archive. Compression affects digital points, since a new value is recorded only when the current value changes. Points of types Blob and String have a similar behavior; new events pass compression only when the value changes. String values are compared ignoring case. Example: “ValUE” and “valUE” are evaluated as equal. For Blob events, any change is significant.</td>
</tr>
<tr>
<td>Step</td>
<td>The Step parameter affects only numeric points. It defines how numeric archived values are interpolated.</td>
</tr>
<tr>
<td></td>
<td>• When the checkbox is clear (OFF) archived values for the point are interpreted as a continuous signal and adjacent archived values are linearly interpolated. Example: at 12:00:00, the value 101.0 is archived and at 12:01:00, the value 102.0 is archived. A request for the archive value at 12:00:30 would return 101.5.</td>
</tr>
<tr>
<td></td>
<td>• When the checkbox is selected (ON) archived values for the point are interpreted discretely and adjacent archived values are not interpolated. An archived value is assumed constant until the next archived value is recorded. Example: at 12:00:00, the value 101.0 is archived, at 12:01:00, the value 102.0 is archived. A request for the value at 12:00:30 would return 101.0.</td>
</tr>
<tr>
<td></td>
<td>Data coming from continuous signals (such as signals from thermocouples, flow meters, and other devices that provide continuous measurement) should be archived in points with the step flag OFF.</td>
</tr>
<tr>
<td></td>
<td>Data coming from discrete measurements (such as sampled lab data, batch charge weight, and other user-defined inputs) should be archived in points with the step flag ON.</td>
</tr>
<tr>
<td></td>
<td>In addition, the step flag affects the compression calculation. When it is ON a linear change of value greater than or equal to CompDev passes compression. When the step flag is OFF the complete compression algorithm is applied.</td>
</tr>
<tr>
<td>DigitalSet</td>
<td>For digital points, the DigitalSet attribute specifies the name of the digital state set associated with the tag. The DigitalSet attribute has no meaning for non-digital tags.</td>
</tr>
<tr>
<td>Scan</td>
<td>Toggling this checkbox, turns scanning ON or OFF for a point. By default scanning is turned ON (selected), which indicates that the program should be able to collect data for the point. Setting the Scan attribute to OFF (cleared) turns off data collection for that point.</td>
</tr>
<tr>
<td>AdviseMode</td>
<td>Specifies that the point should run in the Advised data collection mode. In this mode, data is collected only when a value changes in the controller. It is not based on the scan rate. AdviseMode is selected by default as this mode is the most efficient because data is sent to the Historian server only when the value changes.</td>
</tr>
<tr>
<td>SubObject Description</td>
<td>Descriptive text that distinguishes this point from other points.</td>
</tr>
</tbody>
</table>

### See also

- [Add a Historian object](#) on page 34
- [Add a Historian sub-object](#) on page 34
- [Generate a Historian object](#) on page 37
Generate a Historian object

Once a Historian Object has been configured it can be used to generate a FactoryTalk Historian SE import file, this file is a comma-separated value (.csv) file.

To generate a Historian object file

1. In the System View pane, expand the Historian folder and the ScanClass folder so that the Historian objects are visible.

2. Right-click the Historian object to export and then choose the controller for which support is required.
   - Click All Controllers to create an export file that supports all controllers in the project.
   - Click a specific controller from the list if this object will only be used with that type of controller.

   The Save As dialog box opens.

3. Use the tree control to browse to the location in which to save the Historian object file or use the default location:

   C:\Users\<user name>\Documents\Studio 5000\Projects

   The variable <user name> is replaced by the logged in user account name.

4. In File name type a name for the file or use the default file name which is created according to the following pattern:

   <Controller Name>_Historian object Name>.csv

   The variable <Controller Name> is replaced by the name of the controller selected if a specific controller support was selected. If All Controllers are supported the controller portion of the default file name and the underscore are omitted.

5. Click Save. The Historian object generation is started.

6. Once the process is completed, the Generation Complete status dialog box appears, click OK to close the dialog box.

See also

   Add a Historian object on page 34

   Generate an HMI object on page 40
Add an HMI object

Add an HMI object to support FactoryTalk View SE, FactoryTalk View Machine Edition, or FactoryTalk Alarms and Events displays in the project.

To add an HMI object

1. In the System View pane, expand HMI to see the HMI categories.
   - To add a FactoryTalk View SE or FactoryTalk View Machine Edition HMI object, right-click Displays and then click Add.
   - To add a FactoryTalk Alarms and Events object, right-click FTAE and then click Add.

   The Object Configuration Wizard appears.

2. Under Select a library in the Solution column, click the + symbol to expand a library category and display the libraries registered in the connected ACM database.

3. Click a library, the row highlights to indicate that it is selected. Click Next.

4. In Name, type a unique name for the object and then click Finish.

   The object is added to the System View pane as a child object of the HMI category.

See also

Add a Display sub-object on page 38

HMI Display parameters on page 39

Add an Alarm Group sub-object on page 41

FTAE Alarm Group parameters on page 42

Generate an HMI object on page 40

Update a project library on page 43

Add a Display sub-object

FactoryTalk View objects have sub-objects that represent the display types configured. A FactoryTalk View graphic display represents a run-time operator’s view of plant activity. A graphic display can show system or process data, and provide an operator with ways to write values to external devices such as programmable controllers.
To add a Display sub-object

1. In the System View pane, expand HMI, expand Displays and then click the FactoryTalk View object to which the sub-object will be added.

2. In the object parameters pane, select the Displays tab.

3. Right-click in the white space below the objects shown in the tab. Click Add New.

   Repeat this step to add all of the Displays that require configuration.

4. Click the appropriate column and modify the parameters as needed.

5. Click Apply Changes to save the parameter configuration.

Group sub-objects

Group sub-objects into categories based on the data columns.

- Double-clicking on a column heading updates the display in the parameters tab to group items with the same value together.
- Right-click anywhere in the parameters tab and select Reset Grouping to return to the default display.

See also

Add an HMI object on page 38

HMI Display parameters on page 39

Generate an HMI object on page 40

HMI Display parameters

The Display tab displays a columnar grid of the parameters defined for the displays of the FactoryTalk View object.

This table describes the parameters in the Displays tab.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the display. Double-click to configure a unique name for the display.</td>
</tr>
<tr>
<td>DisplayTitle</td>
<td>Specifies the content of the display title bar.</td>
</tr>
<tr>
<td>DisplayLeft</td>
<td>Sets the left edge of a display to the number of units specified from the left edge of the screen.</td>
</tr>
<tr>
<td>DisplayTop</td>
<td>Sets the top edge of a display to the number of units specified from the top edge of the screen.</td>
</tr>
<tr>
<td>DisplayWidth</td>
<td>Sets the width of a display.</td>
</tr>
<tr>
<td>DisplayHeight</td>
<td>Sets the height of a display.</td>
</tr>
<tr>
<td>DisplayBackColor</td>
<td>Sets the background color of the display.</td>
</tr>
</tbody>
</table>
Parameter | Description
---|---
DisplaySecurity | Identifies the security class of the screen.
UpdateRate | Defines the amount of time to update a display in response to an event. Default setting is 1.0 seconds. For best performance set to .1 seconds.
LeftIndexMax | Identifies the maximum number of entries in the left index.
SubObject Description | Provides a unique description of the display sub-object.

See also

Add a Display sub-object on page 38

Generate an HMI object

Once a HMI display object has been configured it can then be used to generate an import file for use with FactoryTalk View SE or FactoryTalk View ME. This file is an Extensible Markup Language (.xml) file.

To generate an HMI object file

1. In the System View pane, expand the HMI folder and the display category folder so that the objects are visible.

2. Right-click the object to export.
   - For FactoryTalk View objects, select Generate Displays and then click All Displays to create an export file that supports all display types.
   - For FTAE objects, select Generate FTAE and then click a specific controller from the list if this object will only be used with that type of controller or click All Controllers if this object will be used with all project controllers.

   The Save As dialog box opens.

3. Use the tree control to browse to the location in which to save the HMI object file or use the default location:

   C:\Users\<user name>\Documents\Studio 5000\Projects

   The variable <user name> is replaced by the logged in user account name.

4. In File name type a name for the file or use the default file name which is created according to the following pattern:

   <Controller Name>_Object Name.csv

   The controller portion of the default file name and the underscore are omitted when the All Controllers option is selected. For FTAE objects the variable <Controller Name> is replaced by the name of the
controller selected if a specific controller was selected. FactoryTalk View objects are not controller specific.

5. Click **Save**. The HMI object generation is started.

6. Once the process is completed, the **Generation Complete** status dialog box appears, click **OK** to close the dialog box.

**See also**

- [Add a new HMI object](#) on page 38

### Add an Alarm Group sub-object

FactoryTalk Alarm and Event objects have sub-objects that represent the Alarm Groups configured in the FTAE system. In FactoryTalk Alarm and Events, alarms can be assigned to groups. A group represents a set of alarms with a common association. An alarm group can contain other groups organised in a hierarchy (FTAE only supports to 5 levels deep).

**To add an Alarm Group sub-object**

1. In the **System View** pane, expand **HMI**, expand **FTAE** and then click the FactoryTalk FTAE object to which the sub-object will be added.

2. In the object parameters pane, select the **Alarm Group** tab.

3. Right-click in the white space below the objects shown in the tab. Click **Add New**.

   Repeat this step to add all of the Alarm Group that require configuration.

4. An alarm group hierarchy is achieved by associating one alarm group to another using the ParentAlarmGroupId parameter.

5. Click the appropriate column and modify the parameters as needed.

6. Click **Apply Changes** to save the parameter configuration.

**Group sub-objects**

Group sub-objects into categories based on the data columns.

- Double-clicking on a column heading updates the display in the parameters tab to group items with the same value together.

- Right-click anywhere in the parameters tab and select **Reset Grouping** to return to the default display.
See also

Add an HMI object on page 38

FTAE Alarm Group parameters on page 42

Generate an HMI object on page 40

**FTAE Alarm Group parameters**

The Alarm Group tab displays a columnar grid of the parameters defined for the alarm groups of the FactoryTalk Alarms and Events object.

This table describes the parameters in the **Alarm Group** tab.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the sub-object representing the Alarm Group (this is not the name that will be used in the FTAE system). Through name formatting a unique sub-object name is built up from the AlarmGroup and ParentAlarmGroupID parameters.</td>
</tr>
<tr>
<td>AlarmGroupID</td>
<td>Each AlarmGroupID must be unique (0 is reserved for the FTAETagServer ID).</td>
</tr>
<tr>
<td>ParentAlarmGroupID</td>
<td>The ParentAlarmGroupID determines the alarm group hierarchy (Default: 0 = FTAETagServer ID).</td>
</tr>
<tr>
<td>AlarmGroup</td>
<td>The alarm group name that will created in the FTAE system and assigned to FTAE alarm tag objects that reference it.</td>
</tr>
<tr>
<td>SubObject Description</td>
<td>Provides a unique description of the alarm group sub-object.</td>
</tr>
</tbody>
</table>

Find an object

Use the **Find** dialog box to search for an object in the Class View pane, or contents in the Controller Preview tree.

**Note:** This feature is only available if the product has a Standard activation license and is not available in Lite mode.

To find an object

1. On the toolbar, select **EDIT > Find**.
2. In **Find what**, enter the name of an object.
3. In **Look in**, select Class View or Controller Preview.
4. (optional) In **Find Options**, select the check boxes of Match case and Match whole word.
5. Select **Find Next** or **Find Previous**.

See also

Project Development on page 31
Update a project library

Any of the libraries used in a project can be updated to incorporate additions or modifications to library components.

If Newer Library Version Available is enabled in Application Code Manager settings, indications appear next to objects that can be updated.

To update a project library

1. In the System View pane, expand Used Libraries to see the library branches. Library versions are shown in parenthesis.

2. Right-click the library branch to be updated and then click Update.

The Update Used Libraries dialog box is displayed.

3. A tab is shown for each library in the selected library branch. Libraries with available updates will have different revisions.

   If multiple libraries are included in the library branch, click the tab for the library to update. Each library tab has three sub-tabs.

4. In Update Library Revision, click the drop-down arrow to select the new revision number for the library.

5. Click the Library Changes sub-tab to review the parameter changes for the new version.

6. Click the Objects to Update sub-tab to define which objects in the library are to be updated.

   • Under Update, select the checkbox to update the project objects that reference this project library.

   • Clear the checkbox for each object that should not be updated.

7. Click the Relink References to Removed Items sub-tab and select the parameters and sub-parameters to relink to the objects that are affected by this update.

8. Click Finish.

See also

Upgrade the Application Code Manager application on page 17

Library objects on page 12
**Extract attached files**

Files attached to a project, Historian object, HMI display, FactoryTalk Alarms and Events server, controller, library, or library object can be extracted to a folder of your choice. When extracting the attachments from the various scopes in a project, all the objects within the scope have their attachment include conditions evaluated and only those attachments that evaluate to true will be extracted to their relative path. If the relative path is parameterized, the path will be resolved. If the resolved path is invalid, the attachment will be extracted to the root extraction folder. Attachments that are the same, but have a different relative path will be extracted into each of the relative paths. Attachments that already exist at the relative path will be overwritten.

**To extract attached files in System View**

1. In **System View**, right-click on one of the following, then select **Extract Attached Files**:
   - The ACM project
   - A Historian object
   - An HMI display
   - A FactoryTalk Alarms and Events server
   - A controller or controllers
   - A library
   - A library object

2. In **Browse for Folder**, select a folder where the files should be extracted, then select **OK**.

**To extract attached files in Registered Libraries**

1. In **Register Libraries**, right-click an object or its descendant objects with an attachment, and then select **Extract Attached Files**.

2. In **Browse for Folder**, select a folder where the file should be extracted or create a new folder, and then click **OK**.

**To extract attached files in Class View**

1. In **Class View**, right-click an object or its descendant objects with an attachment, and then select **Extract Attached Files**.

2. In **Browse for Folder**, select a folder where the file should be extracted or create a new folder, and then click **OK**.
When you extract an attachment in Class View, if the extraction path of an attachment in the library is configured as 
(ObjectName)\Folder1\Folder2 by an expression in Library Object Manager and the Object instance is "XV100", then upon extraction of the attachment the extracted relative path will be XV100\Folder1\Folder2.

See also

Project Development on page 31

Indications

If Modified Objects or Newer Library Version Available is enabled in Application Code Manager settings, indications appear next to objects that have been modified or can be updated.

<table>
<thead>
<tr>
<th>Indication</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Modification" /></td>
<td>This object has been modified.</td>
</tr>
<tr>
<td><img src="image" alt="Library Update" /></td>
<td>This library has a newer version available in Registered Libraries.</td>
</tr>
<tr>
<td><img src="image" alt="Node Update" /></td>
<td>This node or folder contains one or more libraries that has a newer version available in Registered Libraries.</td>
</tr>
</tbody>
</table>

See also

Update a project library on page 43

ACM Default Settings on page 86
Chapter 4

Registered Libraries

Registered Libraries displays all libraries (classes) in the connected ACM database in a tree view. Add a library to an ACM project to use the objects in it with your project.

A library object defines parameters, subclasses, user interface contents, and portions of controller code (example: Logix) and HMI code (example: FactoryTalk View SE/ME). It can also include custom properties that were applied using the Library Designer.

Right-click any branch in the Registered Libraries tree view to view the commands available.

This table describes each command. Commands appear at the applicable level of the tree.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registered Library</td>
<td>Used when a library object is provided as a HSL4 file and needs to be incorporated into ACM. Registers one or more library files (HSL4).</td>
</tr>
<tr>
<td>View &gt; Library Usage</td>
<td>Displays a report showing the Registered Library usage for the libraries in the connected ACM database.</td>
</tr>
<tr>
<td>Export</td>
<td>Exports the selected library to an HSL4 (xml) formatted file.</td>
</tr>
<tr>
<td>Reconstitute ACD</td>
<td>Opens the ACD Re-constitution Wizard.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes the selected library in the connected ACM database. Libraries that are used in projects cannot be deleted. If the library selected for deletion is in use, a Global Library usage report is displayed instead.</td>
</tr>
<tr>
<td>Delete all unused</td>
<td>Deletes all unused libraries for the selected level in the connected ACM database.</td>
</tr>
<tr>
<td>Refresh</td>
<td>Refreshes the tree view.</td>
</tr>
<tr>
<td>ACM Library</td>
<td></td>
</tr>
<tr>
<td>View &gt; Pending Libraries per Solution</td>
<td>Displays a report showing the registered library usage for the pending libraries in the selected solution. Available only at a solution branch.</td>
</tr>
<tr>
<td>View &gt; Library Usage per Solution</td>
<td>Displays a report showing the registered library usage for the libraries in the selected solution. Available only at a solution branch node.</td>
</tr>
<tr>
<td>View &gt; Pending Libraries per Library Type</td>
<td>Displays a report showing the registered library usage for the selected pending library type. Available only at a library type branch node.</td>
</tr>
<tr>
<td>View &gt; Project Library Usage per Library Type</td>
<td>Displays a report showing the registered library usage for the selected library type. Available only at a library type branch node.</td>
</tr>
<tr>
<td>View &gt; Pending Libraries per Category</td>
<td>Displays a report showing the registered library usage for the selected pending library category. Available only at a library category branch node.</td>
</tr>
</tbody>
</table>
Register an ACM library object

Register an ACM library object to use it in a project.

Prerequisites

- Verify that the ACM application is connected to the correct ACM database.

  **Tip:** The connected database is shown in parenthesis in the Application Code Manager title bar.

To register an ACM library object

1. In the Register Libraries tree, right-click the library object to register.
2. Click Register.
3. The Libraries Registration window opens.
4. Review the information provided in the Library Registration window. Perform any indicated actions.
5. Click **Finish** to complete the registration process.

See also

- View registered library usage on page 50
- Share libraries, templates, and schedules on page 51
- Connect to an ACM database on page 23
- Reconstitute the ACD on page 49
Reconstitute an ACD file

Reconstitute an ACD file when the original Logix Designer project used to create the object is not available for editing. After reconstituting the ACD file, it can be edited using Logix Designer.

Hardware module libraries and ACM v1.x libraries do not support this feature.

To reconstitute an ACD file

1. In the Registered Libraries tree view, right-click the library file(s).
   To select multiple files hold down the <Ctrl> key and select the library files.

2. Click Reconstitute ACD.

   The ACD Reconstitution Wizard opens.

3. The Object Configuration page displays the details of the objects to be included in the ACD file.

   Click Next.

4. The Resolve Dependencies page displays any dependencies between the library objects that must be resolved.

   • Click Add Dependency Library and add the library that contains required add-on instructions (AOIs) or user-defined data types (UDTs).

   • Click Next.

5. The Unresolved Names page displays any objects that have unresolved name value. Click in the Value column and type valid values for any names highlighted on the page then click Next.

6. The L5X Generation Successful page displays a preview of the Logix content contained in the ACD file displays.

7. Click Next to display the Save As screen. In File name type a file name for the ACD file.

8. Click Save to save the file. The ACD successfully saved message displays with a link to open the folder where the file was saved.

9. Click Finish to close the wizard.

See also

Add a controller from an ACD or L5X file on page 57
There are several different reports that provide information about registered library usage. Determine the scope of the report by navigating to the level of interest and then generate the report.

To view registered library usage

1. Select the scope of the report.

   - To generate a report that includes all of the registered libraries in the connected ACM database:
     Right-click **Registered Libraries**, then point to **View**, then click **Library Usage**.

   - To generate a report that includes all of the registered libraries used in a solution:
     In the **Registered Libraries** tree, right-click the solution folder, then point to **View**, then click **Library Usage per Solution**.

   - To generate a report that includes all of the registered libraries used in a library type:
     In the **Registered Libraries** tree, expand the solution folder, right-click the library type folder, then point to **View**, then click **Library Usage per Library Type**.

   - To generate a report that includes all of the registered libraries used in a library category:
     In the **Registered Libraries** tree, expand the solution folder, expand the library type folder, right-click the library category folder, then point to **View**, then click **Library Usage per Library Category**.

   - To generate a report that includes all of the registered libraries used for a library object:
     In the **Registered Libraries** tree, expand the solution folder, expand the library type folder, expand the library category folder, right-click the library object then point to **View**, then click **Library Usage per Library**.

2. The selected report appears in a new window.
Share libraries, templates, and schedules

All libraries in the Registered Libraries tree are available to all ACM users that are connected to the ACM database.

Schedule templates can be created that are limited to just a specific user account. If a template that is limited to a user account needs to be shared with other users, use Import/Export Manager to either copy or move the template from the users folder to the central ACM database.

ACM Program folder templates can be shared by placing the template files in a shared network folder.

Project data that can be reused in multiple ACM projects can be exported to a schedule and shared by placing the Schedule file in a shared network folder.

To avoid the possibility of ACM users overwriting each other’s work in a central ACM database, ACM users should work in different projects or branches of the same project tree view. Project work can be divided by function (example: Controller Hardware, Controller Software, HMI, Historian) or by area (example: Receiving, Mixer, Shipyard).

Use the partial export option to avoid exporting the same data to more than one schedule. If the same data is imported from more than one schedule, the last schedule imported will determine the data.

To help ensure that work is not lost, export a project schedule periodically for backup.

See also

Reports on page 89

Generate a report on page 91

Local library and template file location

By default library object files and template files are stored in this location:

C:\Users\Public\Public Documents\Rockwell Automation\Studio 5000

There are separate Libraries and Templates folders with an Application Code Manager subfolder in each. The library object repositories, (RA-LIB) ACM and (RA-LIB) Process, and the template repository, (RA-TPL) ACM, are located in their respective subfolders.
The (RA-TPL) ACM folder is set as the default documentation path for the Application Code Manager application. To change the documentation path open ACM Default Settings and specify the path to use.

**Tip:** If working in a collaborative design environment with a shared ACM database, template files can be copied to a shared folder so that all users can access them.

**See also**

- Initial configuration of Application Code Manager on page 16
- Registered Libraries on page 47
- ACM Default Settings on page 86
Configure controllers

Right-click any controller branch in the Controller Preview or Class View pane to view the controller configuration commands.

This table describes each Controller command. Commands appear at the applicable level of the tree.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generate Controller</td>
<td>Generates a copy of the selected display object from ACM. Displays the Logix Code Generation dialog box. Generates code (example: LSX) for the selected controller.</td>
</tr>
<tr>
<td>Add New</td>
<td>Adds a new controller object to the project using the Object Configuration Wizard.</td>
</tr>
<tr>
<td>Add New from ACD/LSX</td>
<td>Adds a new controller object to the project from an ACD or LSX file. Additional objects may also be added depending on what is configured and available in the ACD file.</td>
</tr>
<tr>
<td>Update from ACD/LSX</td>
<td>Updates the ACM project based on an existing ACD file.</td>
</tr>
<tr>
<td>Detach from ACD/LSX</td>
<td>Removes the link between the ACM project and the attached ACD or LSX file.</td>
</tr>
<tr>
<td>Merge Controller</td>
<td>Merges content between the original ACM project, updated ACM project, and existing ACD or LSX file. Utilizes the Compare Tool.</td>
</tr>
<tr>
<td>Add New Program</td>
<td>Adds a new program to the current controller object using the Object Configuration Wizard.</td>
</tr>
<tr>
<td>Add New Task</td>
<td>Adds a new task to the current controller object using the Object Configuration Wizard.</td>
</tr>
<tr>
<td>Paste</td>
<td>Pastes the contents of the clipboard to the selected controller.</td>
</tr>
<tr>
<td>Paste Special</td>
<td>Pastes the contents of the clipboard to the selected controller with options.</td>
</tr>
<tr>
<td>Generate Partial Program</td>
<td>Displays the Logix Code Generation dialog box. Generates a partial program for the selected task.</td>
</tr>
<tr>
<td>Generate Partial Routine</td>
<td>Displays the Logix Code Generation dialog box. Generates a partial routine for the selected task.</td>
</tr>
<tr>
<td>Move Up</td>
<td>Moves the highlighted object up the tree.</td>
</tr>
<tr>
<td>Move Down</td>
<td>Moves the highlighted object down the tree.</td>
</tr>
<tr>
<td>Set Main Routine</td>
<td>This command is available when multiple main routines have been added from different library objects. There can only be one main routine. If a main routine is already selected this command is dimmed.</td>
</tr>
<tr>
<td>Set Routine Instance Order</td>
<td>Displays the Instance Execution Order dialog box which specifies the order in which any contributing objects insert their code into the routines.</td>
</tr>
</tbody>
</table>
Command | Description
--- | ---
- Import | Starts Import Export Manager.
- Export | Starts Import Export Manager.
- Delete | Deletes the selected object.
- Copy | Copies the selected object to the clipboard.
- Rename | Renames the selected object.
- Update | Updates the selected libraries to the most recent version.

View
- Project Library Usage per Library | Displays a report showing the registered library usage for the selected library. Available only at a library branch.
- Network Layout | Generates Network Layout report.
- Module I/O Schedule for Rack Module | Displays the I/O Schedule showing rack assignments.
- Chassis Layout | Displays or prints a report with the I/O Modules in the Controller (local) chassis. Pastes the contents of the clipboard to the selected Controller with options.
- Add New Instance | Adds a new task in the Class View pane.
- Object References | Displays the Object References report.
- Exclude AOI (Add On Instructions) | Excludes the AOI (included by default). Only available in the Add-On Instructions branch. When two AOIs with same major and minor revision, but different extended text are added in the Controller Preview tree, AOI conflict shows. You can use Exclude AOI to select which one to use.
- Include AOI (Add On Instructions) | Includes the AOI. Only available in the Add-On Instructions branch.
- Refresh | Refreshes the tree view.

See also
- Generate a controller file on page 54
- Merge controllers on page 60
- Add a hardware module on page 61
- Add a new software object to a controller on page 64

Generate a controller file

After configuring controller code in Application Code Manager, output Logix Designer files with the controller code that for use with the controller.

To generate a controller file

1. In the Controller Preview or Class View pane, right-click the Controller folder or the controller object and then click Generate Controllers.

2. In the Logix Code Generation dialog box, select the Generate and ACM Project Data check boxes.
Configure controllers

Chapter 5

- Select the **Overwrite Existing** check box to replace an earlier controller file with the same name.
- Select **Create ACD** to create a Logix Designer Project file (.ACD) in addition to the Logix Designer XML file (.L5X).

**Important:** Generating controller files with **ACM Project Data** selected produces the most complete record of the controller data. Always select ACM Project Data when generating a controller file as part of a disaster recovery backup.

3. In **Save Path**, type the path where the controller files should be saved or use the default path:

   \C:\Users\<user name>\AppData\Local\Rockwell Automation\Application Code Manager\Output

   The variable <user name> is replaced by the logged in user account name.

4. Click **Generate**.

   The controller files are generated. A status message displays when completed.

5. (optional) Click **Open Folder** to open the file explorer to the controller file location.

**See also**

- [Navigate the Application Code Manager user interface](#) on page 19
- [Generate a partial program](#) on page 64
- [Generate a partial routine](#) on page 65

---

**Add a new controller**

Use the **Object Configuration Wizard** to add a controller from a predefined controller library that resides in the registered libraries, configure its parameters, and include it in the ACM design output.

**Note:** The ability to add multiple controllers to a project or open an existing project containing multiple controllers is only available if the product has a Standard activation license and is not available in Lite mode.

**To add a controller**

1. In the **Controller Preview** or **Class View** pane, right-click the **Controllers** folder and then click **Add New**.

   The **Object Configuration Wizard** displays. Controller objects display in category groups. A triangular marker next to the column name
denotes the current category grouping displayed. Change category groups by clicking a column heading.

2. Click the + symbol to expand a category and display the controllers in that group.

Controllers registered in the connected ACM database are listed.

3. Click the desired controller and click Next.

4. In Name, enter a name for the controller.

5. (optional) In Description, modify the descriptive text as needed to help identify the controller.

6. In the Parameters tab, adjust Controller, HMI, Historian, and Motion parameters as needed.

7. Click Finish.

The controller adds to the Controller Preview and Class View trees.

See also

- Connect to an ACM database on page 23
- Add a controller from an ACD or L5X file on page 57
- Update using an ACD or L5X file on page 58

For projects in the same database, Copy To and Move To allow you to copy or move a controller from one project to another project from the Class View pane.

Note: This feature is only available if the product has a Standard activation license and is not available in Lite mode.

To move or copy a controller to another project

1. In the Class View pane, expand Controllers.

2. Right-click the controller that will be copied or moved to another project.

   - Select Copy To.

     When selecting Copy To, the controller will be copied and pasted to the destination project.
• Select Move To.

When selecting Move To, the controller will be cut and pasted to the destination project. When the controller has unresolvable references, they will be cleared.

3. In Select Destination Project, select an existing project or create a new project as the destination project.

When there are name conflicts between the moved controller and the controller in the existing project, rename the destination controller in Rename Destination Controller.

4. Click OK.

• When prompted, select OK.
• Once the Log File Viewer opens, track the errors, warning, informational, and debug events.

See also

Configure controllers on page 53
Select Destination Project dialog box on page 57
Add a new controller on page 55

Select Destination Project dialog box

The following table shows the options in the Select Destination Project dialog box.

<table>
<thead>
<tr>
<th>Options</th>
<th>Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Displays the name of the destination project.</td>
</tr>
<tr>
<td>Description</td>
<td>Displays the description of the destination project.</td>
</tr>
<tr>
<td>New</td>
<td>Opens the Object Configuration Wizard dialog box to create a new project.</td>
</tr>
<tr>
<td>OK</td>
<td>Starts the copying or moving process.</td>
</tr>
<tr>
<td>Cancel</td>
<td>Returns the home page.</td>
</tr>
</tbody>
</table>

See also

Move or copy a controller to another project on page 56

Configure controllers on page 53

Add a controller from an ACD or L5X file

Add controllers to the Application Code Manager project from an existing Logix Designer XML file (.L5X) or Logix Designer Project file (.ACD).
To add a controller from an ACD or L5X file

1. In the Controller Preview or Class View pane, right-click the Controller folder and then click Add New from ACD/L5X.

2. In the Select an ACD/L5X File dialog box, select the file to add then click Open.

3. In the Import From ACD Wizard - Instance Import Actions page review the items listed. Items with Add in the Action column are imported.

4. If any objects should not be imported, click Add and then select Exclude.

5. Select Next. The Import dialog displays. The information grid shows any errors, warning, info, or debug messages generated during the import. Review any warnings or errors listed.

   • Click View Changes to see a comparison of the original and updated information in an Excel spreadsheet.
   • Information buttons show how many of each type of message are present. Filter the displayed messages by clicking the button for the message type to toggle the display of those messages.

6. Select Next. The Import From ACD Wizard - Content Imported page displays.

7. Select Finish.

   The new object will appear in the Controller Preview pane and Class View pane.

See also

   Update using an ACD or L5X file on page 58
   Detach from ACD/L5X files on page 60
   Merge controllers on page 60

Update using an ACD or L5X file

Update controller configurations in an Application Code Manager project using information from a Logix Designer XML file (.L5X) or a Logix Designer Project file (.ACD).

To update a controller using an ACD/L5X file

1. In the Controller Preview or Class View pane right-click the controller to update and select Update from ACD/L5X.
2. In **Select an ACD/L5X File** choose the ACD or L5X file to use to update the controller.

   The **Import from ACD Wizard - Instance Import Actions** page displays.

   - Objects that are present in the file but not in the target controller will be selected to be added by default. If any of these objects should not be added select **Exclude** from the **Action** column.

   - Objects that already exist in the target controller will default to being excluded from the update. If any of these objects should be updated using the information in the file select **Update** from the **Action** column.

   **Note:** When excluding an object, information in the ACM project is preserved. When an update is performed, information from the ACD or L5X overwrites the ACM project information.

3. Select **Next**. The import operation proceeds. After the operation is finished the **Import Complete** page displays. The information grid shows any error, warning, info, or debug messages generated during the import. Review any **Warnings** or **Errors** listed.

   - Click **View Changes** to see a comparison of the original and updated information in an Excel spreadsheet.

   - Information buttons show how many of each type of message are present. Filter the displayed messages by clicking the button for the message type to toggle the display of those messages.

4. Select **Finish**. The **Import From ACD Wizard - Content Imported** dialog appears. If one or more objects are set to update, then **Update the Original Snapshot** will be checked by default. Clear the check box to prevent the snapshot information from updating the current configuration and then click **Finish** to close the wizard.

   **Note:** Snapshot information is used for the original ACM project data when performing the merge. If all items are excluded, by default **Update the Original Snapshot** is not checked.

5. If **Update the Original Snapshot** was selected click **Next**. The **Import From ACD Wizard - Original Snapshot Updated** confirmation page displays, click **Finish** to close the wizard.

See also

- [Detach from ACD/L5X Files](#) on page 60
- [Reconstitute an ACD file](#) on page 49
- [Generate a controller file](#) on page 54
Detach from ACD/L5X files

If a controller is updated using an ACD or L5X file then the controller will maintain an attachment to that file. The next time ACM is started, the ACM application will attempt to establish a connection to the file. Depending on file systems, file size, and file location establishing a connection to an ACD or L5X file may take more time than expected. If needed, the ACM application can detach the file from the controller.

**To detach from an ACD/L5X file**

- In the Controller Preview or Class View pane, right-click the controller to which the file is attached to and then select Detach from ACD/L5X.

See also

- Merge controllers on page 60
- Update using an ACD or L5X file on page 58
- Add a controller from an ACD or L5X file on page 57

Contribute instances

Use Contributing instances to view and edit the properties of an object instance.

**Note:** This feature is only available if the product has a Standard activation license and is not available in Lite mode.

**To contribute an instance**

1. In the Controller Preview pane, expand Controllers.
2. Right-click a node shared by multiple instances, and then select Contributing instances.
   A list of contributing instances is shown.
3. Select an instance.
   Its parameters property panel opens.

See also

- Configure controllers on page 53
- Add a new instance on page 63

Merge controllers

Use Merge Controller to bring new objects from an ACM project into a new ACD file.
To merge controllers

1. In either the ClassView or Controller Preview pane right-click the object to merge.

2. Select Merge Controller. The Logix Designer Merge Tool displays.

   The Updated Library Objects pane displays the objects that will be merged.

3. Select Next. The Save As dialog displays.

4. Select Save.

   **Note:** After the merge file has been created, confirm all information is correct prior to deploying.

See also

- Generate a controller file on page 54
- Add a hardware module on page 61
- Add a new software object to a controller on page 64

Add a hardware module

Add hardware modules to your controller configuration in the I/O Configuration branch in the Controller Preview pane.

To add a hardware module

1. In the Controller Preview pane, expand the Controllers node, expand the controller object, expand the I/O Configuration folder and then right-click Backplane and then click Add New.

   The Object Configuration Wizard displays.

2. Click the + symbol to expand the library category and display the hardware module libraries registered in the connected ACM database.

3. Select the library that will be copied to create the new hardware module and click Next.

4. (optional) In Description, type a unique description of the hardware module or accept the default description.

5. Click Finish.

   The hardware module object appears in the Controller Preview pane under Backplane.
See also

Use Copy and Paste Special on page 62

Add a new instance on page 63

Add a new software object to a controller on page 64

Delete a hardware module

If a module is no longer relevant to the application it can be deleted from the project.

To delete a module

1. In the Class View pane, right-click the module and then select Delete.

2. In the Delete dialog box, click Yes to confirm that the module should be deleted.

3. If the object is referenced by other objects the Delete Action dialog box displays.
   a. Click Clear to remove all references to the module being deleted.
   b. Click OK to delete the module.

The module is removed from the Class View and Controller Preview panes.

See also

Add a hardware module on page 61

Use Copy and Paste Special

Copy and Paste Special provides the ability to decide whether to include children, sub-objects, and reference values when pasting a copied object. If an object is simply pasted all of the associated items are copied and pasted along with the object. The Copy and Paste Special command is only available from the Class View pane.

To use Copy and Paste Special

1. Select the object that will be copied to create the new object. Right-click and click Copy.
   
   • Right-click the destination of the copied object and click Paste Special.
   • Select the branch where the object will reside. Right-click and click Paste Special.

2. The Paste Special dialog box displays.
• Select the **Include Children** check box to include Communication Module Children.

• Select the **Include Sub-Objects** check box to include I/O Module Sub-Objects (example: Channels).

• Select the **Include IO Reference Values** check box to maintain references to I/O objects.

• Select the **Include Non-I/O Reference Values** to maintain references to other types of objects

3. Click **Paste**.

4. The **Rename** dialog displays. The object names that must be changed to prevent a naming conflict are shown in red.

• Enter a new name when a naming conflict exists. When there are no conflicts, the name is black in color.

• Click **OK** to paste the object.

**See also**

- [Delete a hardware module](#) on page 62
- [Configure controllers](#) on page 53

---

**Add a new instance**

Use **Add a new instance** to add a new task to an existing program or task

**To add a new instance**

1. In the **Class View** pane, right-click the program or task the new instance will be created from and then click **Add New Instance**. The **Object Configuration Wizard** is displayed.

2. (optional) In **Name**, type a new name for this instance.

3. (optional) In **Description**, type identifying information about this instance.

4. If adding a new instance from a program object associate the new instance with a task by choosing an item from the drop down list next to **Task**.

5. If adding a new instance from a routine object, associate the new instance with a task by choosing an item from the drop down list next to **Task**. The routine then needs to be added to a program within the task by choosing an item from the drop down list next to **Program**. The **Program** drop down list is pre-populated with programs available from the selected task.
6. Click **Finish** to add the new instance.

**See also**

- Configure controllers on page 53
- Add a new software object to a controller on page 64

**Add a new software object to a controller**

Add re-usable code for software objects (tasks, routines and programs) to the controller from the Application Code Manager library.

**To add a new software object to a controller**

1. In **Controller Preview**, expand **Controllers**, expand the controller object, right-click a task or program object, then click **Add New**.

2. In the **Object Configuration Wizard**, expand the library categories to display the software libraries registered in the ACM database.

3. Select the library object to add to the selected task or program, then select **Next**.

4. (optional) In **Description**, type a unique description of the software or accept the default description.

5. (optional) Edit software object parameters and fields in the tabs on the bottom half of the **Object Configuration Wizard**.

6. (optional) If the software object depends on other linked library objects, select the **Linked Libraries** tab, then select **Auto Create Linked Objects**. Select linked library creation options, then click **OK**.

7. Select **Finish**.

The software object appears in the **Controller Preview** and **Class View** panes.

**See also**

- Generate a partial program on page 64
- Generate a partial routine on page 65
- Object Configuration Wizard on page 66
- Auto Create Linked Objects on page 67

**Generate a partial program**

**Generate Partial Program** exports a Logix Designer XML file (L5X) that can be
imported into the Logix Designer as a program.

**To generate a partial program**

1. In the Controller Preview pane, expand Controllers, expand the controller object, right-click the program object to export and then click Generate Partial Program.

   The Logix Code Generation dialog box displays.

2. (optional) To rename the file click in the Save As field. The text box becomes editable and the current file name is highlighted. Type the new file name.

3. Verify that the Generate and ACM Project Data check boxes are selected.

   - Select the Overwrite Existing check box if you want to replace an earlier file with the same name.

   The Create ACD option is disabled. Partial program and routine items can only be generated using .L5X files.

4. In Save Path, type the path where the exported files should be saved or use the default path:

   ```
   C:\Users\<user name>\AppData\Local\Rockwell Automation\Application Code Manager\Output
   ```

   The variable `<user name>` is replaced by the logged in user account name.

5. Click Generate.

   The files are generated. A status message displays when completed.

6. (optional) Click Open Folder to open the file explorer to the file location.

**See also**

[Generate a partial routine on page 65](#)

---

**Generate a partial routine**

*Generate Partial Routine* allows you to create an export file (L5X) that can be imported into the Logix Designer as a routine.

**To generate a partial routine**

1. In the Controller Preview pane, expand Controllers, expand the controller object, right-click the routine object to export and then click Generate Partial Routine.
Chapter 5  Configure controllers

The Logix Code Generation dialog box displays.

2. (optional) To rename the file double-click in the Save As field. The text box becomes editable and the current file name is highlighted. Type the new file name.

3. Verify that the Generate and ACM Project Data check boxes are selected.
   - Select the Overwrite Existing check box if you want to replace an earlier file with the same name.
   
   The Create ACD option is disabled. Partial program and routine items can only be generated using .L5X files.

4. In Save Path, type the path where the exported files should be saved or use the default path:

   C:\Users\<user name>\AppData\Local\Rockwell Automation\Application Code Manager\Output

   The variable <user name> is replaced by the logged in user account name.

5. Click Generate.

   The files are generated. A status message displays when completed.

6. (optional) Click Open Folder to open the file explorer to the file location.

See also

Add a new instance on page 63

Object Configuration Wizard

How do I open the Object Configuration Wizard?

Use one of these methods:

- In Controller Preview, right-click a folder, task, program, or I/O configuration node in the project, then select Add New, Add New Task, or Add New Program.

- In Class View, right-click the project or library, then select Add New or Add New Instance.

Use the Object Configuration Wizard to add a library object, program, or task to a project. The Object Configuration Wizard consists of two pages, the Select a Library page and the Parameters page.
Configure controllers

Chapter 5

• **Select a Library** displays the latest revisions of library objects, separated by category. Select **Show All Revisions** to display all revisions. To quickly search for a library, enter criteria in **Filter**.

• **Parameters** displays the name, description, and configuration parameters for a library object, program, or task. Parameters may be broken up into object-specific tabs, such as SubObjects, Linked Libraries, or Interface Links. Editable fields appear black, non-editable fields appear gray.

  **Tip:** If a library object contains links to other linked library objects, these objects can be created automatically. Select the **Linked Libraries** tab (if applicable), then select **Auto Create Linked Objects**.

See also

[Add a new software object to a controller](#) on page 64

[Auto Create Linked Objects](#) on page 67

---

**Auto Create Linked Objects**

How do I open Auto Create Linked Objects?

1. In **Controller Preview**, expand **Controllers**, expand the controller object, right-click a task or program object, then click **Add New**.

2. In the **Object Configuration Wizard**, expand the library categories to display the software libraries registered in the ACM database.

3. Select the library object to add to the selected task or program, then select **Next**.

4. If the software object depends on other linked library objects, select the **Linked Libraries** tab, then select **Auto Create Linked Objects**.

Use **Auto Create Linked Objects** to add objects linked by a library object to a project, rather than manually adding them to the project one at a time.

Hardware modules cannot be Auto Created but can still be linked to if they already exist in the project.

---

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Link Name</strong></td>
<td>Displays the name of the link in the object being added to the project. <strong>Auto Create Linked Objects</strong> uses the link name to find a matching library.</td>
</tr>
<tr>
<td><strong>Library</strong></td>
<td>Displays the name and version of a library that matches the link name. The latest version of the library is always used. If an exact library match cannot be found, the <strong>Library</strong> column displays “No unique library matches link criteria”. The <strong>Action</strong> field is locked to <strong>Ignore</strong>.</td>
</tr>
</tbody>
</table>
| **Action** | Sets the action that **Auto Create Linked Objects** should take for the linked library:  
  • **Create New**: Creates a new library object with the parameters from the matched library object.  
  • **Use Existing**: Uses the matched library object.  
  • **Ignore**: Does not create or use the linked library object. Mandatory linked libraries cannot be set to **Ignore**. |
### Column Description

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object Name</td>
<td>Browse for an existing object name or set the auto created library object name. Disabled if the library object is parameterized. When an exact match library is found, the match icon displays beside the object name.</td>
</tr>
<tr>
<td>Task Name</td>
<td>Assigns the auto created library object to a task. Create a new task by selecting &lt;Add New...&gt;. Disabled if Action is set to Use Existing or Ignore.</td>
</tr>
<tr>
<td>Program Name</td>
<td>Assigns the auto created library object to a program. Create a new program by selecting &lt;Add New...&gt;. Disabled if Action is set to Use Existing or Ignore.</td>
</tr>
</tbody>
</table>

**See also**

- [Add a new software object to a controller](#) on page 64
- [Object Configuration Wizard](#) on page 66
Use ACM Tools

Use Application Code Manager tools to assist in reusable code development and management.

This table describes each of the tools.

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Import Export Manager</td>
<td>Use to import or export ACM project content between Excel spreadsheets (Schedule). Use for bulk additions, duplications, changes; comparing versions; snapshots; backup customer parameter entry, and/or transferring project contents.</td>
</tr>
<tr>
<td>Database Manager</td>
<td>Use to create, delete, upgrade, back up, and restore ACM databases.</td>
</tr>
<tr>
<td>Plugins</td>
<td>Add capabilities to Application Code Manager by installing plugins through the Tools &gt; Plugins command.</td>
</tr>
<tr>
<td>Log Debug Information</td>
<td>Choose to include debug information in the ACM Log File to help with troubleshooting applications. Information is written to the ACM log file when design outputs are generated (example: &lt;CLX&gt;, FactoryTalk View, FactoryTalk Historian, Word) or when schedules are imported or exported.</td>
</tr>
<tr>
<td>Log Viewer</td>
<td>Use to view the contents of the most recent ACM log file. A new ACM log file is created for each ACM session.</td>
</tr>
<tr>
<td>Settings</td>
<td>Use to set the location of the system documentation, the language, and the default settings for Control Logix controllers.</td>
</tr>
<tr>
<td>Open Target ACD</td>
<td>Opens the directory and allows navigation to the ACD file location. Use to instantiate library content into an existing file, versus creating a new ACD file.</td>
</tr>
</tbody>
</table>

See also

- Import Export Manager on page 69
- Database Manager on page 79
- ACM Default Settings on page 86
- Add ACM library content to an existing ACD project on page 87

Import Export Manager

The Import Export Manager imports and exports ACM project content to and from schedules using the Excel file format (.xlsx). Schedules include the scope of the project (example: Project, Controller, Task, Program, Object) and content for the project (example: Device List, Device Interlocks).

The Import Export Manager has the following controls:

<table>
<thead>
<tr>
<th>Control</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Menu Bar</td>
<td>Includes a File menu for identifying the current project and a Tools menu for setting up templates and configuring logging.</td>
</tr>
</tbody>
</table>
Control | Description
--- | ---
**Project:** | Displays the name of the current project selected. If creating a new project, this must be a unique name.

<table>
<thead>
<tr>
<th>Import Tab</th>
<th>Specifies what effect the import will have on the target project and the location of the schedule to import.</th>
</tr>
</thead>
<tbody>
<tr>
<td>New</td>
<td>Creates a new project using the specified project name.</td>
</tr>
<tr>
<td>Replace</td>
<td>Overwrites the specified project with the project being imported.</td>
</tr>
<tr>
<td>Update</td>
<td>Merges the information being imported into the project specified.</td>
</tr>
<tr>
<td>Auto Continue</td>
<td>Continues the import process automatically.</td>
</tr>
<tr>
<td>Excel File</td>
<td>Enters the path to the schedule Excel file (.xlsx) to import or select the ellipsis to browse to the file location.</td>
</tr>
<tr>
<td>Open Backup Folder</td>
<td>Opens the ACM backup folder.</td>
</tr>
<tr>
<td>Backup original</td>
<td>Creates a backup of the original project.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Export Tab</th>
<th>Defines the scope and details of the export schedule.</th>
</tr>
</thead>
<tbody>
<tr>
<td>All Projects</td>
<td>Exports all projects in the connected ACM database.</td>
</tr>
<tr>
<td>Complete Project</td>
<td>Exports the currently specified project.</td>
</tr>
<tr>
<td>Partial</td>
<td>Exports only specific objects and parameter values. To specify the objects to include in the export select the appropriate values from the Type and Controller drop-down menus.</td>
</tr>
<tr>
<td>Use Template</td>
<td>Limit the content of the schedule to either:</td>
</tr>
<tr>
<td>Local</td>
<td>Locally available devices.</td>
</tr>
<tr>
<td>Project</td>
<td>Devices used in the project being exported.</td>
</tr>
<tr>
<td>Global</td>
<td>All devices in the library.</td>
</tr>
<tr>
<td>Show Modified</td>
<td>Colors “changed” (not equal to default) and “unchanged” (equal to default) parameter values in the export file.</td>
</tr>
<tr>
<td>Export Used Libraries</td>
<td>Includes the libraries used by the projects being exported in the export schedule.</td>
</tr>
<tr>
<td>Export</td>
<td>Clicking this button performs the export as specified.</td>
</tr>
<tr>
<td>Export and Open</td>
<td>Clicking this button performs the export as specified and opens the exported schedule file. Not available when exporting all projects.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Compare Tab</th>
<th>Compares the currently selected project to a previously saved schedule file and creates a report.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Only Show Changes</td>
<td>When this item is enabled the comparison report will only include items that are different between the current project and the specified schedule file.</td>
</tr>
<tr>
<td>Excel File</td>
<td>Provides the path and file name of the schedule file to compare with the current project. Use the ellipsis (…) button to navigate to a file to open.</td>
</tr>
<tr>
<td>Compare</td>
<td>Clicking this button runs the compare operation. The results of the comparison are displayed in an Excel spreadsheet.</td>
</tr>
</tbody>
</table>

| Cancel | Closes the Import Export Manager. |
| Previous | Not available |
| Next | Not available |
| Finish | Closes the Import Export Manager. |
| | (Available after the import, export, or compare process is completed.) |

**See also**

[Import a schedule on page 71](#)

[Export a schedule on page 73](#)

[Compare a project to a saved schedule on page 75](#)
Create a new schedule template on page 76

Import a schedule

Use the **Import Export Manager Import** tab to import objects (instances) and parameter values from a schedule (xlsx). Verify that the correct database name displays in the ACM Title Bar before starting the import process.

**Tip:** When importing objects (instances), a compatible library (class) must be registered in the ACM database.

The Catalog Number of the registered library must match the Catalog Number in the schedule and a Library with a revision greater than or equal to the revision in the schedule must be registered.

If a library with the same revision is registered, the library with the same revision is used.

If a library with the same revision is not registered and a library with a greater revision is registered, the newest library is used.

**To import a schedule**

1. Click **Tools, Import Export Manager** to open the Import Export Manager. The most recently used project opens by default.

2. To choose a different project to import, click **File > Open** and then select the project from the list. If the ACM database does not contain any projects, the Project is blank.

3. Select the import method:

   - **New** - Create a new project.
     Use to bring projects into a new ACM database.
   - **Replace** - Overwrite project.
     Use to revert projects to a previously exported version.
   - **Update** - Merge with existing project.
     Use to add content from other projects, or earlier versions of the same project, to the current project.

4. Type a schedule file name (xlsx) in the **Excel File** text box or click the ellipsis to browse to the file.

5. Click **Import**.

   - If you choose **Replace** as the import method, the **Import** dialog box opens.
   - If you choose **New** or **Update** as the import method, the **Import Complete** dialog box opens.

Errors, warnings, info, and debug log entries display.
Tip: Filter the list of exceptions by clicking Errors, Warnings, Info, and/or Debug.

6. (optional) Click Show Log File on the Import Complete dialog box to display the entire contents of the most recent ACM log file in a text editor.

7. If you chose Replace as the import method, click Next to continue with the import and change the data in the ACM database. If you chose New or Update as the import method, click Finish to close the Import Complete dialog box.

8. Click Finish to close the Import Export Manager.

See also

Import Export Manager Import Tab settings on page 72

The Import Export Manager Import tab is used to import objects and parameter values from a schedule file.

This table describes the settings on the Import Export Manager Import Tab.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New – Create new project</td>
<td>Select this option to create a new project based on the imported schedule. The project name in the schedule cannot exist in the connected ACM database.</td>
</tr>
<tr>
<td>Replace – Overwrite project</td>
<td>Select this option to replace the specified project with the project in the imported schedule. The project name in the schedule must be the same as the project name specified in the Project text box. When this import option is run the current version of the specified project is exported to the ACM backup folder and deleted from the ACM database before importing the schedule to the project.</td>
</tr>
<tr>
<td>Update – Merge with existing project (will not delete existing objects)</td>
<td>Select this option to update the specified project with the information contained in the imported schedule. When this import option is run the current version of the specified project is exported to the ACM backup folder and deleted from the ACM database before merging the imported schedule with the project. Schedule content that is new will be added to the specified project while schedule content that is different between the project in the database and the project in the schedule will be updated to match the content from the schedule. However, schedule content that was deleted or was not exported is not deleted from the selected Project.</td>
</tr>
<tr>
<td>Auto-Continue</td>
<td>Select this checkbox to automatically continue the import process using the specified schedule without requiring additional confirmations if no errors or warnings are present.</td>
</tr>
<tr>
<td>Excel File:</td>
<td>Type the path to the schedule Excel file (.xlsx) to import or click the ellipsis to browse to the file location.</td>
</tr>
<tr>
<td>Open Backup Folder</td>
<td>Opens the ACM backup folder.</td>
</tr>
<tr>
<td>Backup original</td>
<td>Creates a backup of the original project.</td>
</tr>
<tr>
<td>Import</td>
<td>Imports the specified schedule file to the specified project. The Import button is available when the import configuration is valid.</td>
</tr>
</tbody>
</table>

See also

Import a schedule on page 71

Export a schedule on page 73
Export a schedule

Use the Import Export Manager Export tab to export objects (instances) and parameter values to a schedule file. Schedules are primarily used for backup and recovery. Verify that the correct database name displays in the ACM title bar before starting the export process.

To export a schedule

1. Click Tools, Import Export Manager to open the Import Export Manager. The most recently used project opens by default.

2. To choose a different project to export, click File > Open and then select the project from the list.

3. Click the Export tab.

4. Configure the scope of the schedule:
   - **All Projects**
     Use to export all projects in the connected ACM database.
   - **Complete Project**
     Use to export the currently specified project.
   - **Partial**
     Use to export only specific objects and parameter values.
     - Under Type, choose either Controller, Hardware, Object, or Library, then specify the controller and instance to export.

5. (optional) Limit the content (example: Device List, Device Interlocks) of the schedule by selecting the Use Template check box. Under the Type category the following options are available:
   - Choose Local to limit the schedule to locally available devices.
   - Choose Project to limit the schedule to only those devices used in the project being exported.
   - Choose Global to include all devices in the library.

6. (optional) Select Show Modified to color code "changed" (not equal to default) and "unchanged" (equal to default) parameter values.

7. (optional) Select Export Used Libraries to include the libraries in the schedule.

8. Click Export.

9. Enter a path and file name in the Save As dialog.
10. Click **Save** to save the schedule as an Excel Workbook (*.xlsx) file.

11. If there is an error or warning, the ACM Log File entries are displayed when the export is complete.

   **Tip:** Errors, warnings, info, and debug information are displayed by default. Filter the list of exceptions by clicking **Errors**, **Warnings**, **Info**, and/or **Debug**.

   Clicking the **Show Log File** command will display the contents of the most recent ACM Log File.

   Click **Finish** to return to the **Import Export Manager**.

12. Click **Finish** to close the **Import Export Manager**.

**See also**

**Import Export Manager Export tab settings** on page 74

**Import Export Manager Export tab settings**

The **Import Export Manager Export** tab is used to export objects and parameter values to a schedule file.

This table describes the settings on the **Import Export Manager Export Tab**.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>All Projects</td>
<td>Select this option to export a schedule for each project in the connected ACM database. A template cannot be applied.</td>
</tr>
<tr>
<td>Complete Project</td>
<td>Select this option to export a schedule for the current project. A template can be applied to the export.</td>
</tr>
<tr>
<td>Partial</td>
<td>Select this option to export a schedule that includes only selected information from the current project. With this option, these types of items can be specifically exported: Controller, Hardware, Object, Library</td>
</tr>
<tr>
<td>Use Template</td>
<td>Select this check box to limit the items exported to the schedule using a schedule template. When selected, choose to use either a <strong>Local</strong>, <strong>Project</strong>, or <strong>Global</strong> schedule template.</td>
</tr>
<tr>
<td>Show Modified</td>
<td>Select this check box to apply color codes in the schedule to denote “changed” (not equal to default) and “unchanged” (equal to default) parameter values.</td>
</tr>
<tr>
<td>Export Used Libraries</td>
<td>Includes all the associated library files for the project/controller in the export.</td>
</tr>
<tr>
<td>Export</td>
<td>Exports the schedule. The <strong>Export</strong> button is available when a valid export configuration has been selected.</td>
</tr>
<tr>
<td>Export and Open</td>
<td>Exports the schedule and opens the schedule in Excel when the export is complete. The <strong>Export and Open</strong> button is available when a valid <strong>Complete Project</strong> or <strong>Partial</strong> export configuration has been selected.</td>
</tr>
</tbody>
</table>

**See also**

**Import Export Template Manager** on page 77
Compare a project to a saved schedule

Use the Import Export Manager Compare tab to compare the current project to a previously saved schedule file and create a report that highlights the differences.

To compare a project to a saved schedule

1. Click Tools, Import Export Manager to open the Import Export Manager. The most recently used project is opened by default.

2. To choose a different project to compare with, click File > Open and then select the project from the list.

3. Click the Compare tab.

4. In Excel File type the name of the schedule file to use for comparison or click the ellipsis to browse to the schedule file.

5. (optional) Select Only Show Changes to limit the content of the comparison report to items in the current project that differ from the schedule.

6. Click Compare.

The comparison report opens in Excel.

7. Click Finish to close the Import Export Manager.

See also

Export a schedule on page 73

Import Export Manager Compare Tab settings on page 75

Import Export Manager Compare Tab settings

Use the Import Export Manager Compare tab to compare the current project to a previously saved schedule.

This table describes the settings on the Import Export Manager Compare Tab.

<table>
<thead>
<tr>
<th>Control</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Only Show Changes</td>
<td>Select this check box to create a report that only lists the project content that has changed.</td>
</tr>
<tr>
<td>Excel File:</td>
<td>Click the ellipsis to identify the schedule file (*.xlsx) to compare with the current project.</td>
</tr>
<tr>
<td>Compare</td>
<td>Compares the current project to the selected schedule file. The Compare button is available when a valid schedule has been selected.</td>
</tr>
<tr>
<td>Cancel</td>
<td>Cancels the comparison. The Cancel button is available once the comparison is in progress.</td>
</tr>
<tr>
<td>Previous</td>
<td>Not used</td>
</tr>
</tbody>
</table>
Create a new schedule template

Use the Template Manager tool of the Import Export Manager to create schedule templates. Schedule templates determine the scope of the schedule (local, project, or global), the library objects used, and the parameters that are visible.

To create a new schedule template

1. Click Tools > Import Export Manager.

The Import Export Manager appears.

2. In the Import Export Manager, click Tools > Import Export Template Editor.

The Template Manager appears with the Template Editor tab displayed.

3. In the Template area, make the following selections:
   a. Select the scope of the template:
      • Local
      • Project
      • Global
   b. (if project scoped) Click the down arrow and select the project name.
   c. Click the down arrow and select the schedule template.

4. Click New. The Template Name dialog box appears.

5. In New Template Name, enter a name for the new schedule template and click OK.

The new schedule template name appears in the Template area.

6. In the Library Section Selection area, add the object and sub-object parameters to the new schedule template.
   a. Select an object in the Library Section Selection area.
b. In the **Not Visible Parameters** area select the parameters to include.

   Click **Select All** to select all of the parameters for an object.

c. Click **Add** >

d. Repeat as needed.

e. In the **Visible Parameters** area, order the parameters as they should appear in the template file.

7. Click **Save** to save the changes.

8. Click **Reload** to cancel all edits made since the last time the **Save** command was executed.

9. Click **Finish** to close the **Import Export Template Manager**.

See also

- **Import a schedule** on page 71

---

**Import Export Template Manager**

How do I open the Template Manager?

1. Click **Tools, Import Export Manager** to open the **Import Export Manager**.

2. In the **Import Export Manager** window, click **Tools > Import Export Template Editor**.

The **Template Manager** can be used to create custom schedule templates, copy schedule templates, or move schedule templates from one location to another.

Three schedule template locations are available:

- **Local** – Located in the Windows User Folder. Available only to the ACM User.

- **Project** – Located in the ACM database. Available to all ACM users with this project currently open.

- **Global** – Located in the ACM database. Available to all ACM users connected to this ACM database.

See also

- **Template Manager Template Editor tab settings** on page 78

- **Template Manager Copy/Move Templates tab settings** on page 78
### Template Manager Copy/Move Templates tab settings

The Copy/Move Templates tab is used to copy or move a template. This table describes the settings on the Copy/Move Templates tab.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source Template</td>
<td>Selects a schedule template. Select a schedule template location, a project name (if located in a project), and a schedule name.</td>
</tr>
<tr>
<td>Rename</td>
<td>Rename the source template</td>
</tr>
<tr>
<td>Duplicate</td>
<td>Duplicate the source template</td>
</tr>
<tr>
<td>Delete</td>
<td>Delete the source template</td>
</tr>
<tr>
<td>Copy</td>
<td>Copies the source template to the location specified in Destination Template.</td>
</tr>
<tr>
<td>Move</td>
<td>Moves the source template to the location specified in Destination Template.</td>
</tr>
<tr>
<td>Destination Template</td>
<td>Selects a location to move or copy the selected source template to. Choose either:</td>
</tr>
<tr>
<td></td>
<td>• Global - Places the template in the ACM database and makes it available to all users</td>
</tr>
<tr>
<td></td>
<td>• Project - Places the template in the ACM database and makes it available when the specified project is opened.</td>
</tr>
<tr>
<td></td>
<td>• Local - Places the template in the locate user template folder on the local computer. The template is only available to the local user.</td>
</tr>
<tr>
<td>Open User Template Folder</td>
<td>Opens the file browser scoped to the local user template folder.</td>
</tr>
<tr>
<td>Finish</td>
<td>Close the Template Manager and returns to the Import Export Manager.</td>
</tr>
</tbody>
</table>

**See also**

Import Export Template Manager on page 77

### Template Manager Template Editor tab settings

Use the Template Editor tab of the Template Manager to create, delete, or edit schedule templates. This table describes the settings on the Template Editor tab.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Template</td>
<td>Selects a schedule template. Use the combo boxes to select a schedule template location, a project name (if located in a project), and a schedule name.</td>
</tr>
<tr>
<td></td>
<td>Click New to create a new schedule template</td>
</tr>
<tr>
<td></td>
<td>Click Delete to delete the selected schedule template.</td>
</tr>
<tr>
<td></td>
<td>If the schedule template format is old, Template Manager asks for permission to upgrade the template format to the latest version when the schedule template is selected.</td>
</tr>
<tr>
<td>Library Section Selection</td>
<td>Selects a library object or sub-object definition from the registered libraries. The object or sub-object parameters display in the Not Visible Parameters or Visible Parameters list boxes.</td>
</tr>
<tr>
<td>Not Visible Parameters</td>
<td>Displays the object or sub-object parameters that are not be included in schedules exported using this schedule template.</td>
</tr>
<tr>
<td></td>
<td>To include a parameter in the exported schedule, select the item from the Not Visible Parameters list and then click Add to move it to the Visible Parameters list.</td>
</tr>
<tr>
<td></td>
<td>Use the mouse key combinations Ctrl-click, Shift-Click, Click-and-Drag, and the Select All and Deselect All buttons to select multiple parameters.</td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Visible Parameters</td>
<td>Displays the object or sub-object parameters included in schedules exported using this schedule template. To remove a parameter from the exported schedule, select the item from the Visible Parameters list and then click Remove to move it to the Not Visible Parameters list. Use the mouse key combinations Ctrl-click, Shift-Click, Click-and-Drag, and the Select All, and Deselect All buttons to select multiple parameters. Use the Up, Down, Top, and Bottom commands to control the order in which the parameters display in the exported schedule. Use the sort buttons to display the parameters in category groups or alphabetically.</td>
</tr>
<tr>
<td>Add &gt;</td>
<td>Moves the parameter selected in the Not Visible Parameter list to the Visible Parameter list.</td>
</tr>
<tr>
<td>&lt; Remove</td>
<td>Moves the parameter selected in the Visible Parameter list to the Not Visible Parameter list.</td>
</tr>
<tr>
<td>Save</td>
<td>Saves the schedule template to the schedule template settings file (C:\Documents and Settings&lt;username&gt;\Local Settings\Application Data\Rockwell Automation\Application Code Manager \Templates&lt;template&gt; ).</td>
</tr>
<tr>
<td>Reload</td>
<td>Cancels any template changes made since the last Save command was executed by reloading the schedule template from the schedule template settings file (C:\Documents and Settings&lt;username&gt;\Local Settings\Application Data\Rockwell Automation\Application Code Manager \Templates&lt;template&gt; ).</td>
</tr>
<tr>
<td>Finish</td>
<td>Closes Template Manager.</td>
</tr>
</tbody>
</table>

### See Also

- Template Manager Copy/Move Templates tab settings on page 78

### Database Manager

The **Database Manager** controls how Application Code Manager interacts with the SQL Server database. It also provides the ability to create, backup, and recover databases.

### See also

- Create an ACM database on page 79
- Upgrade an ACM database on page 80
- Backup an ACM database on page 81
- Restore an ACM database on page 82
- Database Manager settings on page 84

### Create an ACM database

Application Code Manager (ACM) uses a database to store re-usable code. At least one database is required, but additional databases can be created as needed.

**To create an ACM database**

1. Open Application Code Manager, in the main menu, click **Tools > Database Manager**.
2. In **Server Name**, select the computer name and SQL Server instance from the drop down list or type a computer name and SQL Server instance in the following format:

   `<Computer Name> \ <SQL Server Instance>`

3. In **Log on to the server**, type the **User name** and **Password** to use to authenticate the connection to the SQL Server, then click **Connect**.

   **Status** updates to **Connected**.

4. In **Specify the database**, type a unique name for the ACM database.

5. In **Actions**, select **Create database** and then click **Execute Task**.

   Once the database is created an **Action completed successfully** message displays. Click **OK**.

6. Click **Close** to close **Database Manager**.

**See also**

[Connecting to an ACM database on page 24](#)

---

### Upgrade an ACM database

Application Code Manager (ACM) uses a database to store re-usable code. If ACM is upgraded to a newer version the existing ACM database can still be used, but the database schema must be upgraded.

**To upgrade an ACM database**

1. Open Application Code Manager, in the main menu, click **Tools > Database Manager**

2. In **Server Name**, select the computer name and SQL Server instance from the drop down list or type a computer name and SQL Server instance in the following format:

   `<Computer Name> \ <SQL Server Instance>`

3. In **Log on to the server**, type the **User name** and **Password** to use to authenticate the connection to the SQL Server, then click **Connect**.

   **Status** updates to **Connected**.

4. In **Specify the database**, select the name of the ACM database you want to upgrade.

5. In **Actions**, select **Upgrade database** and then click **Execute Task**.
Once the database is created an **Action completed successfully** message displays. Click **OK**.

6. Click **Close** to close **Database Manager**.

**See also**

- [Connecting to an ACM database](#) on page 24
- [Upgrade the Application Code Manager application](#) on page 17
- [Backup an ACM database](#) on page 81

**Backup an ACM database**

Application Code Manager (ACM) uses a database to store re-usable code. Making a backup of the ACM database on a regular schedule is recommended to ensure that your re-usable code library can be restored in case of hardware failure or unintended changes to the data.

### Important:

When backing up a remote database, the file location must be accessible with read and write access granted to the user account under which the remote database service runs.

**To backup an ACM database**

1. Open Application Code Manager, in the main menu, click **Tools > Database Manager**

2. In **Server Name**, select the computer name and SQL Server instance from the drop down list or type a computer name and SQL Server instance in the following format:

   <Computer Name> \ <SQL Server Instance>

3. In **Log on to the server**, type the **User name** and **Password** to use to authenticate the connection to the SQL Server, then click **Connect**. **Status** updates to **Connected**.

4. In **Specify the database**, select the ACM database you want to backup.

5. In **Actions**, select **Backup database** and then click **Execute Recovery**. The **Select file to backup database** dialog box opens.

6. In **File name** type a name for the backup file being created and then click **Save**.

   Once the backup file is created an **Action completed successfully** message displays. Click **OK**.
7. Click **Close** to close **Database Manager**.

**See also**

Connecting to an ACM database on page 24

---

**Restore an ACM database**

Application Code Manager (ACM) uses a database to store re-usable code. If ACM is being moved to new database server or if the existing database server has incorrect ACM data, the backup made of the database can be restored to the database server so that all of the previous data is available.

---

**Important:**

When restoring a remote database, the file location must be accessible with read and write access granted to the user account under which the remote database service runs.

Restoring an ACM database backup onto a machine other than the one on which it was created will require database ownership to be set. This is because the user associated with the backed-up database does not exist in the new environment. If ownership is not set, then ACM users will continuously receive the following message: “Timed out waiting for the heart beat notification from the SQL Broker from within the ACM application. Please check the log file for details”.

Database ownership can be set by the database administrator. The database administrator will need to run the script as below within either a SQL Management Studio query window or via a SQLCMD session:

```
ALTER AUTHORIZATION on DATABASE: [Database Name] TO [SQL User]
```

- The `[Database Name]` section must be replaced with the ACM database name.
- The `[SQL User]` section must be replaced with the name of the user who will be assigned ownership of the database.

Running the above script ensures that you will no longer receives the timeout messages.

---

**To restore an ACM database**

1. Open Application Code Manager, in the main menu, click **Tools > Database Manager**

2. In **Server Name**, select the computer name and SQL Server instance from the drop down list or type a computer name and SQL Server instance in the following format:

   `<Computer Name> \ <SQL Server Instance>`

3. In **Log on to the server**, type the **User name** and **Password** to use to authenticate the connection to the SQL Server, then click **Connect**.

   **Status** updates to **Connected**.

4. In **Specify the database**, select the name of the ACM database on which you want to restore the data.
Tip: If you are restoring databases to a new database server or if the database is not present on the database server, create the database before restoring the data to it.

5. In **Actions**, select **Restore database** and then click **Execute Recovery**. The **Selected database backup to restore** dialog box opens.

6. Select the database backup file (.bak) to restore and then click **Open**.

7. The **Confirm database restore** dialog box opens warning you that the current database will be overwritten and asking you if you are sure you want to restore the database.

   Click **Yes** to confirm the restoration process.

8. Once the database is restored an **Action completed successfully** message displays. Click **OK**.

9. Click **Close** to close **Database Manager**.

See also

**Connecting to an ACM database** on page 24

---

Delete an ACM database

Application Code Manager (ACM) uses a database to store re-usable code. At least one database is required.

**To delete an ACM database**

1. Open Application Code Manager, in the main menu, click **Tools > Database Manager**

2. In **Server Name**, select the computer name and SQL Server instance from the drop down list or type a computer name and SQL Server instance in the following format:

   `<Computer Name> \ <SQL Server Instance>`

3. In **Log on to the server**, type the **User name** and **Password** to use to authenticate the connection to the SQL Server, then click **Connect**.

   **Status** updates to **Connected**.

4. In **Specify the database**, select the ACM database to delete.

5. In **Actions**, select **Delete database** and then click **Execute Task**. The **Confirm database delete** dialog box opens.

6. Click **Yes** to confirm deletion of the database
Once the database is deleted an **Action completed successfully** message displays. Click **OK**.

7. Click **Close** to close **Database Manager**.

**See also**

- [Create an ACM database](#) on page 79
- [Restore an ACM database](#) on page 82

**Database Manager settings**

How do I open the Database Manager dialog box?

- On the main menu, click **Tools > Database Manager**.

The **Database Manager** dialog box contains the settings that control how Application Code Manager interacts with the SQL Server database. It also provides the ability to create, backup, and recover databases.

This table describes the settings in the **Database Manager** dialog box.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Server name:</strong></td>
<td>Identifies the server that is hosting the SQL Server used by ACM. Use the drop-down list to select a computer name and SQL Server instance from, or enter a computer name and SQL Server instance in the following format: *&lt;Computer Name&gt;* \*&lt;SQL Server Instance&gt;*</td>
</tr>
<tr>
<td><strong>Refresh</strong></td>
<td>Queries the network for a list of SQL Server computer names and instances and updates the selections available in the drop-down list with the results.</td>
</tr>
<tr>
<td><strong>Log on to the server</strong></td>
<td></td>
</tr>
<tr>
<td>Using SQL Server Authentication</td>
<td>(Note: This account and password are configured during installation of SQL Server.)</td>
</tr>
<tr>
<td><strong>User name:</strong></td>
<td>The SQL Server user name associated with the ACM database. The default user name is “sa”.</td>
</tr>
<tr>
<td><strong>Password:</strong></td>
<td>The SQL Server password associated with the user name.</td>
</tr>
<tr>
<td><strong>ACM default sa password</strong></td>
<td>Use the default ACM password instead of typing a password. Check this box if the default SQL Server password was entered when SQL Server was installed.</td>
</tr>
<tr>
<td><strong>Connect</strong></td>
<td>When this button is clicked Application Code Manager attempts to connect to the database named in the <strong>Select a database name</strong> on the SQL Server entered in the <strong>Server name</strong> combo box.</td>
</tr>
<tr>
<td><strong>Status</strong></td>
<td>Displays the result of the database connection attempt.</td>
</tr>
<tr>
<td></td>
<td>• Connected</td>
</tr>
<tr>
<td></td>
<td>• Login failed for user</td>
</tr>
<tr>
<td><strong>Specify the database</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Select a database name</strong></td>
<td>Identifies the database to manage. Enter a name or select an existing name from the list. If the name entered is not present on the SQL Server, the option is provided to create a new database.</td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Create database</td>
<td>Creates a database with the database name entered in the Select a database name on the SQL Server entered in the Server name when Execute Task is clicked.</td>
</tr>
<tr>
<td>Upgrade database</td>
<td>Upgrades the ACM database named in the Select a database name combo box with the current schema when Execute Task is clicked.</td>
</tr>
<tr>
<td>Delete database</td>
<td>Deletes the database named in Select a database name from the SQL Server identified in Server name when Execute Task is clicked.</td>
</tr>
<tr>
<td>Execute Task</td>
<td>Performs the selected task.</td>
</tr>
</tbody>
</table>

**Actions – Disaster Recovery**

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backup database</td>
<td>Makes a backup copy of the database file named in Select a database name when Execute Recovery is clicked.</td>
</tr>
<tr>
<td>Restore database</td>
<td>Replaces the database file named in Select a database name combo box with the backup database file when Execute Recovery is clicked.</td>
</tr>
<tr>
<td>Execute Recovery</td>
<td>Performs the selected recovery.</td>
</tr>
</tbody>
</table>

**See also**

- Create an ACM database on page 79
- Upgrade an ACM database on page 80
- Delete an ACM database on page 83
- Backup an ACM database on page 81
- Restore an ACM database on page 82

**Log Debug Information**

To assist in troubleshooting Application Code Manager enable logging of debug information to the ACM Log File.

Information is written to the ACM log file when design outputs are generated (example: <CLX>, FactoryTalk View, FactoryTalk Historian, Word) or when schedules are imported or exported.

Once the additional logging information is no longer needed, disable log debug information.

**To log debug information**

- Click Tools > Log Debug Information.

A check mark appears next to the Log Debug Information entry in the Tools menu. Clicking the item again disables logging of debug information.

**See also**

- Use ACM Tools on page 69
**Log File Viewer**

The **Log File Viewer** provides a quick way to view the most recent events sent to the ACM log file. The Log File Viewer can be opened from the Application Code Manager **Tools** menu and from the **Import Export Manager Tools** menu.

The **Log File Viewer** displays errors, warning, informational, and debug events in a grid.

At the top of the **Log File Viewer** is a set of buttons listing the type of log entries and the number of entries for each type:

<table>
<thead>
<tr>
<th>Type</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error</td>
<td>2</td>
</tr>
<tr>
<td>Warning</td>
<td>12</td>
</tr>
<tr>
<td>Info</td>
<td>236</td>
</tr>
<tr>
<td>Debug</td>
<td>97</td>
</tr>
</tbody>
</table>

The entries displayed in the event grid can be filtered by type of events by clicking the button corresponding to the item to remove from the list.

When an item is not included in the view the button is shaded gray:

<table>
<thead>
<tr>
<th>Type</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error</td>
<td>2</td>
</tr>
<tr>
<td>Warning</td>
<td>12</td>
</tr>
<tr>
<td>Info</td>
<td>236</td>
</tr>
<tr>
<td>Debug</td>
<td>101</td>
</tr>
</tbody>
</table>

By default, the **Log File Viewer** displays events in ascending date time order.

The order can be changed to be either ascending or descending and be based on any of the columns displayed in the events grid. Click on the column heading to sort the display by **Type**, **Date/Time**, or **Description**. A small arrow appears next to the column heading being sorted and denotes whether the current sort order is by ascending (up arrow) or descending (down arrow) values.

**See also**

- Use ACM Tools on page 69
- Import Export Manager on page 69
- Log Debug Information on page 85

**ACM Default Settings**

How do I open ACM Default Settings?

- From the **Tools** menu, select **Settings**.

Use **ACM Default Settings** to specify the location of the system file, the language shown in the display, how Control Logix content is generated, and whether or not to show indications.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Documentation Path</td>
<td>The location of the template files. By default these are located at Users\Public\Documents\Studio 5000\Template\Application Code Manager.</td>
</tr>
<tr>
<td>Language</td>
<td>The language used by Application Code Manager. The default value uses the operating system language.</td>
</tr>
</tbody>
</table>
### Setting Description

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto-expand Controller Preview</td>
<td>Determines whether the tree in <strong>Controller Preview</strong> is automatically expanded. The default value is selected.</td>
</tr>
<tr>
<td>Control Logix Generation Defaults</td>
<td>Determines what is selected to include in Logix code generated by ACM and whether Application Code Manager should overwrite existing output with new code.</td>
</tr>
<tr>
<td>Overwrite Existing Output</td>
<td>The default setting for <strong>Overwrite Existing</strong> in the Logix Code Generation window.</td>
</tr>
<tr>
<td>ACM Project Data</td>
<td>The default setting for <strong>ACM Project Data</strong> in the Logix Code Generation window.</td>
</tr>
<tr>
<td>Create ACD</td>
<td>The default setting for <strong>Create ACD</strong> in the Logix Code Generation window.</td>
</tr>
<tr>
<td>Indications</td>
<td>Determines whether indications are shown.</td>
</tr>
<tr>
<td>Modified Objects</td>
<td>Sets whether an indication is shown when a library object has been modified. The default value is cleared.</td>
</tr>
<tr>
<td>Newer Library Version Available</td>
<td>Sets whether an indication is shown when a node or folder contains one or more libraries that has a newer version available in <strong>Registered Libraries</strong>. The default value is cleared.</td>
</tr>
</tbody>
</table>

**See also**

- Use ACM Tools on page 69
- Generate a controller file on page 54
- Create a new schedule template on page 76
- Indications on page 45

### Add ACM library content to an existing ACD project

Add content from the ACM library to an existing ACD project by opening the ACD in Application Code Manager.

**To add ACM library content to an ACD project**

1. Click **Tools** and click **Open Target ACD**. The **Open** dialog will display.
2. Select the desired ACD file and click **Open**.
3. The **Target ACD File** tab will replace the **Controller Preview** pane.
4. Drag and drop content from the registered libraries or an existing ACM project to the **Target ACD File** tab. The **Target ACD Generation Wizard - Object Configuration** page appears.
5. Resolve any values in the **Unresolved Names** page, then click **Next**. The **Target ACD Generation Wizard - Merge Actions** page appears.
6. Review the list of merge actions.
• If any merge actions are incorrect, click **Back** to update the parameters as needed.

• If the merge actions are correct, click **Next**.

  The **Target ACD Generation Wizard - L5X Generation Successful** page appears.

7. Click **Finish**. The **Save As** dialog box opens. In **File name** either:

   • Type a new name for the ACD file that contains the merged content.
   • Leave the original name to overwrite the existing ACD project (default).

8. Click **Save** to save the target ACD file.

**See also**

  [Update a project library on page 43](#)
## Reports

Application Code Manager includes a variety of reports to help you track the usage of code objects in your projects.

This table lists the reports available:

<table>
<thead>
<tr>
<th>Report</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project History</td>
<td>For the current project, shows the major revision, minor revision, the user account that made the revision, the date and time that the revision occurred, the comment associated with the revision, and the status of the project at the revision point.</td>
</tr>
<tr>
<td>All Library Usage (Current Project)</td>
<td>For the current project, shows the name of the solution, the type of library, the category of the library object, the catalog number of the library object, the major revision of the library, the minor revision of the library, the number of usages of the library object in the project, and the status of the library.</td>
</tr>
<tr>
<td>Library Usage per Solution (Current Project)</td>
<td>For the selected library in the current project, shows the type of library, the category of the library object, the catalog number of the library object, the major revision of the library, the minor revision of the library, the number of usages of the library object in the solution, and the status of the library.</td>
</tr>
<tr>
<td>Library Usage per Library Type (Current Project)</td>
<td>For the selected library type in the current library of the current project, shows the category of the library object, the catalog number of the library object, the major revision of the library, the minor revision of the library, the number of usages of the library object in the solution, and the status of the library.</td>
</tr>
<tr>
<td>Library Usage per Category (Current Project)</td>
<td>For the selected category in the selected library type of the current library associated with the current project, shows the catalog number of the library object, the major revision of the library, the minor revision of the library, the number of usages of the library object in the solution, and the status of the library.</td>
</tr>
<tr>
<td>Library Usage for Selected Library (Current Project)</td>
<td>For the selected library object within a selected category in the selected library type of the current library associated with the current project, shows the total number of times the library object is referenced and the number of times the object is used by each controller in the current project.</td>
</tr>
<tr>
<td>Chassis Layout</td>
<td>For the chassis of the selected controller object, shows the slot assignments, module names, module type, and module information.</td>
</tr>
<tr>
<td>I/O Schedule</td>
<td>For the selected I/O module of the current controller object, shows the I/O points, type, module information, and description.</td>
</tr>
<tr>
<td>Network Layout</td>
<td>For the selected communication module of the current controller object, shows the IP Address assigned to the module the module name, the module type, and connections information.</td>
</tr>
<tr>
<td>Library Usage per Project</td>
<td>For the connected ACM database, shows the solution, the library type, the library category, the catalog number, the major revision, the minor revision and the number of times used in projects.</td>
</tr>
<tr>
<td>Pending Libraries in database</td>
<td>For the connected ACM database, shows the libraries that have not yet been published. Identifying them by solution, library type, category, catalog number, major revision and minor revision.</td>
</tr>
<tr>
<td>Library Usage per Solution in database</td>
<td>For the selected solution in the connected ACM database, shows the library type, the library category, the catalog number, the major revision, the minor revision and the number of times used in projects.</td>
</tr>
</tbody>
</table>
Chapter 7  Reports

Report Description
Pending Libraries per Solution in database
- For the selected solution in the connected ACM database, shows the libraries that have not yet been published. Identifying them by library type, category, catalog number, major revision and minor revision.

Library Usage per Library Type in database
- For the selected library type of a solution in the connected ACM database, shows the library category, the catalog number, the major revision, the minor revision and the number of times used in projects.

Pending Libraries per Library Type in database
- For the selected library type of a solution in the connected ACM database, shows the libraries that have not yet been published. Identifying them by category, catalog number, major revision and minor revision.

Library Usage for Selected Library in database
- For the selected library object within a library type, shows the total number of times the object is used and the usages per project.

See also
- Report command reference on page 90
- Generate a report on page 91
- View registered library usage on page 50

Report command reference
Reports are generated by selecting the report command from the View context menu. Different reports are available depending on the object selected.

Use this table to locate each report. Items enclosed in brackets [] are replaced by the name of the item in your ACM database.

<table>
<thead>
<tr>
<th>Report</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project History</td>
<td>System View &gt; [Project] &gt; View &gt; Project History</td>
</tr>
<tr>
<td>Project Library Usage</td>
<td>System View &gt; [Project] &gt; Used Libraries &gt; View &gt; Project Library Usage Count</td>
</tr>
<tr>
<td>Project Library Usage per Solution</td>
<td>System View &gt; [Project] &gt; Used Libraries &gt; [Library folder] &gt; View &gt; Project Library Usage per Solution</td>
</tr>
<tr>
<td>Project Library Usage per Solution, Library Type</td>
<td>System View &gt; [Project] &gt; Used Libraries &gt; [Library folder] &gt; [Library type folder] &gt; View &gt; Project Library Usage per Library Type</td>
</tr>
<tr>
<td>Project Library Usage per Solution, Library Type, Category</td>
<td>System View &gt; [Project] &gt; Used Libraries &gt; [Library folder] &gt; [Library type folder] &gt; [Library category folder] &gt; View &gt; Project Library Usage per Category</td>
</tr>
<tr>
<td>Project Library Usage per Library</td>
<td>System View &gt; [Project] &gt; Used Libraries &gt; [Library folder] &gt; [Library type folder] &gt; [Library category folder] &gt; [Library object] &gt; View &gt; Project Library Usage per Library</td>
</tr>
<tr>
<td>Chassis Layout</td>
<td>Controller Preview &gt; [Controller object] &gt; I/O Configuration &gt; View &gt; Chassis Layout</td>
</tr>
<tr>
<td>I/O Schedule</td>
<td>Controller Preview &gt; [Controller object] &gt; I/O Configuration &gt; Backplane &gt; [I/O Module] &gt; View &gt; Module I/O Schedule for Rack Module</td>
</tr>
</tbody>
</table>
### Reports

Reports are generated from the View context menu. Different reports are available depending on the object selected.

#### Generate a report

Reports are generated from the View context menu. Different reports are available depending on the object selected.

To generate a report

1. Select the object in the tree that you want to report on. Reports are available from objects in the System View, Controller Preview, Class View, and Registered Libraries panes.

2. Right-click the object, select View and then choose the report to run.

3. The report appears in a new window.

   - If there are multiple pages in the report, use the navigation controls to move forwards and backwards through the report information.
   - To rerun the report to incorporate changed information, click Refresh.
   - To send the report to a printer, click Print.

The Print dialog box opens. Confirm the printer in Select Printer is correct. Optionally, specify the Page Range and Number of copies.
to print. By default one copy of all pages in the report are printed. Click **Print** to print the report.

- To save the report to a file, click **Export** and then choose **Excel**, **PDF**, or **Word**.

  The **Save As** dialog box opens. In **File name**, type a name for the report being exported. **Save as type** is already selected for the appropriate file format (.pdf, .xlsx, or .docx). Click **Save** to save the report.

**See also**

- **Reports** on page 89
- **View registered library usage** on page 50
- **Report command reference** on page 90
**ACM Console**

The ACM Console is a command-line interface for Application Code Manager that supports scripting and is used to quickly perform operations in the ACM database.

This table lists the commands available in the ACM Console. For more information, such as parameters, data types, and usage information for a command, type `help <nameofcommand>`.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>begincreate</code></td>
<td>Required to start a set of CREATE commands.</td>
</tr>
<tr>
<td><code>createcontroller</code></td>
<td>Creates a controller in a project using a controller definition from an Extended Script (.xml) file. Must be preceded by a <code>begincreate</code> command and followed by an <code>endcreate</code> command to commit to the Application Code Manager database. Only one controller may be defined in the Extended Script file.</td>
</tr>
<tr>
<td><code>createobjects</code></td>
<td>Creates objects in a controller using a controller and object definitions from an Extended Script (.xml) file. Must be preceded by a <code>begincreate</code> command and followed by an <code>endcreate</code> command to commit to the Application Code Manager database. Only one controller may be defined in the Extended Script (.xml) file.</td>
</tr>
<tr>
<td><code>createproject</code></td>
<td>Creates a new project using a project definition from an Extended Script (.xml) file. Must be preceded by a <code>begincreate</code> command and followed by an <code>endcreate</code> command to commit to the Application Code Manager database. Only one project may be defined in the Extended Script (.xml) file.</td>
</tr>
<tr>
<td><code>deletecontroller</code></td>
<td>Deletes a controller from a project.</td>
</tr>
<tr>
<td><code>deleteproject</code></td>
<td>Deletes a project from the Application Code Manager database.</td>
</tr>
<tr>
<td><code>editparameters</code></td>
<td>Edits the values of an object instance’s parameters.</td>
</tr>
<tr>
<td><code>endcreate</code></td>
<td>Required to end a set of CREATE commands.</td>
</tr>
<tr>
<td><code>exportallprojects</code></td>
<td>Exports all ACM projects in the database to Excel files.</td>
</tr>
<tr>
<td><code>exportlibrariesbyattribute</code></td>
<td>Exports libraries filtered by attribute to HSL4 files.</td>
</tr>
<tr>
<td><code>exportlibrariesbyproject</code></td>
<td>Exports libraries used in a project to HSL4 files.</td>
</tr>
<tr>
<td><code>exportlibrariesbyquery</code></td>
<td>Exports libraries filtered by query to HSL4 files.</td>
</tr>
<tr>
<td><code>exportpartial</code></td>
<td>Exports part of an ACM project to an Excel file.</td>
</tr>
<tr>
<td><code>exportproject</code></td>
<td>Exports an ACM project to an Excel file.</td>
</tr>
<tr>
<td><code>generatecontroller</code></td>
<td>Generates the specified controller as an L5X or ACD file.</td>
</tr>
<tr>
<td><code>generatepartial</code></td>
<td>Generates the specified program or routine as an L5X file.</td>
</tr>
<tr>
<td><code>help</code></td>
<td>Displays command help.</td>
</tr>
<tr>
<td><code>importproject</code></td>
<td>Imports a project into an ACM database from an Excel file.</td>
</tr>
<tr>
<td><code>publishlibrary</code></td>
<td>Extracts and publishes a library from an ACD.</td>
</tr>
<tr>
<td><code>registerlibrary</code></td>
<td>Registers the library into the ACM database.</td>
</tr>
<tr>
<td><code>run</code></td>
<td>Runs a console script.</td>
</tr>
<tr>
<td><code>switchdatabase</code></td>
<td>Switches to a different ACM database at any point within a script.</td>
</tr>
</tbody>
</table>
The ACM Console is a separate application from Application Code Manager.

**To open the ACM Console**

1. Minimize or close the Application Code Manager if it is open.
2. On the desktop, double-click the ACM Console icon or click Start > Rockwell Software > ACM Console.

The ACM Console appears.

**List all commands**

The help function in the ACM console can provide a list of all the ACM commands.

**To list all commands**

1. Open the ACM Console.
2. At the $ prompt, type `help` and then press Enter.

A list of all commands is displayed.

**Generate a limited list of commands**

Use the help function in the ACM Console to provide a limited list of commands.
To generate a limited list of commands

1. Open the ACM Console.

2. At the $ prompt, type `help` and the first letters of the command that to limit the list to and then press the Tab key.

Example: typing "help g" and then pressing the Tab key results in the prompt automatically completing the "g" to generate, pressing tab again returns a comma delimited list of `generatecontroller` and `generatepartial`.

See also

Generate detailed command information on page 95

ACM Console on page 93

Open the ACM Console on page 94

Generate detailed command information

Use the help function in the ACM Console to provide detailed information about a specific command.

To generate detailed command information

1. Open the ACM Console.

2. Type `help` then the full command and then press Enter.

Example: type `help exportproject`, then press Enter.

Detailed information on the `exportproject` command displays.

See also

ACM Console on page 93

Open the ACM Console on page 94

Console scripts

A console script is a text file (.txt) containing a set of valid Application Code Manager Console commands. The commands are run in order, top to bottom.

If using the `CREATEPROJECT`, `CREATECONTROLLER`, or `CREATEOBJECTS` commands, the console script requires an Extended script (.xml) file. The Extended script is called as part of the command to provide data to Application Code Manager about the project, controller, or object that is created.
See also

Create an Application Code Manager Console script on page 96
Run an Application Code Manager Console script on page 96
Extended scripts on page 97

Create an Application Code Manager Console script

Create an Application Code Manager Console script to store a set of commands for later or repeated use.

To create an Application Code Manager Console script

1. Open a text editor, such as Notepad.
2. Enter one or more valid Application Code Manager Console commands into the script.
   
   **Important:** Only enter one Application Code Manager Console command per line of the script.

3. Save the script as a plain text (.txt) file.

See also

Run an Application Code Manager Console script on page 96
ACM Console on page 93

Run an Application Code Manager Console script

Run an Application Code Manager Console script to execute a set of predefined commands on the Application Code Manager database.

To run an Application Code Manager Console script

1. Double-click the ACM Console shortcut on the desktop.
2. Select File > Run Script.
3. In Select an ACM Script File locate the file, then click Open.

The script executes.

Alternatively:
At the Application Code Manager Console prompt, type `run filelocationpath\scriptfilename.txt`, then press Enter. Replace `filelocationpath` with the file path, such as `C:\Users\<username>\Desktop\` and `scriptfilename.txt` with the script filename.

See also

ACM Console on page 93

Extended scripts

An Extended script is an XML (.xml) file that contains information about a project, controller, or library object. The Extended script is called from a CREATE command in an Application Code Manager Console script, which passes the object data contained in the Extended script to the Application Code Manager database.

Projects and controllers may not be defined in the same XML file. For best results, create three Extended scripts: one for the project, one for the controller, and one for all the associated objects.

Tip: When creating a Project or Controller XML file, set up a reference (@Project or @Controller) to use the project name passed in from the Application Code Manager Console script. This allows a single Project or Controller XML file to be used for multiple projects.

An example of an Application Code Manager Console script that calls an Extended script is:

```
BEGINCREATE
  CREATEPROJECT "myProject" "C:\Script Files\Project.xml"
  CREATECONTROLLER "myProject" "myController" "C:\Script Files\Controller.xml"
  CREATEOBJECTS "myProject" "myController" "C:\Script Files\Objects.xml"
ENDCREATE
```

See also

Generate an example Extended script on page 97

Generate an example Extended script

Generate an example Extended script to copy and paste the XML structures and parameters to a new Extended script file for editing.

To generate an example Extended script

1. Create a Project, Controller, and Object in Application Code Manager.
2. Generate an L5X controller file with the **ACM Project Data** option enabled.

3. Open the .L5X controller file in a text editor.

4. Copy the `<IObs>` and `<ICOObjs>` nodes to a new Extended script file.

**See also**

[Generate a controller file](#) on page 54

### Library `<IObj>`

The Library `<IObj>` node represents a library instance configuration.

**Parent node:**

- `<IObs>`

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the library object instance.</td>
</tr>
<tr>
<td>Task</td>
<td>The task name in which the instantiated library content will reside if the library is a task type library. If applicable, Project and Controller objects may be left empty.</td>
</tr>
<tr>
<td>Program</td>
<td>The program name in which the instantiated library content will reside if the library is a program type library. If applicable, Project and Controller objects may be left empty.</td>
</tr>
<tr>
<td>Description</td>
<td>The description message of the library instance.</td>
</tr>
</tbody>
</table>
| Guid               | The unique ID of the library in the ACM database. During instantiation, if a Guid is supplied, the database is queried to find a match.  
  - If the Guid is supplied then the Sol, Catalog Number, Maj and Min attributes are ignored.  
  - If the Guid is not supplied then Sol and Catalog Number must be supplied. |
| Sol                | The solution category defined within the library. This is a mandatory field if Guid is not supplied. |
| Catalog Number     | The catalog number as defined within the library. This is a mandatory field if Guid is not supplied. |
| Maj                | The major revision number of the library. This is an optional field. If omitted and Guid is not supplied then the highest major library revision number will be selected. |
| Min                | The minor revision number of the library. This is an optional field. It can only be included if the Maj attribute is present. If it is excluded then the highest minimum library revision number will be selected. |

### Parameter `<IPar>`

The Parameter `<IPar>` node represents a library parameter.

**Parent node:**
- `<IPars>`

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the parameter. This is a mandatory field and must be supplied.</td>
</tr>
<tr>
<td>Value</td>
<td>The value assigned to the parameter. This is a mandatory field and must be supplied.</td>
</tr>
<tr>
<td>RefVal</td>
<td>This attribute only applies to reference type parameters. If supplied, this value adds the reference to the parameter value, such as <code>#refvalue</code>. If not supplied, then a reference value is not added to the parameters value.</td>
</tr>
</tbody>
</table>

**SubObject `<ISObj>`**

The SubObject `<ISObj>` node represents a Sub Object. SubObject parameters must be supplied as `<IPar>` nodes within the child `<IPars>` node.

**Parent node:**

- `<ISObjs>`

**Child node(s):**

- `<IPars>`

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the SubObject, such as <code>Inp_intlk01</code>.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of SubObject, such as <code>Interlock</code>.</td>
</tr>
<tr>
<td>Description</td>
<td>The description of the SubObject.</td>
</tr>
<tr>
<td>OverrideDesc</td>
<td>If formatting is applied to the SubObject description field in the library and this field is set to true, then the description can be overridden.</td>
</tr>
</tbody>
</table>

**Linked Library `<ILLib>`**

The Linked Library `<ILLib>` node represents a linked library reference.

**Parent node:**

- `<ILLibs>`

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name displayed for the Linked Library.</td>
</tr>
</tbody>
</table>
Interface Links <IILink>

The Interface Links <IILink> node represents an interface link for a library instance. An Interface Link requires one <IMems> child nodes and one or more <IMem> grandchild nodes.

**Parent node:**

- <IObjs>

**Child node(s):**

- <IMems>

### Attribute Description

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>The instance name of the target library.</td>
</tr>
</tbody>
</table>

#### Interface Links <IILink>

- Keying
  - **ExactMatch**: ACM matches input members against their respective **Key Id** and **Revision** field values.
  - **Disabled**: ACM does not match input members against their respective **Key Id** and **Revision** field values.

- RefInt
  - A reference to the library instance to which the Interface Link points.

See also

- **Interface Members <IMem>** on page 100

### Interface Members <IMem>

The Interface Member <IMem> node represents the configuration of an Interface Link member.

**Parent node:**

- <IMems>

### Attribute Description

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the Interface Member.</td>
</tr>
<tr>
<td>Value</td>
<td>The library instance name and output interface member to which to connect.</td>
</tr>
<tr>
<td>RefVal</td>
<td>The name of the output interface member in the destination library.</td>
</tr>
</tbody>
</table>
See also

Interface Links <IILink> on page 100

**Controller Object <ICOObj>**

The Controller Object <ICOObj> node represents an object that appears in the Application Code Manager Controller Preview pane.

Controller Objects are only required when additional object configuration is required for a controller object, such as Instance Execution Ordering.

**Parent node:**

- <ICOObjs>

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name to be assigned to the controller object.</td>
</tr>
<tr>
<td>Obj</td>
<td>The instantiated name of the library object.</td>
</tr>
<tr>
<td>COParent</td>
<td>The name of the parent within which the object resides.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of object, such as PROGRAM, ROUTINE, or AOI.</td>
</tr>
<tr>
<td>NameSub</td>
<td>The substitution name, if applied.</td>
</tr>
<tr>
<td>Order</td>
<td>Specifies where a routine would be displayed under its parent program node within the ACM Controller Preview panel. For example, a routine with an Order value of 0 is displayed at the top of the tree. A routine with an Order value of 1 is displayed next in the list.</td>
</tr>
<tr>
<td>MarkDel</td>
<td>Excludes an AOI for a specific controller.</td>
</tr>
<tr>
<td>MainRtn</td>
<td>Specifies the level of the routine. 2: Sets the routine as the active routine. 1: Sets the routine as an inactive routine. 0: Sets the routine as a normal routine.</td>
</tr>
<tr>
<td>InstExecOrder</td>
<td>Specified for ROUTINE types, otherwise the value is blank. The value specifies the routine execution order number. This value will determine in what sequence routines are processed when generating a controller from within ACM.</td>
</tr>
</tbody>
</table>
Legal Notices

Copyright notice

Copyright © 2020 Rockwell Automation Technologies, Inc. All Rights Reserved. Printed in USA.

This document and any accompanying Rockwell Software products are copyrighted by Rockwell Automation Technologies, Inc. Any reproduction and/or distribution without prior written consent from Rockwell Automation Technologies, Inc. is strictly prohibited. Please refer to the license agreement for details.

End User License Agreement (EULA)

You can view the Rockwell Automation End-User License Agreement ("EULA") by opening the License.rtf file located in your product's install folder on your hard drive.

The software included in this product contains copyrighted software that is licensed under one or more open source licenses. Copies of those licenses are included with the software. Corresponding Source code for open source packages included in this product are located at their respective web site(s).

Alternately, obtain complete Corresponding Source code by contacting Rockwell Automation via the Contact form on the Rockwell Automation website: http://www.rockwellautomation.com/global/about-us/contact/contact.page Please include "Open Source" as part of the request text.

A full list of all open source software used in this product and their corresponding licenses can be found in the OPENSOURCE folder included with these Release Notes. The default installed location of these licenses is C:\Program Files (x86)\Common Files\Rockwell\Help\<Product>\ReleaseNotes\OPENSOURCE\index.htm.

Trademark Notices

Allen-Bradley, Rockwell Automation, and Rockwell Software are trademarks of Rockwell Automation, Inc.

Any Rockwell Automation software or hardware not mentioned here is also a trademark, registered or otherwise, of Rockwell Automation, Inc.

Other Trademarks

All other trademarks are the property of their respective holders and are hereby acknowledged.
Trademarks not belonging to Rockwell Automation are property of their respective companies

**Warranty**

This product is warranted in accordance with the product license. The product’s performance may be affected by system configuration, the application being performed, operator control, maintenance, and other related factors. Rockwell Automation is not responsible for these intervening factors. The instructions in this document do not cover all the details or variations in the equipment, procedure, or process described, nor do they provide directions for meeting every possible contingency during installation, operation, or maintenance. This product’s implementation may vary among users.

This document is current as of the time of release of the product; however, the accompanying software may have changed since the release. Rockwell Automation, Inc. reserves the right to change any information contained in this document or the software at any time without prior notice. It is your responsibility to obtain the most current information available from Rockwell when installing or using this product.

**Contact Rockwell Automation**

Customer Support Telephone — 1.888.382.1583

Rockwell Automation support

Use the following resources to access support information.

<table>
<thead>
<tr>
<th>Technical Support Center</th>
<th>Knowledgebase Articles, How-to Videos, FAQs, Chat, User Forums, and Product Notification Updates</th>
<th><a href="https://rockwellautomation.custhelp.com">https://rockwellautomation.custhelp.com</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct Dial Codes</td>
<td>Find the Direct Dial Code for your product. Use the code to route your call directly to a technical support engineer.</td>
<td><a href="http://www.rockwellautomation.com/global/support/direct-dial.page">http://www.rockwellautomation.com/global/support/direct-dial.page</a></td>
</tr>
</tbody>
</table>

Documentation feedback

Your comments will help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete the How Are We Doing? form at http://literature.rockwellautomation.com/idc/groups/literature/documents/du/ra-du002_-en-e.pdf.