MORNINGSTAR Wealth

Custodial Integrator

Installation Guide

(APX)

©2025 Morningstar. All Rights Reserved.

APX is a registered trademark of Advent Software, Inc.

Custodial Integrator Product Version:	V3.19
Document Version:	44
Document Issue Date:	April 10, 2025

Technical Support:	(866) 856-4951
Web:	byallaccounts.morningstar.com

Table of Contents

ABOUT THE MANUAL 4
AUDIENCE AND FORMAT 4
RELATED DOCUMENTS 4
INSTALL REQUIREMENTS 4
SYSTEM OVERVIEW 4
SYSTEM REQUIREMENTS 5
CUSTODIAL INTEGRATOR COMPUTER 5
DATABASE COMPUTER 5
APX 5
REQUIRED SOFTWARE 6
ADDITIONAL CONSIDERATIONS 7
INSTALL CUSTODIAL INTEGRATOR 7
OVERVIEW 7
SOFTWARE DISTRIBUTION 8
CUSTODIAL INTEGRATOR 8
PREPARE TO INSTALL 8
INSTALL 10
DATABASE COMPUTER INSTALL – SQL SERVER 10
CUSTODIAL INTEGRATOR COMPUTER INSTALL OR UPGRADE 19
CONFIGURE CUSTODIAL INTEGRATOR 20
CONSIDER PASSWORD ENCRYPTION MODELS 23
INSTALL MULTIPLE INSTANCES THAT ACCESS THE SAME DATABASE 23
SETUP FOR MULTIPLE CUSTODIAL INTEGRATORS THAT ACCESS MODE THE SAME DATABASE 23
DEFAULT ENCRYPTION MODEL 23
ENHANCED ENCRYPTION MODEL 23
INSTALL MULTIPLE INSTANCES OF CUSTODIAL INTEGRATOR ON SAME MACHINE WITH DIFFERENT DATABASES 26
TROUBLESHOOT INSTALL PROBLEMS 28
GENERAL TECHNIQUES 28
KNOWN ISSUES 28
CUSTODIAL INTEGRATOR CONFIGURATION REFERENCE 30
CUSTODIAL INTEGRATOR PARAMETERS 30

CI Customizations....31
APX Customizations....34
Corporate Firewall....39
Database....39
Debugging....40

About the Manual

Audience and Format

This manual describes the system requirements for the Custodial Integrator ("CI") product, the procedure for installing Custodial Integrator, and how to perform initial configuration of Custodial Integrator. The audience for this manual is an Information Technology professional who is responsible for installing and maintaining software systems. You should be familiar with basic system administration procedures, manipulation of Windows services, and working directly in the Windows file system.

Related Documents

The following related documents are available from ByAllAccounts:

- The Custodial Integrator Solution: provides an overview of the Custodial Integrator solution.
- Custodial Integrator Validation FAQ: provides an overview of the account validation process.

Install Requirements

System Overview

The Custodial Integrator architecture consists of the following main components:

- Custodial Integrator Database: a Microsoft SQL Server database
- Custodial Integrator Installation Folder: a Windows folder hierarchy where permanent files are stored
- Custodial Integrator Working folder: a Windows folder hierarchy where temporary files are stored
- Custodial Integrator Application: the Java application that brings data from WebPortfolio service and prepares it for PortfolioCenter. Custodial Integrator **is a "fat client" that presents a User Interface and interacts with the** Custodial Integrator **Database**, Custodial Integrator Folder, and the WebPortfolio web service. The application requires Oracle Java Runtime Environment (JRE) software.

Custodial Integrator interacts with the following other components:

- Your Advent APX installation
- The WebPortfolio Service web site at URL https://www.byallaccounts.net/

Custodial Integrator requires the following third-party software:

- Oracle Java Runtime Environment (JRE)
- Microsoft SQL Server

See Required Software, page 6 for version requirements.

The remainder of this manual refers to the following:

- APX Computer: the computer where your APX software is installed.
- Custodial Integrator Computer: the computer on which the Custodial Integrator application is installed and run. You will install the Java Runtime and Custodial Integrator on this computer.

– Database Computer: the computer where Microsoft SQL Server is installed. The Custodial Integrator database will be created on this computer.

Note that you may choose to install Custodial Integrator and the database software on the same computer.

System Requirements

This section describes the base system requirements for the Custodial Integrator and third-party software installation.

Custodial Integrator Computer

The following table defines the base system requirements for the Custodial Integrator Computer.

Component	Requirement	Notes
CPU	Intel Pentium or compatible 166-megahertz (MHz) or higher	
Disk space	15 MB	Additional space based on your download volume is needed to hold data downloads.
RAM	As recommended for your operating system	
Operating System	Windows 10 Windows Server 2008 Windows Server 2012 Windows Server 2012 R2 Windows Server 2016	Note: Running Custodial Integrator via Windows Terminal Services is supported.
Internet connectivity	Minimum 56k bps	
Internet Browser	Internet Explorer version 9.0 or higher	Browser is required only for the installation of the Custodial Integrator application.

Database Computer

The database computer must comply with the minimum system requirements for Microsoft SQL Server edition that you are using. The supported versions of Microsoft SQL Server versions are indicated in Required Software, page 6. Requirements differ for the full version of Microsoft SQL Server versus Express. Please refer to Microsoft's minimum system requirements description for the Microsoft SQL Server system you plan to use with Custodial Integrator to ensure you have the required CPU, disk space, RAM, and operating system.

The disk requirement for the Custodial Integrator Database is 70 MB.

APX

This version of Custodial Integrator supports APX 1.0 or later. Custodial Integrator makes use of your APX installation in the following ways:

1. Custodial Integrator uses the APX Import/Export utility (apxix.exe) to export Security (sec.inf) and Security Type (type.inf) data from your AOX installation. This information enables Custodial Integrator to produce transactions, positions, and prices that use the security symbols and security types you have defined in APX.
2. Custodial Integrator uses extracts a group of Portfolio codes for use in Custodial Integrator from an APX group that you specify.
3. Custodial Integrator uses the APX Import/Export utility to import:
   - Transactions into a designated Trade Blotter
   - Prices into the APX for a specified date
   - Positions into a Position Blotter
   - Position lots into a Position Lot Blotter if position lots are enabled

Required Software

Custodial Integrator requires that the APX BackOffice Utilities be installed on the same computer where Custodial Integrator will be installed. Custodial Integrator also requires that APX have the DataPort component installed to enable Custodial Integrator to post reconciliation positions to APX.

Custodial Integrator requires the installation of the third-party software shown in the table below.

Software	Target Computer	Vendor	Version	Distribution	Disk Space
Microsoft SQL Server	Database Computer	Microsoft	SQL Server 2016 or later is required, including Express versions.	Available from Microsoft	Depends on the SQL Server version. Refer to Microsoft documentation
Java Runtime Environment (JRE)	Custodial Integrator Computer	Oracle Corporation	1.8 or later	CI distribution: 1.8 update 261 is included in Custodial Integrator kit	239 MB (for 64-bit) 180 MB (for 32-bit
Microsoft OLE DB Driver for SQL Server (MSOLEDBSQL)	Custodial Integrator Computer	Microsoft	18.2 or later	Custodial Integrator distribution: Microsoft OLE DB Driver for SQL Server (MSOLEDBSQL) 18.5.0.0 is included in Custodial Integrator kit	11.1 MB

Additional Considerations

1. APX authentication

   Custodial Integrator invokes the APX import/export utility apxix.exe. This utility authenticates to the APX server in various ways depending on how APX is installed and configured for you. If authentication is not properly configured for your Windows user then this can result in Custodial Integrator being unable to utilize the apxix.exe utility. You should consult with your APX administrator if you have problems with Custodial Integrator when it is attempting to export from or import to APX. The primary symptom will be that Custodial Integrator is waiting for an extended period for APX to respond when in fact APX has refused access due to authentication failure. Custodial Integrator is unable to ascertain that this has occurred without analyzing the Apxix log file. If you are uncertain as to the cause of problems of this nature please contact ByAllAccounts Technical Support for assistance with diagnosis.

2. Custodial Integrator Computer

   The Custodial Integrator Computer should have good network proximity to the APX Computer.

3. Database Computer

   If you already have a supported Microsoft SQL Server installation, you may use that installation to house the Custodial Integrator database. If you do not have a Microsoft SQL Server installation or choose not to use your existing installation, you may install Microsoft SQL Server Express on the Custodial Integrator Computer or on a separate Database Computer.

4. Internet Connectivity

   If you have a corporate firewall, Custodial Integrator must be configured to tunnel through the firewall or to bypass the firewall to access to https://www.byallaccounts.net.

5. Multiple Users

   This installation is designed to support a single Custodial Integrator user and should only be installed on a single computer. If you wish to install for multiple Custodial Integrator users or provide a redundant installation for failover, please contact ByAllAccounts for assistance.

INSTALL CUSTODIAL INTEGRATOR

Overview

The Custodial Integrator installation consists of the following steps, each of which is described in a subsequent section of this chapter:

1. Prepare for the installation by gathering required information and approvals
2. Install Microsoft SQL Server on the Database Computer. This is accomplished using a kit distributed by Microsoft. You may alternatively use a previously installed instance of Microsoft SQL Server.
3. Install the Custodial Integrator Application on the Custodial Integrator Computer. This install includes:
   a. Java Runtime (pre-requisite)
   b. Custodial Integrator product installation

   c. Custodial Integrator database initialization

4. Configure Custodial Integrator
5. Verify Installation

Depending on your configuration, you may need to refer to Install multiple instances that access the same database on page 23 or Install multiple instances of Custodial Integrator on same machine with different databases on page 26.

Software Distribution

Custodial Integrator

The Custodial Integrator software distribution is available on the Internet at the following URL:

https://www.byallaccounts.net/CI/

A login and password is required to access this distribution and is available on request for licensed customers. The distribution contains Custodial Integrator documentation and software kits.

Prepare to Install

To prepare for the Custodial Integrator installation, please complete the following steps:

1. Verify System Requirements for both the Custodial Integrator Computer and the Database Computer as described in the previous section.
2. Existing Java Runtime Environment (JRE) installation
   a. If a version of JRE has been installed on the Custodial Integrator Computer, determine the version.
   b. If the version is 1.8 or later, the installation will leave it as is. If it is not at least version 1.8, the installation will upgrade it to 1.8 update 261. Therefore, you must determine if this Java upgrade is acceptable on the Custodial Integrator computer. If another application on the Custodial Integrator computer depends on your current version of JRE and cannot tolerate a later version, you will have to determine whether to move this other application or to install Custodial Integrator on an alternate computer.
3. Windows Administrative Access
   a. Obtain a Windows Login that is a member of the Administrators Group on the Custodial Integrator Computer for the Custodial Integrator installation.
   b. Obtain a Windows Login that is a member of the Administrators Group on the Database Computer for the SQL Server installation.
4. Microsoft SQL Server (Note: for existing SQL Server installations only)
   a. Request that your SQL Server administrator approve the installation of the Custodial Integrator database.
   b. Obtain the password for the "sa" user or obtain a Windows Login that has "sa" or "dbo" privileges in the existing SQL Server installation that allows Windows authentication.
5. Corporate Firewall

   If you have a corporate firewall that restricts access to the Internet, you may need one or more of the following:

   a. Consultation from your ISA Server administrator as to the appropriate methods for tunneling or bypassing your firewall
   b. Proxy host name
   c. Proxy port for https (default: 443)
   d. Proxy username and password (for Basic authentication only; NTLM authentication is not supported and requires firewall bypass)

6. APX
   a. Obtain the location of your APX installation and information about some of the APX Custom Labels that your APX administrator may have defined (additional details provided later)

Install

This section describes the installation procedure for Custodial Integrator and required third-party products. We recommend that you read through the entire text of the installation procedure before starting the installation.

Database Computer Install – SQL Server

If you plan to use an existing Microsoft SQL Server installation, omit this step.

Note: These steps describe installing Microsoft SQL Server 2012 Express SP4, but you can install another version supported by Custodial Integrator. Refer to Required Software, page 6.

During the installation, there are a few important options that you must select as described here for SQL Server to be configured properly for use with Custodial Integrator.

Downloads

You can download Microsoft® SQL Server® 2012 Service Pack 4 (SP4) Express at this link https://www.microsoft.com/en-us/download/details.aspx?id=56042.

Note: If you are downloading the SQL Server Management Studio stand-alone application to perform a database migration, make sure to use the same version or a newer version than the SQL Server version used on your old server.

On the download pages for SQL Server Express 2012 SP4 or SQL Server Management Studio there are many options to choose from when downloading.

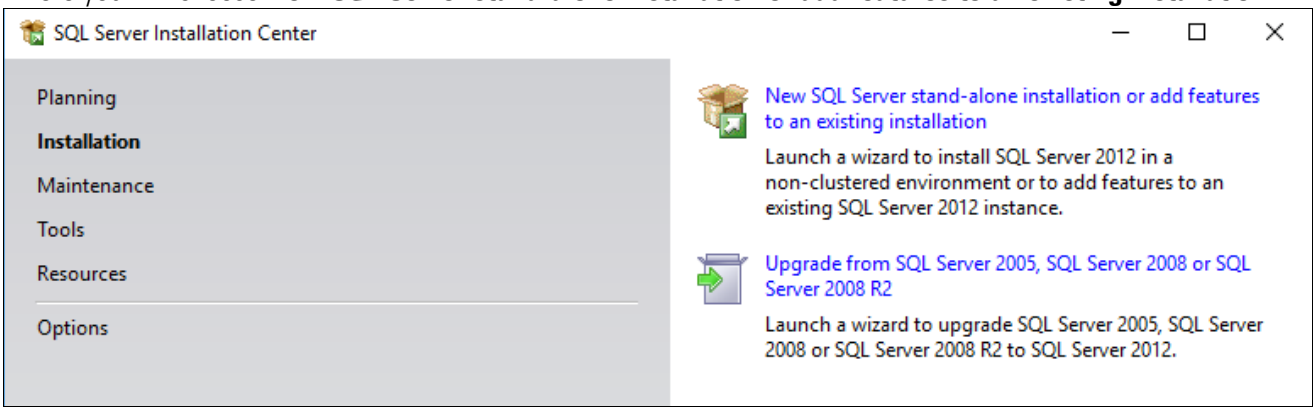
This list includes most of the files listed on the SQL Server download pages with their descriptions:

- **SQLEXPR_x86_ENU.exe**
  The installation files for SQL Server Express that gives the option of running as a 32-bit and 64-bit application.
- **SQLEXPRWT_x86_ENU.exe**
  The installation files for SQL Server Express that gives the option of running as a 32-bit and 64-bit modes. This includes the SQL Server Management Studio Express Application.
- **SQLEXPR_x64_ENU.exe**
  The installation files for SQL Server Express that are for 64-bit systems only.
- **SQLEXPRWT_x64_ENU.exe**
  The installation files for SQL Server Express that are for 64-bit systems only and the application SQL Server Management Studio Express.
- **SQLManagementStudio_x64_ENU.exe**
  The standalone application SQL Server Management Studio for 64-bit systems only.
- **SQLManagementStudio_x86_ENU.exe**
  The standalone application SQL Server Management Studio for 64-bit with the option of 32-bit systems.

Choose the download(s) you need.

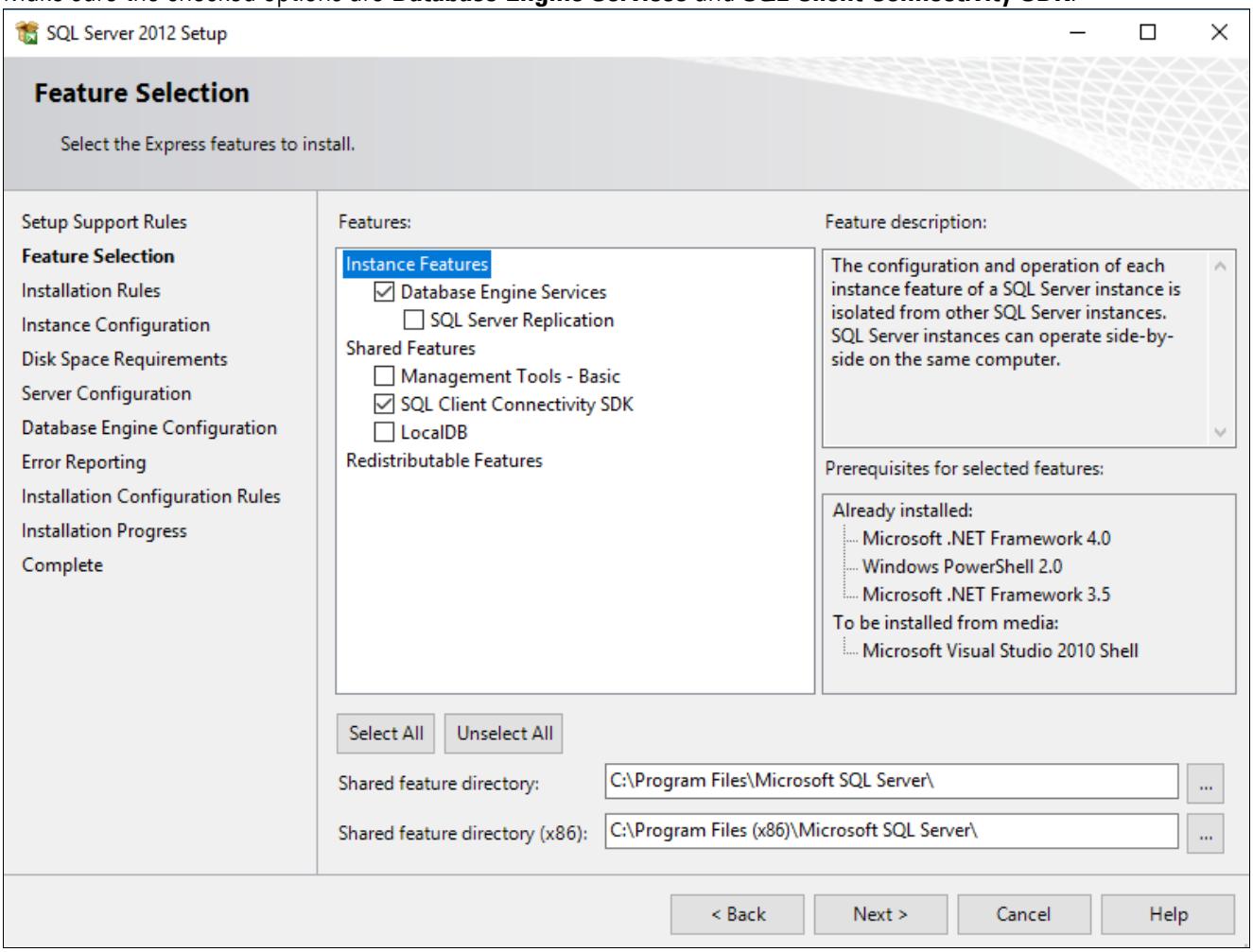
Run SQL Server Install

1. Run the installation file that you downloaded from the Microsoft website. The installation wizard will launch where you will choose **New SQL Server stand-alone installation or add features to an existing installation**.

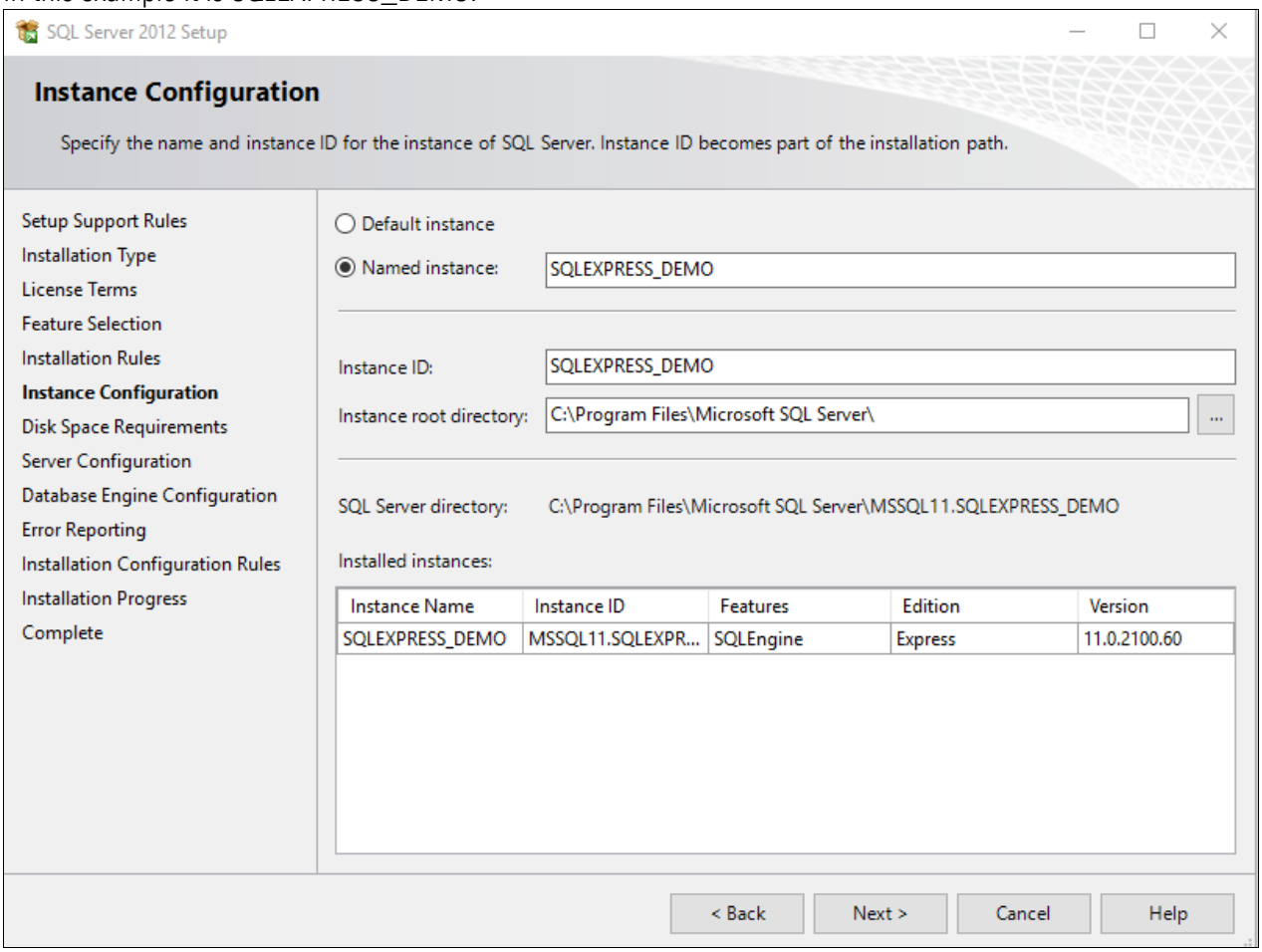


2. Accept the license terms prompt before the install begins. The application will run a system check.
   Note: The system may require a reboot when the scan is complete.

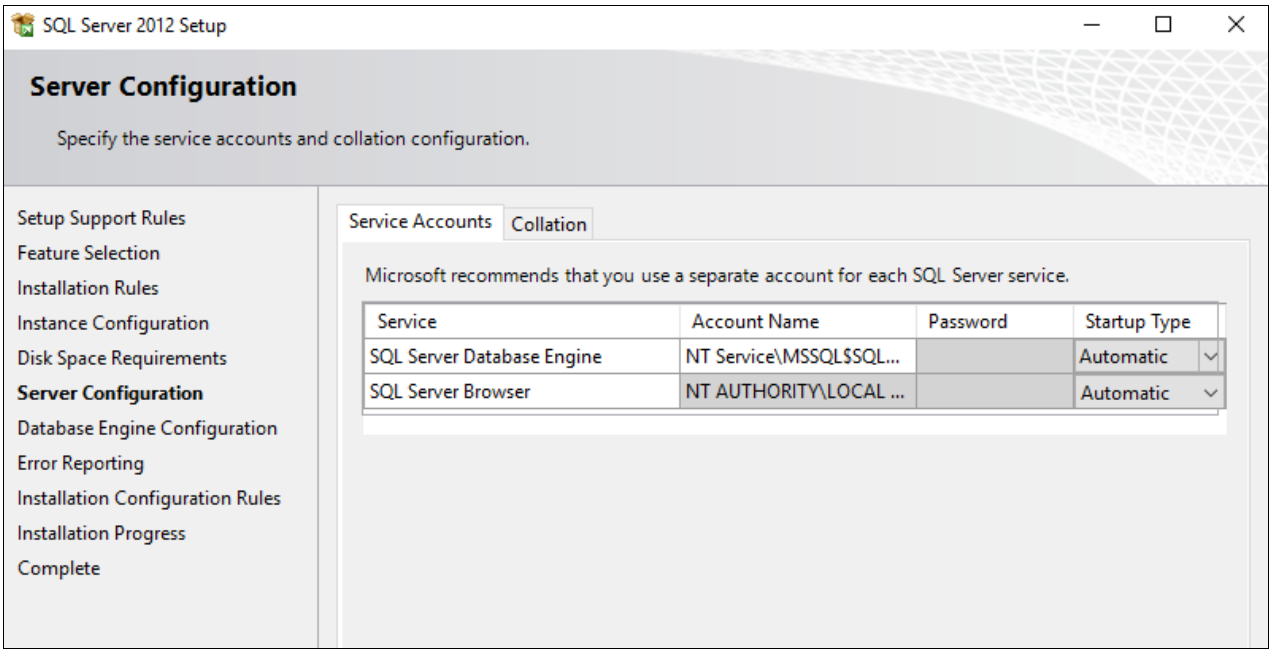
3. Make sure the checked options are **Database Engine Services** and **SQL Client Connectivity SDK**.



4. Check the **Named Instance** radio button and enter the name you wish use to identify your SQL server instance. In this example it is SQLEXPRESS_DEMO.

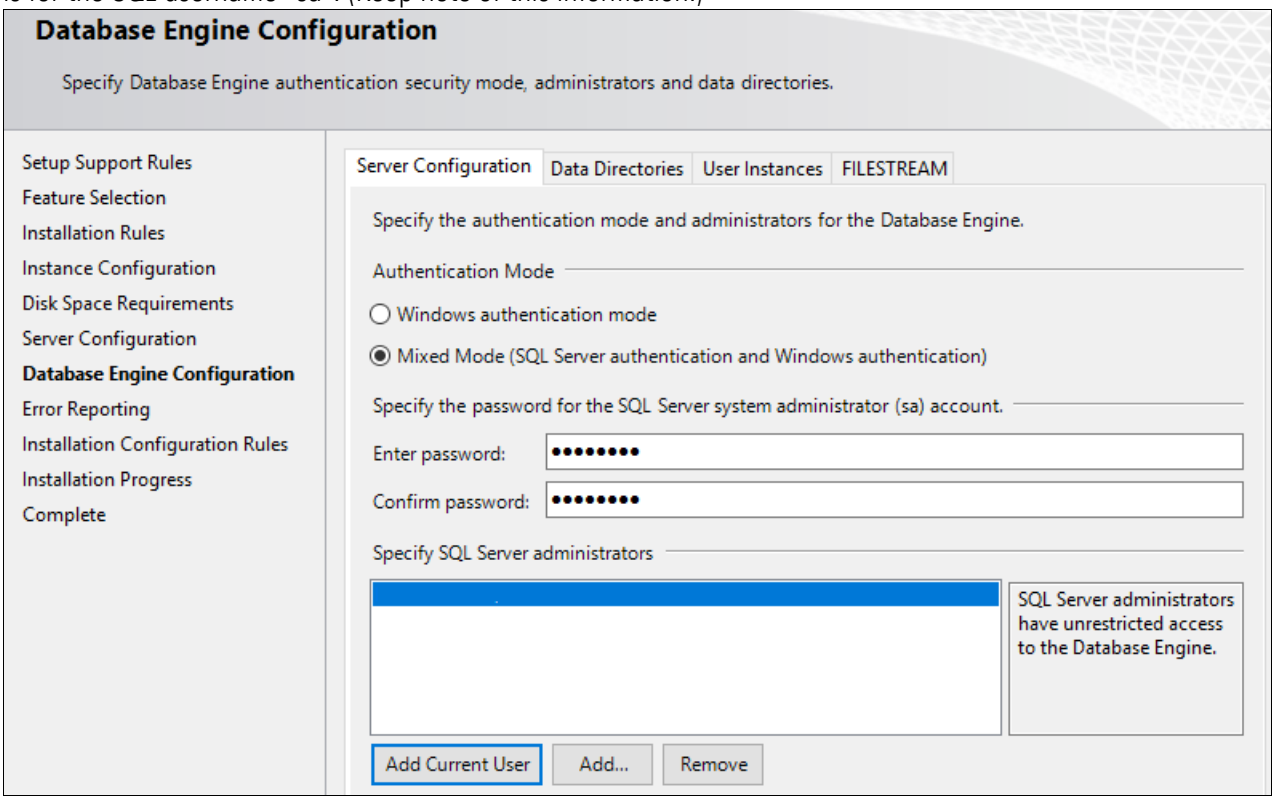


5. Confirm that both of the SQL Services are set to **Automatic** (as shown below) to ensure the services are running after the installation.

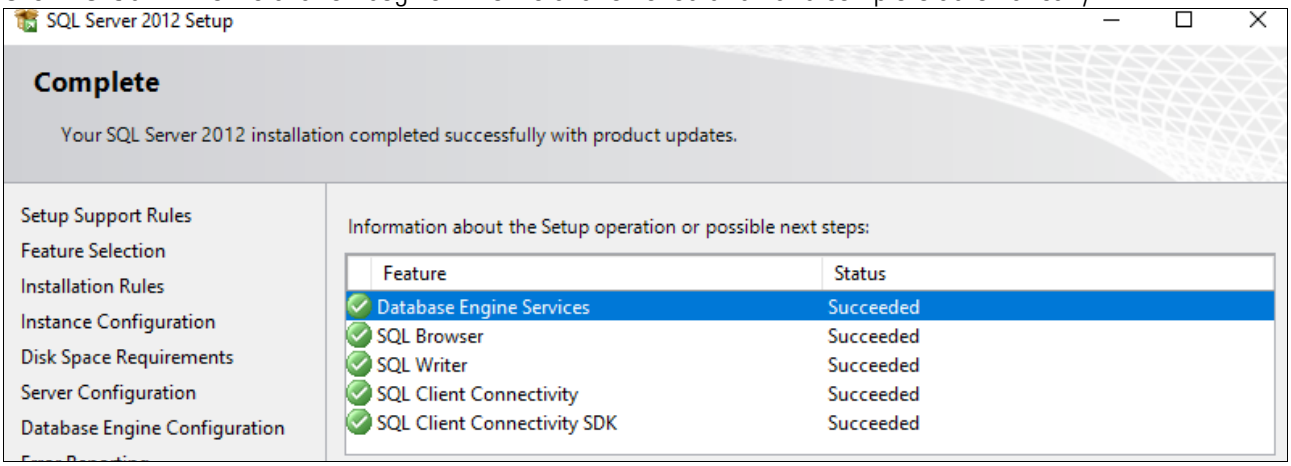


6. Click **Next**.

7. Select the radio button labeled **Mixed Mode**. Enter a password in twice to create the password. This password is for the SQL username "sa". (Keep note of this information.)



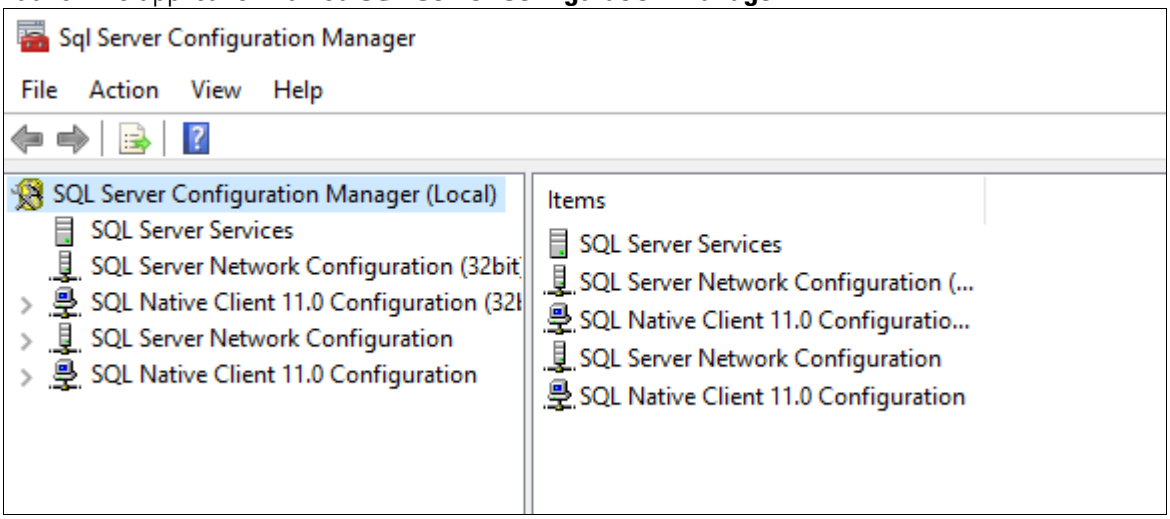
8. Click **Next** until the installation begins. The installation should run and complete automatically.



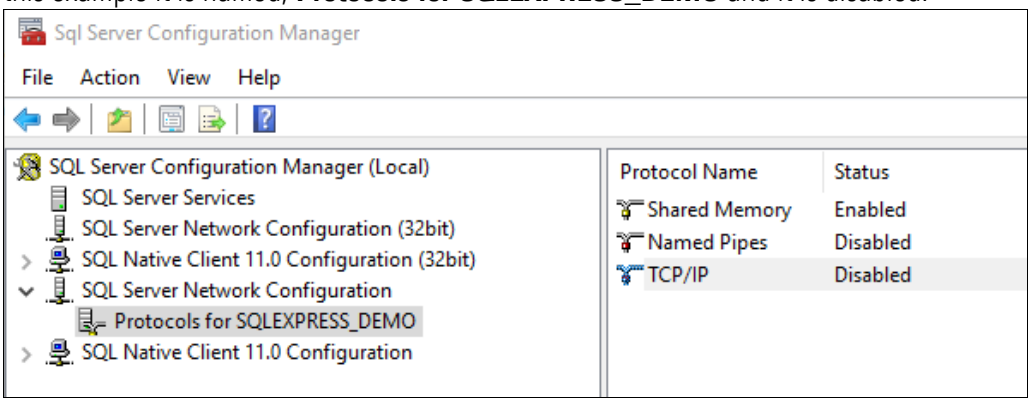
9. Once it is complete, click **Close** then confirm that the TCP/IP ports are enabled by using the following steps.

Confirm TCP/IP Ports are Enabled

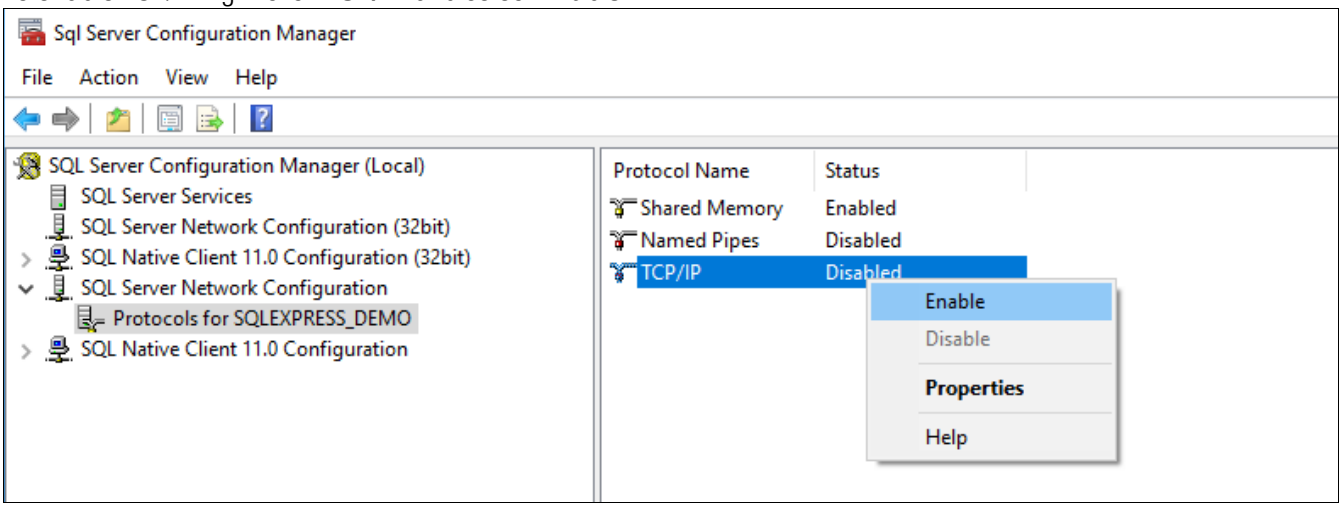
1. Launch the application named **SQL Server Configuration Manager**.



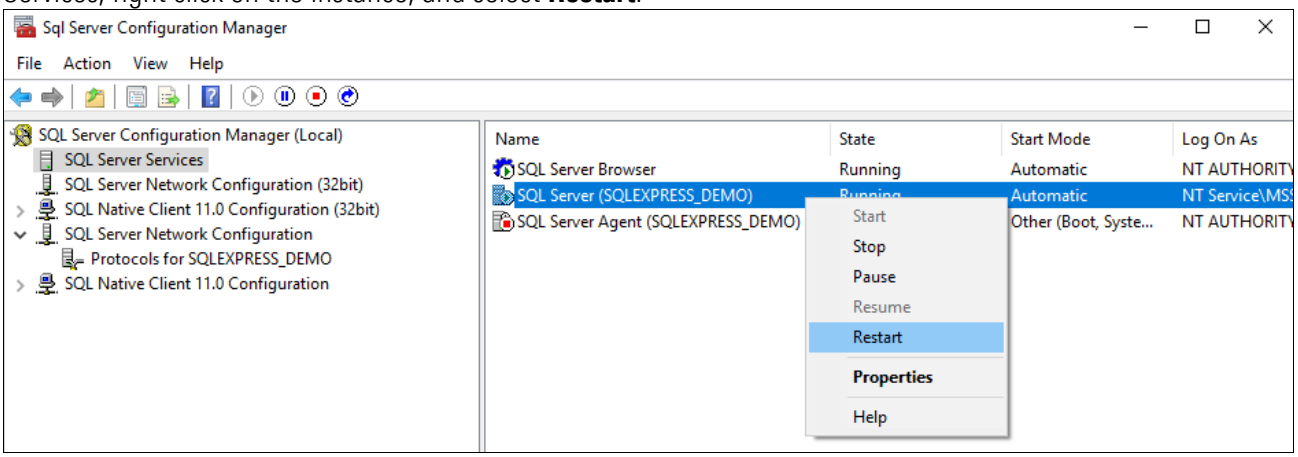
2. Highlight the protocols for the instance name which is located under the **SQL Server** Network **Configuration**. In this example it is named, **Protocols for SQLEXPRESS_DEMO** and it is disabled.



3. To enable TCP/IP right click **TCP/IP** and select **Enable**.



4. If you had to enable the TCP/IP port on your SQL Server service, you must restart it. Highlight the SQL Server Services, right click on the instance, and select **Restart**.



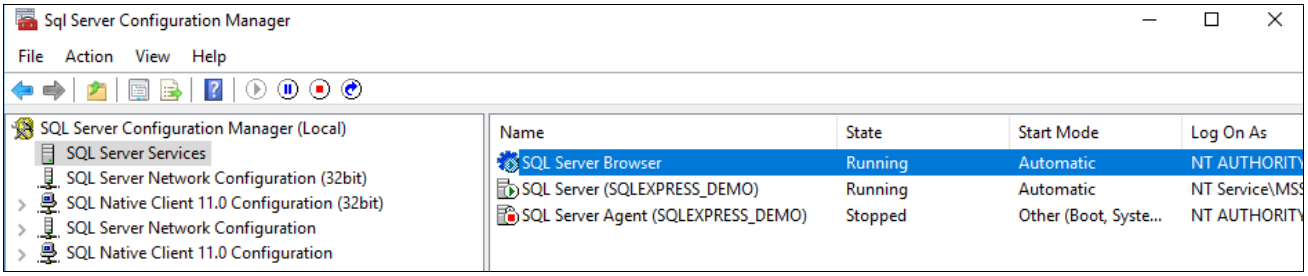
5. After it restarts, configure the SQLBrowser service to run automatically by using the following steps.

Configure SQLBrowser Service of SQL Server to Run Automatically

In addition to being configured to accept remote connections, the SQLBrowser service of SQL Server must also be configured to run automatically if it is not already. As an example, to perform this configuration for SQL Server 2012 SP4, complete the following steps.

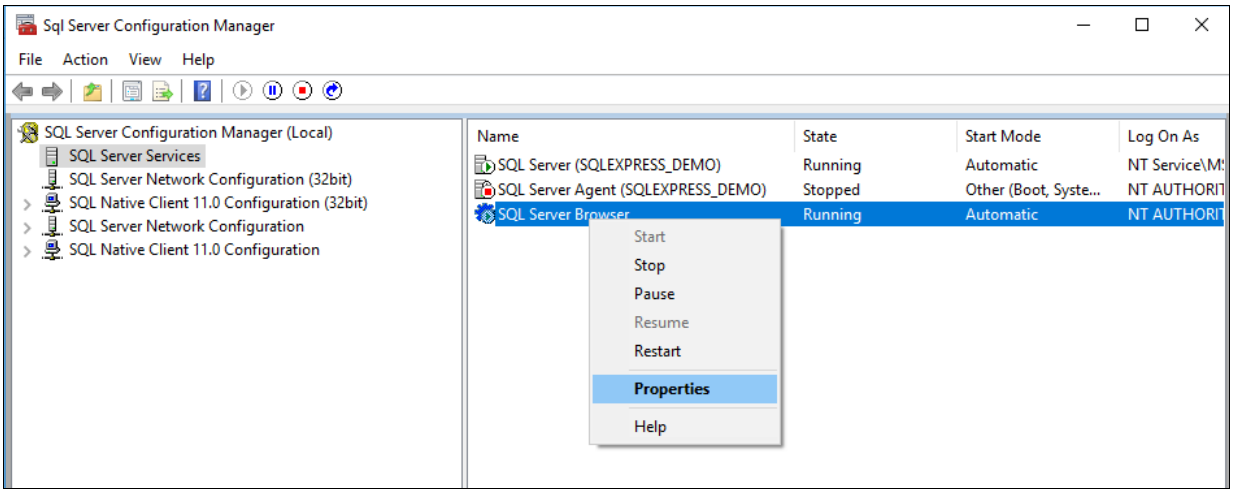
1. Open the **SQL Server Configuration Manager**. The path to this program in the Windows menu may differ slightly on different versions of Windows.

2. Click on the **SQL Server Services** node.

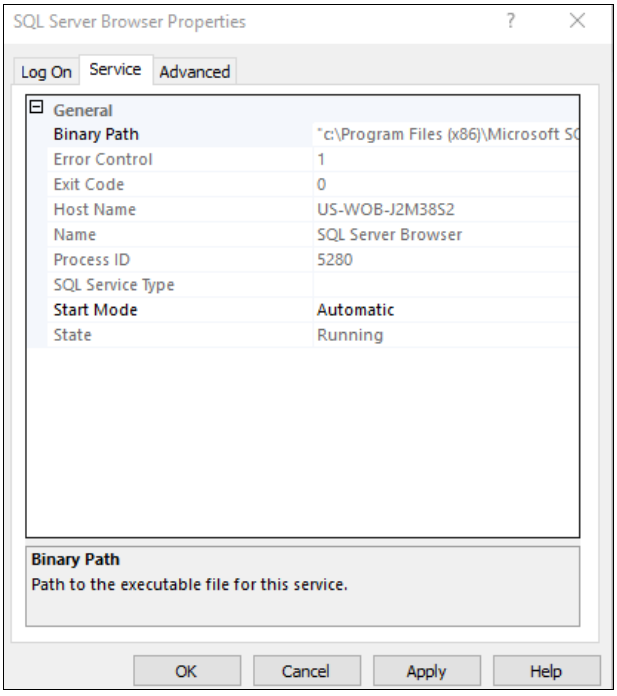


In this example, SQL SERVER Browser is set to start automatically already. If yours is not, use the following steps.

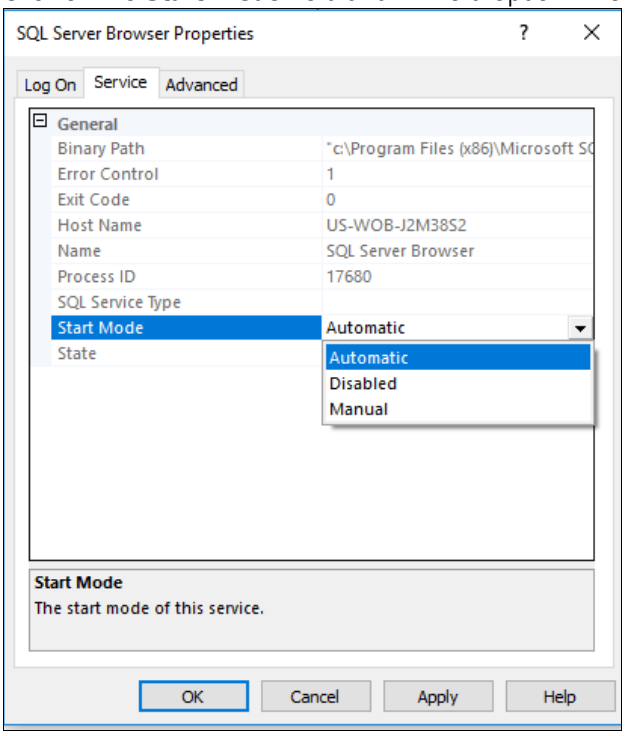
3. Right-click on the **SQL Server Browser** in the right side of the display and select **Properties** from the pop-up menu.



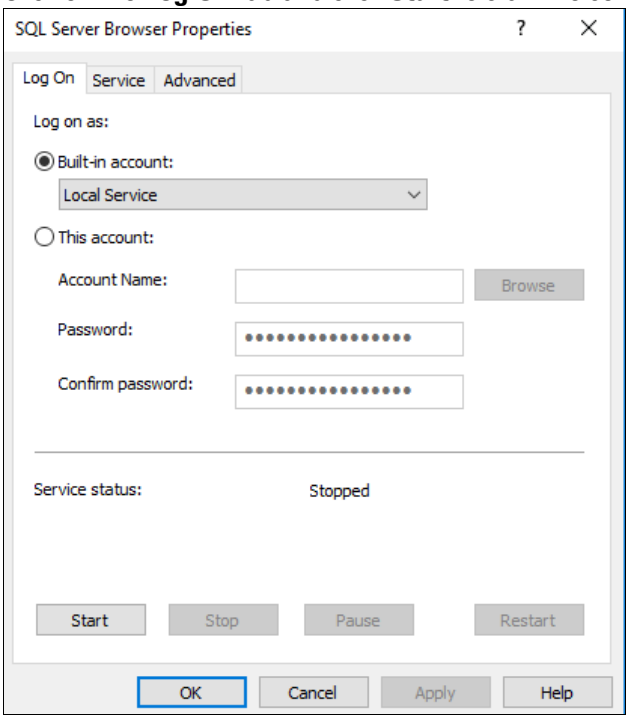
4. Click on the **Service** tab.



5. Click on the **Start Mode** field and in the dropdown field on the right set the value to **Automatic** and click **Apply**.



6. Click on the **Log On** tab and click **Start** to start the service, and click **OK**.



Now install Custodial Integrator.

Custodial Integrator Computer Install or Upgrade

Perform the following installation by first logging into the Custodial Integrator Computer with a Windows login that is a member of the Administrators group on that computer and then completing all steps presented by the installation wizard.

Note: If you are upgrading Custodial Integrator, please review the release notes between the version you currently run and the most recent version to understand changes and actions you may need to take to complete the upgrade. For example, if this is an upgrade from a version prior to 3.5.002, you will be prompted to re-enter the username and password. In version 3.7.001, encryption options changed and instructions are in the Release Notes. Any version upgrading to 3.16.001 may require edits to the runCI.bat. Refer to the *Custodial Integrator Release Notes* at http://www.byallaccounts.net/Manuals/Custodial_Integrator/apx/CI_releasenotes.pdf.

1. Open an Internet Explorer browser window and enter the Custodial Integrator Application installation URL: https://www.byallaccounts.net/CI/
2. Enter the distribution page user name and password when prompted. This is **not** the same user name and password you use to access the WebPortfolio application.
3. Click on the **Advent APX**® link.
4. Click on the **Install Custodial Integrator Now** link.
5. Choose to **Run** the file now (if your browser provides this option) or **Save** the file to disk and then double-click on the saved file to run the installation.
6. If the wizard determines that it must install pre-requisite software it will now display a listing of these items. Click **OK** to continue.
7. If the wizard determines it needs to install Java on the Custodial Integrator Computer then it will present installation screens for this product now.
    a. Review and accept the License Agreement.
    b. The Java Runtime installation proceeds and finally the **Java Setup - Complete** screen displays. Click **Finish** to continue in the wizard.
8. The wizard will proceed to install the Custodial Integrator Application on the Custodial Integrator computer.
    a. In Welcome to the InstallShield Wizard for Custodial Integrator click Next.
    b. In Setup Type, the Complete option is the default and will install Custodial Integrator in the default location with all documentation. Click Next.
    c. In the Database screen you will see that the default name for the Custodial Integrator database is BaaWpAci. If you choose to use a different name then change the text in the Name field. Click Next.
    d. In CI SQL Login Parameters you will provide the SQL Server login id and password that the Custodial Integrator application will use to access its database. This login must minimally have the following SQL Server privileges for the database named in the previous dialog: public, db_datareader, db_datawriter. Enter the SQL login id and password now. Click Next.
    e. In SQL Server, use the Browse… button to locate the SQL Server's computer/instance name. SQL Server will by default be named hostname/SQLExpress where hostname is the name of the computer – be sure to

choose the correct instance. Under Connect using you may select Windows authentication if the Windows Login you are using has "sa" or "dbo" privileges in the target database. If your Windows Login is not authorized in this way then enter a login id and password for a SQL user that has "sa" or "dbo" privileges. Click Next.

f. In Start Copying Files review the settings. Click Back to go back and alter settings. Once the settings are correct click Next.

g. The Custodial Integrator Application will be installed and the Custodial Integrator database will be created within the target Microsoft SQL Server. In **InstallShield Wizard Completed** click **Finish.**

Configure Custodial Integrator

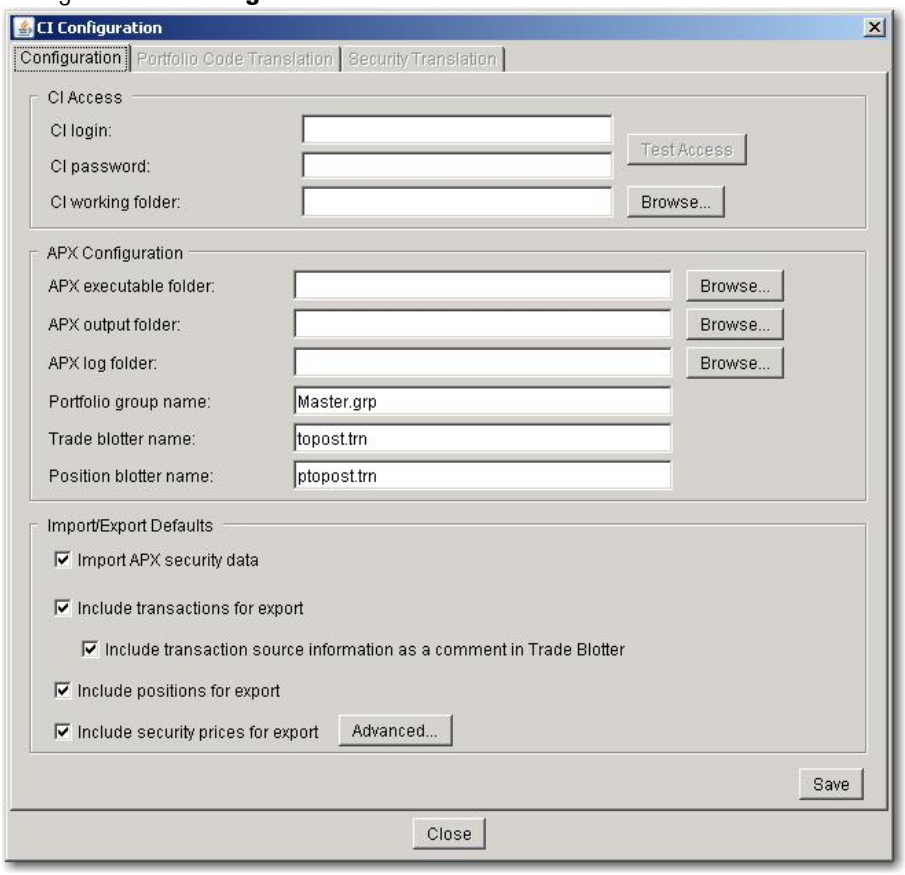
The file \Program Files\Custodial Integrator\CI.ini provides a starting point running Custodial Integrator. Determine which Custodial Integrator parameters you must use by reviewing the **Custodial Integrator** Parameters section on page 30.

1. Edit the \Program Files\Custodial Integrator\CI.ini file to add any required Custodial Integrator Parameters.
2. Run Custodial Integrator by double-clicking on the Desktop shortcut "Custodial Integrator". If the application does not come up or reports an error at startup, please check the **Troubleshoot Install Problems** section on page 28. If the problem continues to occur contact ByAllAccounts Technical Support for assistance.

Once you run Custodial Integrator without error, the Custodial Integrator installation is complete and you can move on to configuration and installation verification. Next steps include verification of the basic configuration information you have provided so far (in CI.ini), providing Custodial Integrator additional configuration information required to access your financial information in WebPortfolio as well as to interact with your APX installation, and troubleshooting any issues you encounter in your environment.

1. Create one test Portfolio in your APX installation. You will need this portfolio to verify the Custodial Integrator installation. Create a Group within APX, make note of the group's name, and put the test Portfolio into that Group. Your Windows user must minimally have read access to that group. Contact your APX administrator if you need assistance with this step.
2. Run Custodial Integrator by double-clicking on the Desktop Shortcut.
3. The main view of Custodial Integrator displays, showing the steps to move data from WebPortfolio to APX.

4. In step 1 in the user interface, click the **Setup…** button. Custodial Integrator now displays the CI Configuration dialog with the **Configuration** tab shown as follows:



5. In the *CI Access* section, enter the following test credentials (note: these credentials give you access to test data only and should only be used to validate the Custodial Integrator configuration):

   a. CI login: CITEST

   b. CI password: WEEPINGWILLOW1

   Note: This login and password give you access to test data only and should only be used to validate the Custodial Integrator configuration. Later, access your own data using the CI Login and CI Password assigned to you by ByAllAccounts.

   c. Custodial Integrator working folder: Create a working folder for Custodial Integrator to use for temporary files. Enter that folder name into this field or use the 'Browse…' button to select the folder within a Windows explorer.

6. Verify that Custodial Integrator can successfully use the Custodial Integrator test credentials to access data from WebPortfolio. Click the **Test Access** button to perform the test. The results of the Test Access are displayed in a popup dialog and at the bottom of the *CI Access* section.

7. In the *APX Configuration* section make the following changes:

   a. Enter your APX root installation folder in *APX executable folder.*

b. Create a folder that Custodial Integrator can use to transfer data to/from APX. This folder's name should not contain any spaces. Grant your Windows user write access to that folder. Enter this folder in the *APX Output Folder* field.

c. Determine where the APX Import/Export utility is configured to write its log files. Enter this folder into the *APX Log Folder* field.

d. Replace 'Master.grp' with the name of the test Group you created in APX.

e. Leave the Trade blotter name and Position blotter name as is for now.

8. In the *Import/Export Defaults* section, the following options are available to specify which types of files should be generated for APX in a typical cycle. If an option is checked then that type of file will be generated during the export step unless you override this setting in the main view.

   - Import APX security data
   - Include transactions for export
   - Include positions for export
   - Include security prices for export

   You will want to minimally check the following in the *Import/Export Defaults:*

   a. Check *Import APX security data* to import APX Security Info and Security Type information from APX each time you run Custodial Integrator to perform a data translation.

   b. Check *Include transactions for export* to transfer Transactions from WebPortfolio to APX

   c. Check *Include positions for export*

9. Click **Save** to save your configuration settings.

10. Click the **Portfolio Code Translation** tab. Custodial Integrator will load the test accounts available and display them in the **Untranslated WebPortfolio accounts** section. Verify that you see accounts listed in this section. Define at least one Portfolio code translation. You can do this by selecting any one of the test accounts from the untranslated section, clicking **Add**, selecting your test portfolio from the **APX portfolio codes** section, and clicking **Add** again.

11. Click **Close** to close the CI Configuration tab and return to Custodial Integrator's main view.

12. Verify that Custodial Integrator can load data from APX by clicking **Import**. Custodial Integrator will invoke the APX Import/Export utility to export the sec.inf and type.inf files for use in Custodial Integrator. Custodial Integrator will also download financial information for the selected WebPortfolio accounts into Custodial Integrator and attempt to match securities to those defined in APX. The result of your import will most likely be an error message with text 'error validating securities' (meaning that Custodial Integrator could not locate some of the securities referenced by WebPortfolio in the APX security master) and the appearance of the 'Security Translation…' button. If there are no security translation errors, the **Import** button will be grayed out and the **Export** button will be active. In either case, you have verified the Custodial Integrator installation and configuration.

13. Click **Setup** and go to the Portfolio Code Translation tab. Delete the one entry you created by selecting that row in the table at the top of the dialog and clicking **Delete.** Now navigate to the Configuration tab and remove the

Custodial Integrator Access test credentials you entered. Custodial Integrator will later be configured using the actual Custodial Integrator credentials that provide access to your firm's financial information in WebPortfolio. Click **Save** to save the changes.

14. Close the Configuration dialog and exit Custodial Integrator.

Consider password encryption models

As of version 3.7, Custodial Integrator supports two models for managing encrypted user logins and passwords: default and enhanced. The default model stores the encrypted login/password combinations in the database and requires no special instructions. The enhanced model uses an on-disk keystore model that uses the password-protected keystore file (ciksfile) to encrypt and decrypt user logins and passwords which are stored in the database in an encrypted state. That model is more complicated to set up, especially when multiple instances of Custodial Integrator access the same database, described in Setup for Multiple Custodial Integrators that access mode the same database, below.

Note: Do not remove the PBEKeysetPass or KSKeysetPass parameters. Do not change them for any reason without express guidance from ByAllAccounts Technical Support. If the value of either does need to be changed, then the value must be changed to match in all instances.
These parameters are described in Database, page 39.

Install multiple instances that access the same database

This section explains what you need to know if you want multiple instances of Custodial Integrator to access the same database.

Note: If you are upgrading Custodial Integrator, please review the release notes between the version you currently run and the most recent version to understand changes and actions you may need to take to complete the upgrade. For example, instructions regarding changes to encryption are included in the section for version 3.7.001. Refer the *Custodial Integrator Release Notes* at
http://www.byallaccounts.net/Manuals/Custodial_Integrator/apx/CI_releasenotes.pdf.

Note: Each instance must use the same user login and password and only one instance should run at any given time.

Setup for Multiple Custodial Integrators that access mode the same database

Default encryption model

The default encryption model for user logins and passwords does not necessitate any special considerations when installing multiple instances of Custodial Integrator that access the same database. If you are using that model, you can skip the rest of this section.

Enhanced encryption model

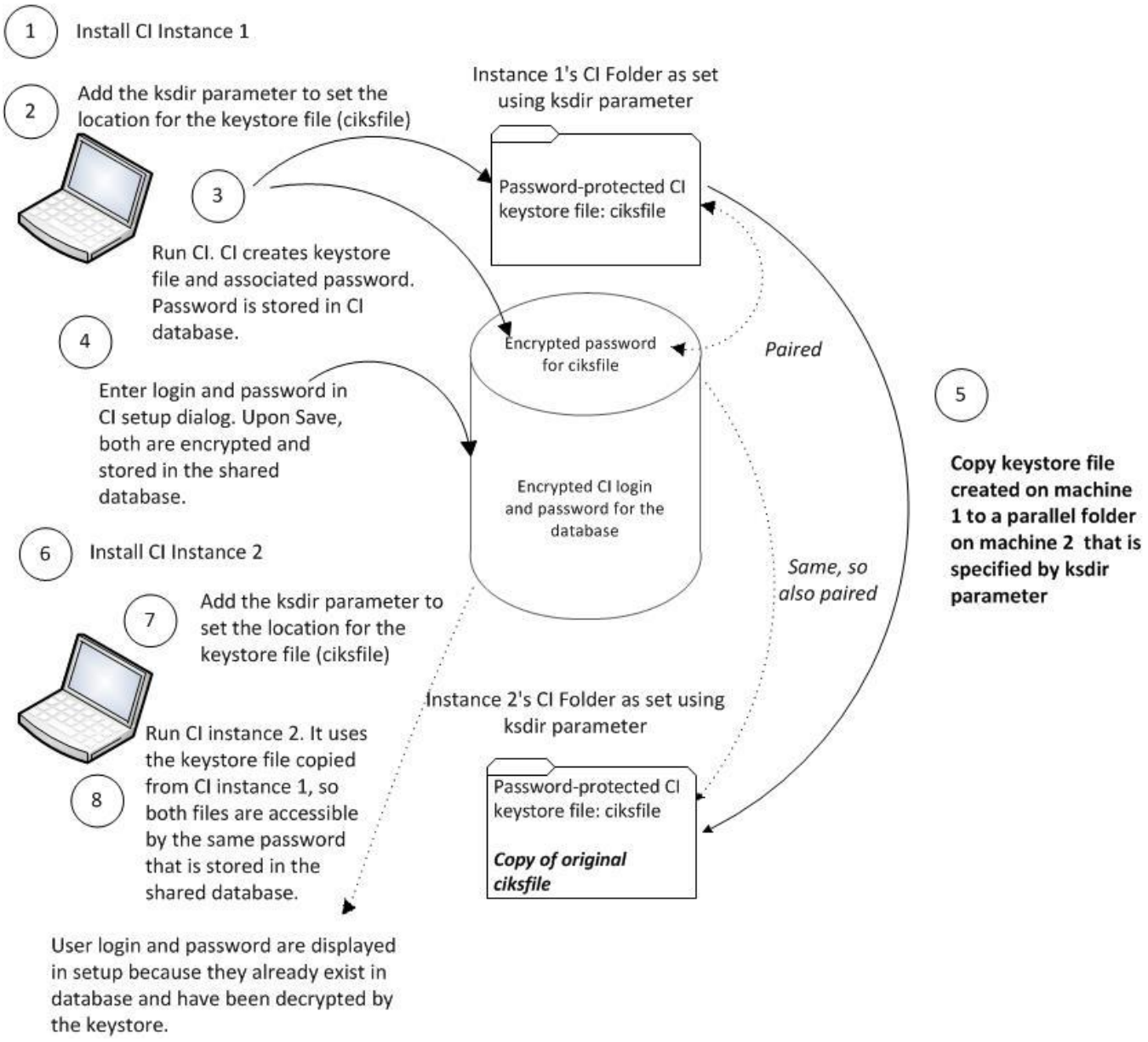
Because the enhanced encryption model is more complicated than the default encryption model, it requires special instructions as described here.

Note: If you are upgrading Custodial Integrator from version 3.15.001 or earlier, you may need to edit each runCI.bat instance as described in the release notes for 3.16.001. Refer to
http://www.byallaccounts.net/Manuals/Custodial_Integrator/apx/CI_releasenotes.pdf.

The password for the keystore file is stored in the database and is uniquely paired to the keystore file, and each instance of Custodial Integrator that uses the same database must use the same pair. The diagram below illustrates the process to use to ensure that they use the same pair.

1. Install Custodial Integrator instance 1.
2. In the runCI.bat (or CI.ini depending on where the database name is set), set the ksdir parameter to the location for the keystore file (ciksfile).
3. Run Custodial Integrator instance 1 for the first time after install. That creates the keystore file and its associated password which is stored in the Custodial Integrator database. The keystore file is stored in the location that was defined using the ksdir parameter in step 2.
4. Enter the user login and password in the Custodial Integrator setup dialog. Upon saving, the user login and password are encrypted and stored in the shared database.
5. **Before** installing Custodial Integrator instance 2, copy the keystore file created on machine 1 to a parallel folder on machine 2.
6. Install Custodial Integrator instance 2 on machine 2.
7. As you did for Custodial Integrator instance 1, set the location for the keystore file using the ksdir parameter.
8. Run Custodial Integrator instance 2. Custodial Integrator instance uses the keystore file that was copied from Custodial Integrator instance 1, so both keystore files are accessible by the same password that is stored in the shared database.

User login and password are displayed in setup because they already exist in the database and have been decrypted by the keystore.


1
Install CI Instance 1
2
Add the ksdir parameter to set the location for the keystore file (ciksfile)
3
Run CI. CI creates keystore file and associated password. Password is stored in CI database.
4
Enter login and password in CI setup dialog. Upon Save, both are encrypted and stored in the shared database.
6
Install CI Instance 2
7
Add the ksdir parameter to set the location for the keystore file (ciksfile)
8
Run CI instance 2. It uses the keystore file copied from CI instance 1, so both files are accessible by the same password that is stored in the shared database.
User login and password are displayed in setup because they already exist in database and have been decrypted by the keystore.
Instance 1's CI Folder as set using ksdir parameter
Password-protected CI keystore file: ciksfile
Encrypted password for ciksfile
Encrypted CI login and password for the database
Paired
Same, so also paired
5
Copy keystore file created on machine 1 to a parallel folder on machine 2 that is specified by ksdir parameter
Instance 2's CI Folder as set using ksdir parameter
Password-protected CI keystore file: ciksfile
Copy of original ciksfile

Install multiple instances of Custodial Integrator on same machine with different databases

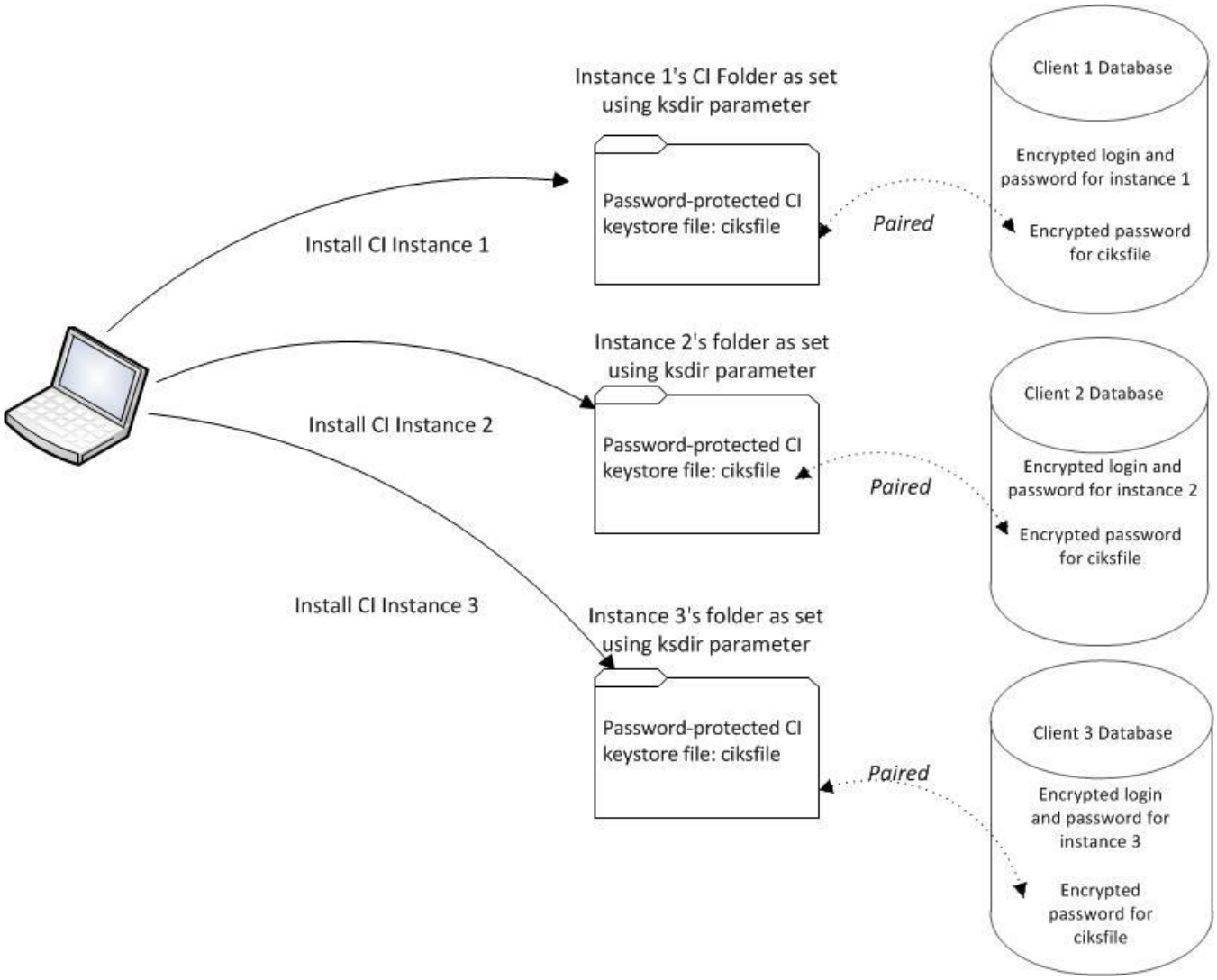
This section describes how to install multiple instances of Custodial Integrator on the same machine that access different databases. The instructions vary a bit depending on whether you use the default or enhanced encryption model for the logins and passwords. For information about the encryption methods, refer to Consider password encryption models, page 23.

Use the setup shown in the following diagram when multiple instances of Custodial Integrator are installed on one machine and each instance accesses its own database.

Use the following procedure to get all Custodial Integrator instances working properly:

1. For each client create a separate *runci<client name>.bat* script that specifies the following parameters settings which must be unique for each client:
   - the client's database name (*dbname:*)
   - if you are using the enhanced encryption model, use the *ksdir* parameter set the location for the uniquely-paired keystore file (*ciksfile*). The password for the keystore file stored in the database is uniquely paired to the keystore file for the instance.
   - If you use bulk insert, specify the bulk insert parameters (*usebulkinsert:y* and *bulkinsertdatafolder:*)
   - If you are upgrading Custodial Integrator from version 3.15.001 or earlier, you must edit runCI<*clientname*>.bat as described in the release notes for 3.16.001. *Refer to* http://www.byallaccounts.net/Manuals/Custodial_Integrator/apx/CI_releasenotes.pdf.
2. When setting up Custodial Integrator for each client also configure a separate location for their Custodial Integrator working and output folders.
3. Run Custodial Integrator for one of the customers.
4. Go to the Setup > Configuration tab. If this is a Custodial Integrator upgrade from a version prior to 3.5.002, you will be prompted to re-enter the username and password.
5. Make sure the Custodial Integrator working folder is set to the correct folder for the instance.
6. Repeat steps 3- 5 for the additional Custodial Integrator instances.
7. If you are using the enhanced method and used the ksdir parameter to set the location for the keystore file, the process should create a separate keystore file (filename: Custodial Integrator ksfile) in the that folder for each customer, and should allow each instance of Custodial Integrator to run without problems.

Note: If you need to create new CI instances in the future, you'll need to create a separate working folder and INI folder for the new instance and you'll need to add the appropriate parameters to the CI.ini and runCI.bat files for each new instance.


Instance 1's CI Folder as set using ksdir parameter
Client 1 Database
Encrypted login and password for instance 1
Encrypted password for ciksfile
Password-protected CI keystore file: ciksfile
Paired
Install CI Instance 1
Instance 2's folder as set using ksdir parameter
Client 2 Database
Encrypted login and password for instance 2
Encrypted password for ciksfile
Password-protected CI keystore file: ciksfile
Paired
Install CI Instance 2
Install CI Instance 3
Instance 3's folder as set using ksdir parameter
Client 3 Database
Encrypted login and password for instance 3
Encrypted password for ciksfile
Password-protected CI keystore file: ciksfile
Paired

Troubleshoot Install Problems

General Techniques

Custodial Integrator logs information about its activities to a log file. If Custodial Integrator encounters an error, the error should appear with full detail in the log file. Custodial Integrator log files can be found in the \Program Files\Custodial Integrator\log folder. Custodial Integrator log files have a unique, generated name. To obtain the Custodial Integrator log file associated with the most recent Custodial Integrator session, sort the Custodial Integrator log files by date in the Windows Explorer. Custodial Integrator will remove log files older than 14 days, so if you need to retain a log file for support purposes, copy the log file to another folder.

You can direct Custodial Integrator to provide additional information for troubleshooting. Use the **debugall=**1 CI.ini parameter to direct Custodial Integrator to log more detailed information about what it is doing. This parameter is described in the **Custodial Integrator** Parameters section on page 30.

Known Issues

This section describes common installation problems and suggestions for troubleshooting or working around them.

General Errors

Symptom	Possible Cause	Action
Configuration **Test Access** reports 'Error loading WebPortfolio Data'	Incorrect Custodial Integrator proxy configuration for Internet access	Check the Custodial Integrator log file for more details. Contact ByAllAccounts Technical Support for assistance.
CryptoException error	Password encryption problem. Custodial Integrator database has a stored keystore password in the database and a keystore file exists in the ksdir, but they do not match.	1. Close Custodial Integrator. 2. Delete the ciksfile in the ksdir folder 3. Restart Custodial Integrator 4. Go to Setup and reenter your user id and password (you may be prompted to do that as soon as Custodial Integrator starts). 5. Click on the "Test Access" button on the right of the screen to verify that the login and password are correct. 6. Proceed as normal. If you have multiple Custodial Integrator instances on a single database and are using the enhanced encryption model, then after these steps you need to copy the ciksfile from the ksdir of the Custodial Integrator instance that was just recovered to each ksdir folder (one per instance) before running these other Custodial Integrator instances. See Install multiple instances that access the same database on page 23 for more information.

Database Errors

In the case of a database error, Custodial Integrator will generally report one of the following messages:

- An error occurred when trying to connect to the database.
- An internal error occurred.

Additional details may also be presented in the popup error box. For all errors of this type, check the Custodial Integrator log file for more detailed information.

Symptom	Possible Cause	Action
Error connecting to SQL Server with error message "Login failed for sa user - not associated with a trusted SQL Server connection".	SQL Server not configured for mixed-mode authentication.	Check with your SQL Server administrator.
[Microsoft][SQLServer Driver for JDBC]Error establishing socket.	SQL Server not configured for TCP/IP OR insufficient resources available on the SQL Server computer to allocate a new socket.	Check with your SQL Server administrator.

CUSTODIAL INTEGRATOR CONFIGURATION REFERENCE

Custodial Integrator Parameters

You can configure the CI Application by adding runtime parameters to the CI.ini file. The basic form of the parameters contained by CI.ini is:

parameter name=value

where parameter name is one of the parameters described in the following sections and "=" separates the parameter from its value, and "value" is the value you supply for the parameter. Parameters are separated from each other by a linefeed. Parameters do not need to be listed in any particular order.

The following example defines your APX system currency as "eu" and your SQL Server password as "mypassword":

axyscur=eu

sqlpw=mypassword

CI Customizations

These parameters customize the behavior of your CI installation.

Parameter label	Parameter value(s)	Default	Description	Example
apx4PositionType				
	character	(none)	Specifies the one character to be output for APX v4 in the 'source type' field of the position blotter. Applies ONLY to APX versions 4.x and later. If the parameter is not set or the value is empty, then CI assumes that the APX version is less than 4 and will not output the 'source type' field. Therefore for APX 4.x, this parameter must be set in CI.ini or runCI.bat. However, if for a version prior to APX 4.x DO NOT set this parameter, or the generated position blotter will be incorrect for that earlier APX version.	apx4PositionType=p
apx4lottype				
	character	(see description)	Specifies the one character to be output in the "source type" field of the position lots blotter. Applies ONLY to APX versions 4.x and later. The parameter is optional and is ignored unless apx4positiontype is set and lots are enabled and exported. If apx4PositionType is set but no value is set for apx4lottype, then apx4lottype defaults to the letter 'l' (which is lower case L).	apx4LotType=l.
apxixwait				
	{number}	20	Number of minutes for CI to wait for the ApxIx utility to produce output when invoked.	apxixwait=10

defaultAccountIdentifier				
	WPAccountNumber WPInternalID WPAccountName	none	Used to set the default account identifier for untranslated accounts to one of three values: WPAccountNumber sets it to the Account Number from WebPortfolio. WPAccountName sets it to the account name from WebPortfolio. WPInternalID sets it to the internal ID from WebPortfolio. You should consistently use the same source for the default account identifier.	
inputFolder				
	(full path or a path relative to the CI startup folder)	(CI startup folder)	Used to specify the location of custom transaction translations, holding filters, and transaction filters files. If the input folder is specified and the folder does not exist, or the user does not have access to it, an error message is displayed to the user and the application exits.	inputfolder=C:\CI \data\customtransl ations
posstmt				
	y n	none	If value is 'y' then CI will generate text position statements in the program files\custodial integrator\exported\pos folder whenever positions are included in the export process. One position statement is generated for each APX portfolio.	posstmt=y

ignorePositionalExtServLevelAccounts				
	y n	n	Specifies whether accounts that have the value "Positional" in the EXTERNAL_SERVICE_LEVEL field should be ignored in CI. When the parameter is set to 'y', such accounts will not appear in the list of untranslated accounts and will not be translated in CI Autorun. Therefore, these accounts will never show in CI exported files. Effective both in CI interactive and in CI autorun modes. **Important note for those upgrading from any release prior to CI 3.15**: Refer to your release notes for 3.15 for important upgrade information.	ignorePositionalExtServLevelAccounts=y

APX Customizations

These parameters provide information about your APX configuration or about how you would like data translated for APX.

Parameter label	Parameter value(s)	Default	Description	Example
alwaysSetMarkMarket				
	y n	n	With the APX version 4, Mark-to-Market is always required to have a value (y/n) independently of the trade currency. If you are using APX version 4, you must set this parameter to 'y'. This setting tells CI to automatically determine the Mark-to-Market value by assigning 'n' for deposit and withdrawal transactions and 'y' for all other transaction types and for all positions and lots. Note: When this parameter is set to 'y' CI ignores the setting of *defMarkMarket.* Note: If you are using any of APX versions 1 through 3, do not set parameter, or set it to 'n' so CI will keep the current behavior and determine Mark-to-Market value based on the currency (system/foreign) and on the value of parameter *defMarkMarket*.	*alwaysSetMarkMarket=y*
axysacbond				
		f	The letter code in your APX configuration for the asset class 'Fixed Income'. Older versions of APX may use a letter that is different from 'f', which can adversely affect CI's data translation for APX.	axysacbond=i
axysaccash				
		c	The letter code in your APX configuration for the asset class 'Cash'. Older versions of APX may use a letter that is different from 'c' and this can adversely affect CI's data translation for APX.	axysaccash=s

APX Customizations (continued):

Parameter label	Parameter value(s)	Default	Description	Example
axysbrok				
		$brok	The value to use for the Broker column when a value is required for this column of a transaction (e.g. when Commission has a value, the Broker column must also be valued). The value must not be an empty string.	axysbrok=dummyb
axyscommission				
	y n	Y	If 'y' then if commission value is available for a transaction, both the Commission and Broker columns are filled in for the transaction. $brok is used for the Broker value (see *axysbrok* parameter). If 'n' then the APX Commission and Broker columns are not filled in, regardless of the information available for a transaction.	Axyscommission=n
axyscur				
		us	The currency code you have defined as your APX system currency.	Axyscur=eu
axyslotlocation				
		253	The value to use in the lot location column.	axyslotlocation=253
defdivquan				
			For Dividend transactions with no reported Quantity, use this value in the quantity column in the topost.trn transaction file.	Defdivquan=1
defmarkmarket				
	y n	n	Transactions and positions against a currency other than your APX system currency require a value for the Mark to Market field. This parameter defines the value to use for the Mark to Market field when this field is required. **Note**: This parameter is ignored if the *alwaysSetMarkMarket* parameter is set to "y".	Defmarkmarket=y

APX Customizations (continued):

Parameter label	Parameter value(s)	Default	Description	Example
defperfcw				
	y n	n	Value to use in the Perf/CW column of APX topost.trn file.	Defperfcw=y
fasectype				
	APX security type		Gives the APX security type that you use to identify Fixed Annuities. This causes the following special treatment for these positions in the position file: if no units or price are given by the Financial Institution then Quantity is set to 1 and Price is set to the position's market value. If units are given by the Financial Institution but no price (or a $0 price) then the price is computed by dividing market value by the units.	Fasectype=vfus
defFeeType				
	APX security type	epus	The default APX Security Type to be used when translating "fee"* transactions.	DefFeeType=epus
defFeeSymbol				
	APX security symbol	custfee	The default APX Security Symbol to be used when translating "fee"* transactions.	DefFeeSymbol=custfee
overwriteposzerounits				
	useMV use1 none	useMV	Used to handle positions where neither quantity nor price is reported by the financial institution. If not specified or useMV is specified then the quantity is set to the market value of the position and the price is set to $1 (or 100 for fixed income). If use1 is specified then set the quantity to 1 and the price equal to the market value. If none is specified then do not report any quantity for the position in either the reconciliation file or the position file.	Overwriteposzerounits=use MV

APX Customizations (continued):

Parameter label	Parameter value(s)	Default	Description	Example
overwriteprice				
	y n	n	CI only provides prices for securities that do not already have a price in the target price file. Use this option to cause CI to provide a price for all securities that it processes and to overwrite any price already present in the target APX price file for these securities.	overwriteprice=y
taxSymbol				
		taxwith	The Axys Security Symbol for use in general tax transactions.	taxSymbol=tax1
foreignTaxSymbol				
		fortax	The Axys Security Symbol for use in foreign tax transactions.	foreignTaxSymbol=frtax
xlateFeeDesc<n>**				
			This parameter specifies the string that should be compared to the "fee"* transaction's description. If the description contains this string then the transaction is considered to "match" this translation and corresponding xlateFeeType and xlateFeeSymbol parameter values are used.	xlateFeeDesc1=Tax
xlateFeeType<n>**				
		epus	This parameter specifies the APX Security Type to be used for "fee"* transactions that match the corresponding xlateFeeDesc parameter.	xlateFeeType1=exus
xlateFeeSymbol<n>**				
		custfee	This parameter specifies the APX Security Symbol to be used for "fee"* transactions that match the corresponding xlateFeeDesc parameter.	xlateFeeSymbol1=with

APX Customizations (continued):

Parameter label	Parameter value(s)	Default	Description	Example
xlateFeeSymbol<n>**				
		custfee	This parameter specifies the APX Security Symbol to be used for "fee"* transactions that match the corresponding xlateFeeDesc parameter.	xlateFeeSymbol1=with

*CI defines a "fee" transaction as any one of the following:

a) A transaction with a WebPortfolio transaction type of: Fee, Investment Expense, or Service Charge.
b) A non-cash security transaction with a WebPortfolio transaction type of: Debit transaction that has been determined to translate to an APX "lo" type transaction. This covers various cases of expenses and tax deductions that might need special security type and symbol translations.
c) A "dp" or "wd" transaction generated during CI processing of any type of WebPortfolio transaction. Such transactions are typically generated while processing a transaction that implies that a "fee" also occurred, like a Sell transaction that really represents a sell of a fund to pay a fee and so will be generated for APX as a "sl" followed by a "dp" (fee).

** CI users can specify that different APX Security Type and/or APX Security Symbols be used for "fee" transactions whose description contains specific text. This is accomplished via the three parameters xlateFeeDesc, xlateFeeType, and xlateFeeSymbol. These parameters may appear either as a pair (xlateFeeDesc along with either xlateFeeType or xlateFeeSymbol) or as a trio (all parameters present) in the CI.INI file. The parameters are correlated by the trailing number in their name. An unlimited number of correlated parameters can be specified (e.g. xlateFeeDesc2, xlateFeeType2, etc.). Translations are applied in the order that they appear in the file. Once a match is made only that translation is used. If no translation matches then any default parameters (defFeeType, defFeeSymbol) are used.

Examples:

1. Customize translation for "RECORDKEEPING" fee transactions at all Financial Institutions to "epus" "expense":

   xlateFeeDesc1=RECORDKEEPING FEE

   xlateFeeType1=epus

   xlateFeeSymbol1=expense

2. Customize translation for "RECORDKEEPING" fee transactions at all Financial Institutions to "epus" "expense" and also customize translation for "TAX":

   xlateFeeDesc1=RECORDKEEPING FEE

   xlateFeeType1=epus

   xlateFeeSymbol1=expense

xlateFeeDesc2=TAX

xlateFeeType2=epus

xlateFeeSymbol2=with

Corporate Firewall

If the CI Computer must access the Internet through a proxy, you will need to configure your proxy settings for CI using one or more of the following parameters:

Parameter label	Default	Description	Example
proxyhost		The name or IP address of the proxy host for Internet access.	proxyhost=comp1
proxyport	443	The proxy port for https Internet access.	proxyport=123
proxyuser		Username for Basic authentication at the **proxyhost**.	proxyuser=someu
proxypw		Password for **proxyuser**.	proxypw=mypw

Database

The following command line parameters can be used to control CI's database access.

Parameter label	Default	Description	Example
sqlhost	localhost	The name or IP address of the SQL Server computer serving the CI database.	sqlhost=smac
sqllogin	sa	Login to use to access SQL Server using SQL Server authentication mode.	sqllogin=mylogin
sqlpw	applesandoranges	Password for "sa" login, or for sqllogin (if specified).	sqlpw=mypw
dbname	BaaWpAci	Name of CI database.	dbname=mydb
ksdir	(CI working folder)	Optional CI.ini parameter, only used for enhanced encryption model of user login and password. Specifies the location of the folder where the keystore file (ciksfile) resides. This file contains the encryption/decryption key for the user credentials for the enhanced encryption model. By default, the keystore file is created in CI working folder. For information about how to use this parameter when installing or upgrading multiple CI instances that access the same database see page 23. For installing or upgrading multiple	ksdir=<full path of keystore folder>

		instances of CI on same machine with different databases, see page 26.	
PBEKeysetPass	(PBE encryption key)	This is a mandatory parameter. It provides a password used in the encryption of select sensitive data stored in the CI database. This parameter is automatically set during CI installation or CI upgrade from versions earlier than 3.12. **Note:** This parameter must NOT be removed, and its value should NOT be changed without guidance from ByAllAccounts Technical Support. For setups with multiple CI instances see page 23.	PBEKeysetPass=<fixed string value>
KSKeysetPass	(PBE encryption key)	This is a mandatory parameter. It provides a password used in the encryption of select sensitive data stored in the CI database. This parameter is automatically set during CI installation or CI upgrade from versions earlier than 3.12. **Note:** This parameter must NOT be removed, and its value should NOT be changed without guidance from ByAllAccounts Technical Support. For setups with multiple CI instances see page 23.	KSKeysetPass=<fixed string value>

Debugging

The following parameter is used to control the debugging capabilities of CI.

Parameter label	Parameter value(s)	Default	Description	Example
debugall	1 0	0	Use a value of 1 to enable all CI debugging features, including enhanced error and event logging.	debugall=1