MORNINGSTAR Wealth

Custodial Integrator

Installation Guide

Custodial Integrator Product Version:	V3.16
Document Version:	33
Document Issue Date:	May 11, 2021

Technical Support:	(866) 856-4951
Telephone:	(781) 376-0801
Fax:	(781) 376-8040
Web:	byallaccounts.morningstar.com

Table of Contents

About the Manual

Audience and Format

This manual describes the system requirements for the Custodial Integrator ("CI") product, the procedure for installing CI, and how to perform initial configuration of CI. 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.

Install Requirements

System Overview

The CI architecture consists of the following three main components:

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

CI interacts with the following other components:

- The WebPortfolio Service web site at URL https://www.byallaccounts.net/

CI requires the following third-party software:

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

See Required Software, page 5 for version requirements

The remainder of this manual refers to the following:

- CI Computer: the computer on which the CI application is installed and run. You will install the Java Runtime and CI on this computer.
- Database Computer: the computer where Microsoft SQL Server is installed. The CI database will be created on this computer.

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

System Requirements

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

CI Computer

The following table defines the base system requirements for the CI 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 CI 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 CI 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 5. 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 CI to ensure you have the required CPU, disk space, RAM, and operating system.

The disk requirement for the CI Database is 70 MB.

Required Software

CI 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 2012 or later is required, including Express versions. For full TLS 1.2 support the minimum requirement is SP4 for SQL Server 2012 and SP3 for SQL Server 2014. Earlier 2012 and 2014 versions may require additional updates to support TLS 1.2 Later versions (2016, 2019) all support TLS 1.2. *	Available from Microsoft	Depends on the SQL Server version. Refer to Microsoft documentation.
Java Runtime Environment (JRE)	CI Computer	Oracle Corporation	1.8 or later	CI distribution: 1.8 update 261 is included in CI kit	239 MB (for 64-bit) 180 MB (for 32-bit
Microsoft OLE DB Driver for SQL Server (MSOLEDBSQL)	CI Computer	Microsoft	18.2 or later	CI distribution: Microsoft OLE DB Driver for SQL Server (MSOLEDBSQL) 18.5.0.0 is included in CI kit	11.1 MB

* For information about TLS 1.2 support see https://support.microsoft.com/en-us/help/3135244/tls-1-2-support-for-microsoft-sql-server

Additional Considerations

1. Database Computer

   If you already have a version of Microsoft SQL installation supported by CI, you may use that installation to house the CI 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 CI Computer or on a separate Database Computer.

2. Internet Connectivity

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

3. Multiple Users

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

INSTALL CUSTODIAL INTEGRATOR

Overview

The CI 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 CI Application on the CI Computer. This install includes:
    a. Java Runtime (pre-requisite)
    b. CI product installation
    c. CI Database initialization
4. Configure CI
5. Verify Installation

Depending on your configuration, you may need to refer to Consider password encryption models

As of version 3.7, CI 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 CI access the same database, described in Setup for Multiple CIs that access the same database.

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 34.

Install multiple instances that access the same database, page 21 or Install multiple instances of CI on same machine with different databases, page 24.

Software Distribution

Custodial Integrator

The CI 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 CI installation, please complete the following steps:

1. Verify System Requirements for both the CI 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 CI Computer, determine the version. 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 CI computer. If another application on the CI 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 CI on an alternate computer.
3. Windows Administrative Access
   a. Obtain a Windows Login that is a member of the Administrators Group on the CI Computer for the CI 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 CI 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)

Install

This section describes the installation procedure for CI 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 CI. Refer to Required Software, page 5.

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 CI.

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.

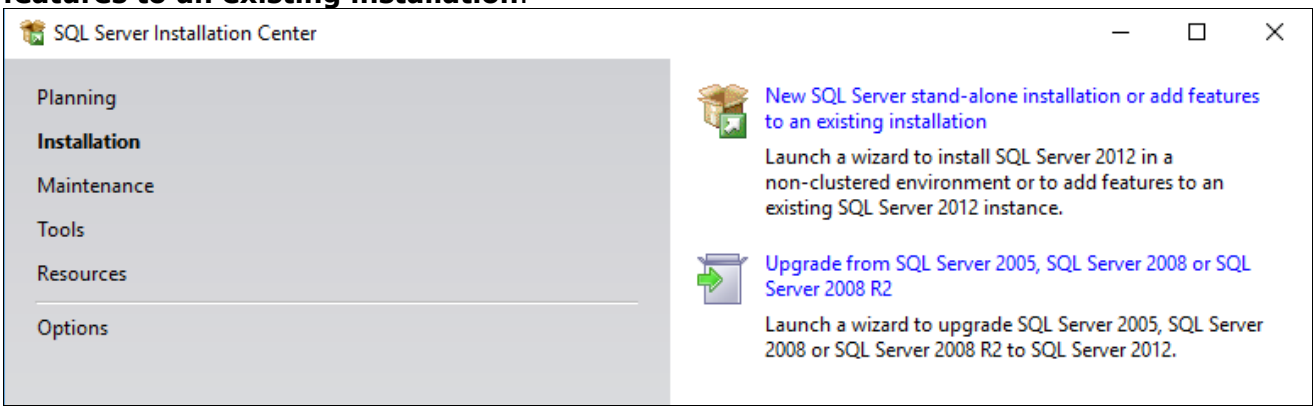
This list includes most of the files listed on the SQL Server download pages with their descriptions:

- **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.

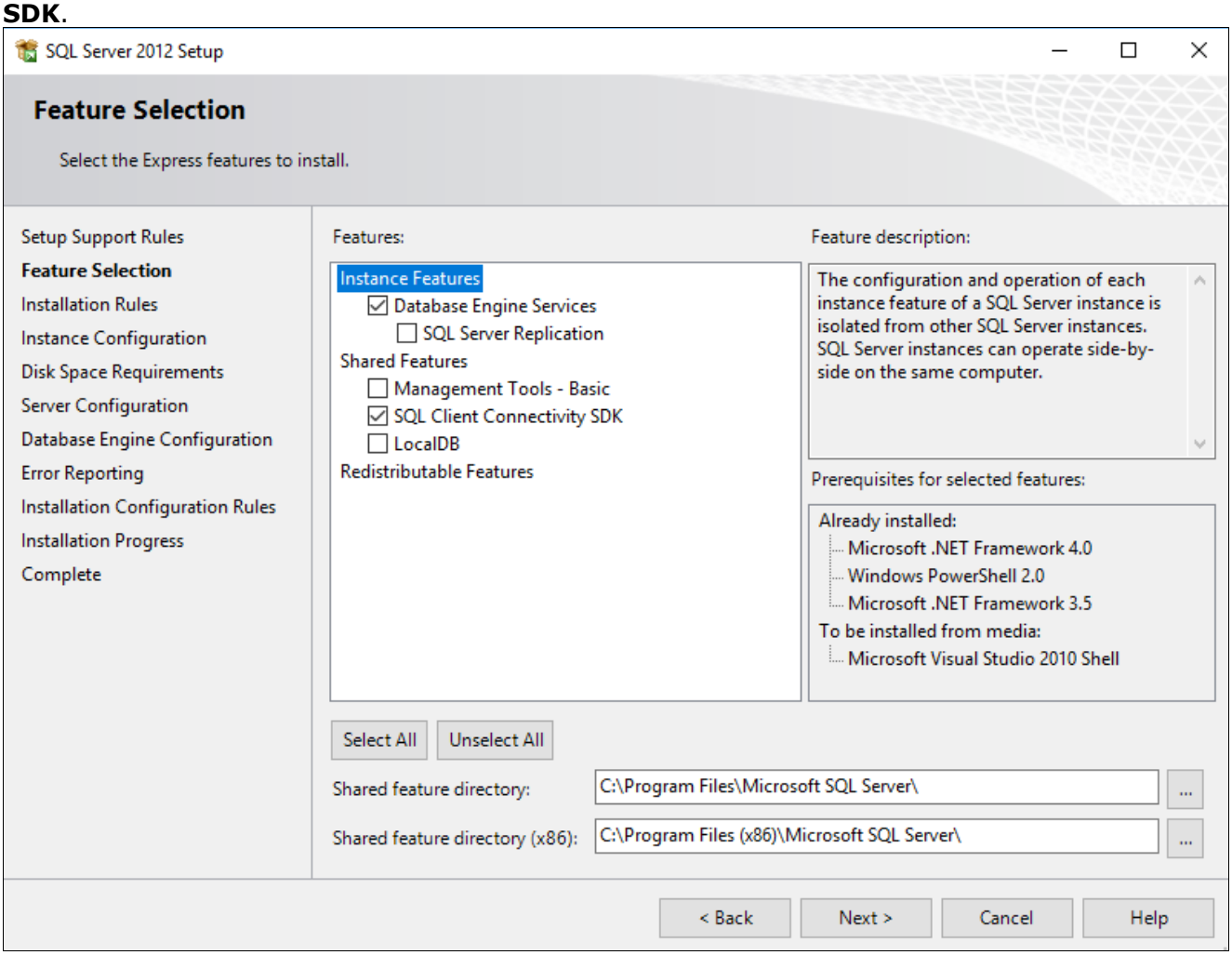
Choose the download(s) you need.

Run SQL Server Install

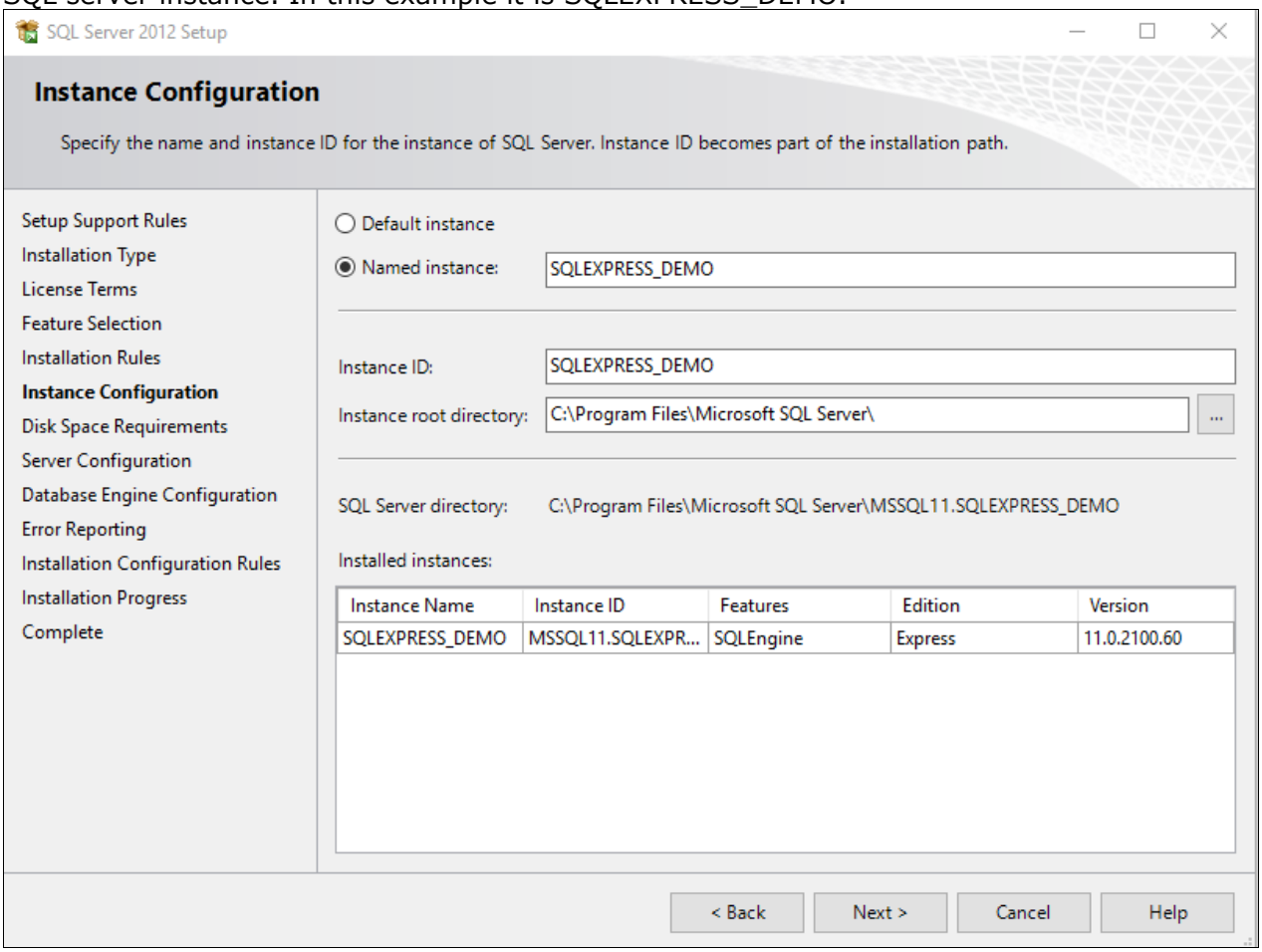
1. Run the installation file that you downloaded from the Microsoft website. The installation wizard will launch where you will choose **New SQL Server stand-alone installation or add features to an existing installation**.



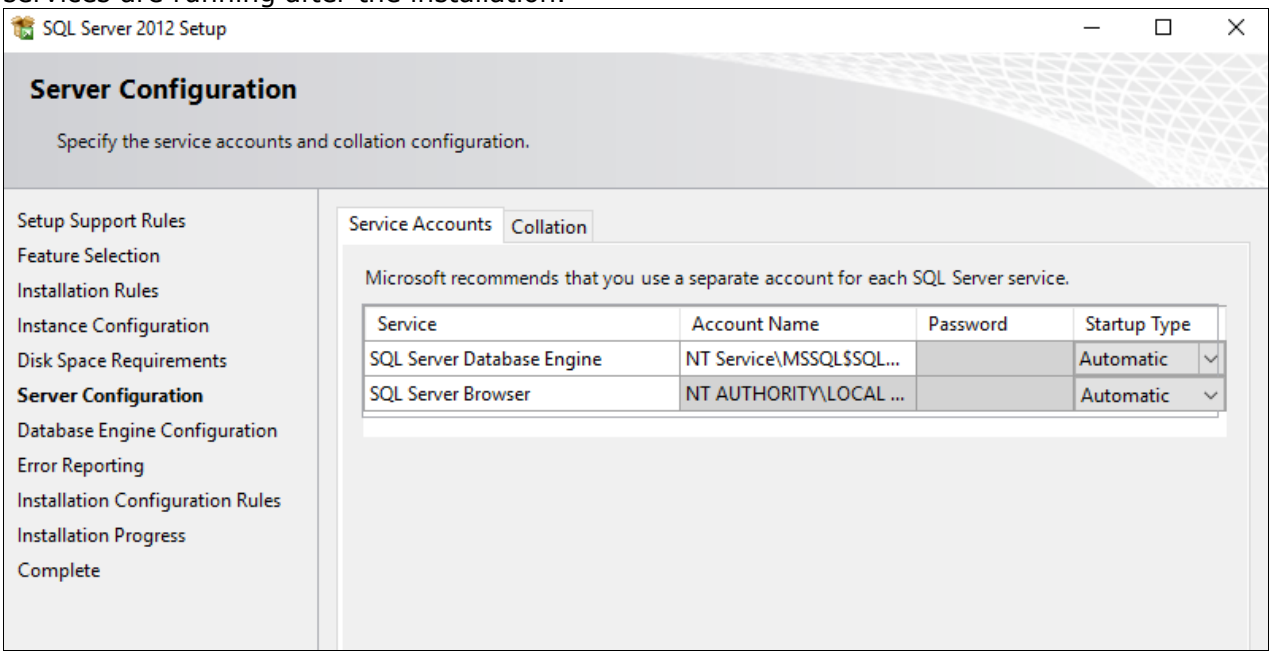
2. Accept the license terms prompt before the install begins. The application will run a system check.
   Note: The system may require a reboot when the scan is complete.
3. Make sure the checked options are **Database Engine Services** and **SQL Client Connectivity SDK**.



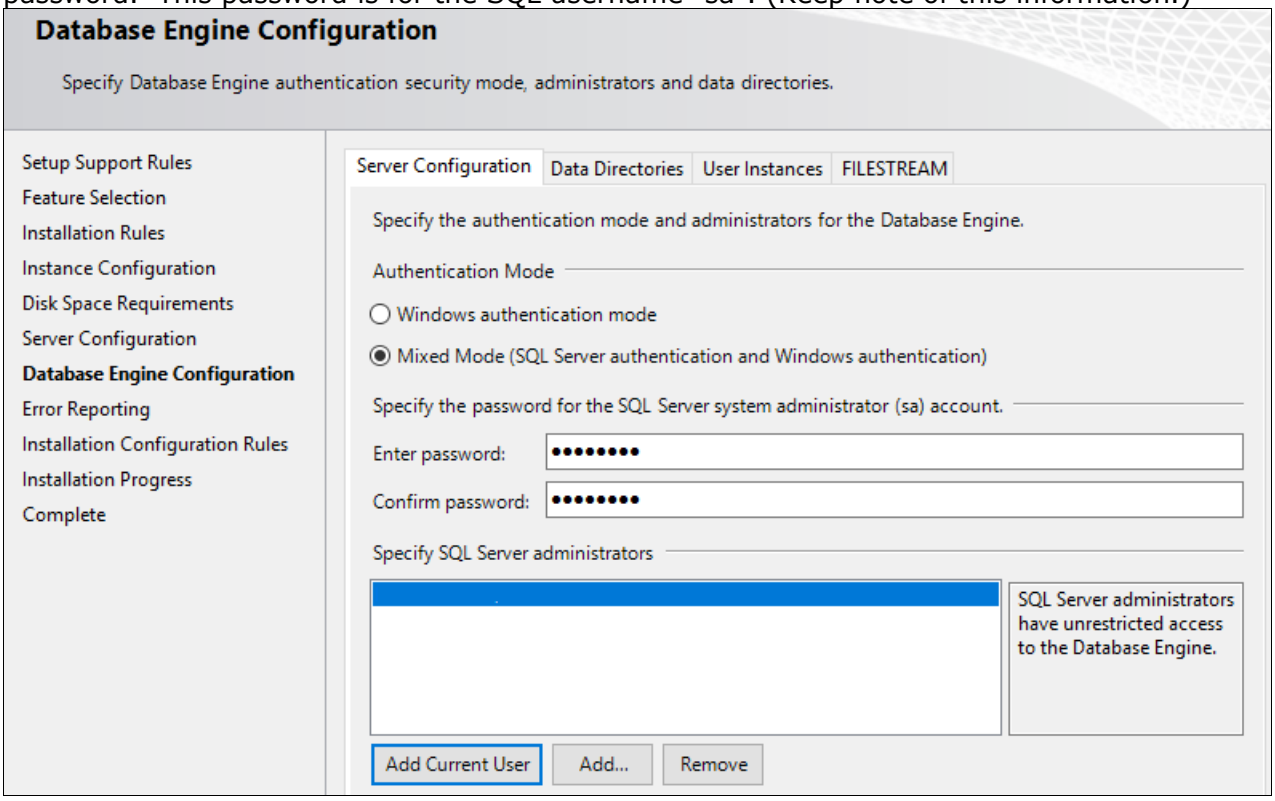
4. Check the **Named Instance** radio button and enter the name you wish use to identify your SQL server instance. In this example it is SQLEXPRESS_DEMO.



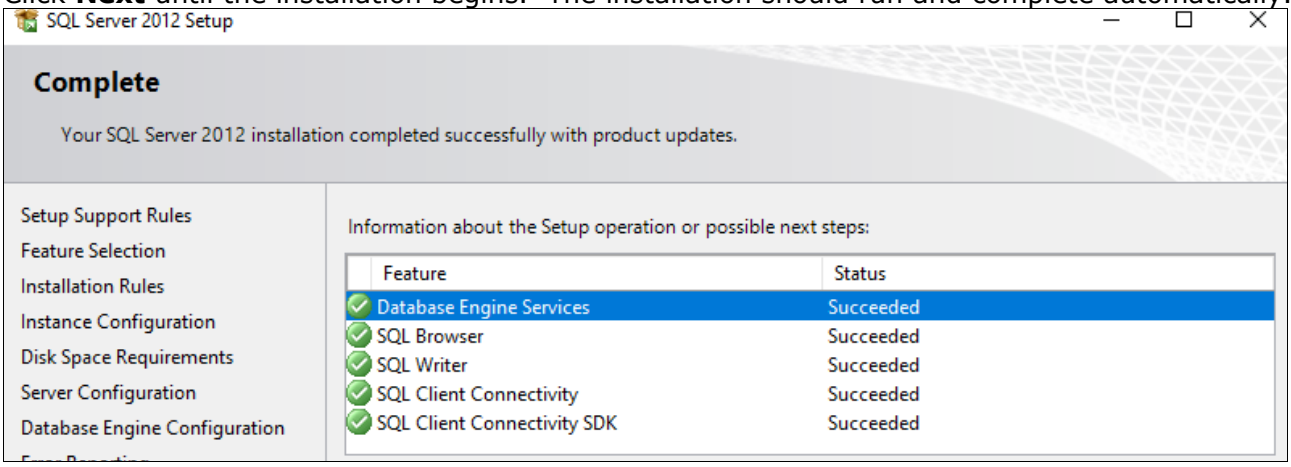
5. Confirm that both of the SQL Services are set to **Automatic** (as shown below) to ensure the services are running after the installation.



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.)



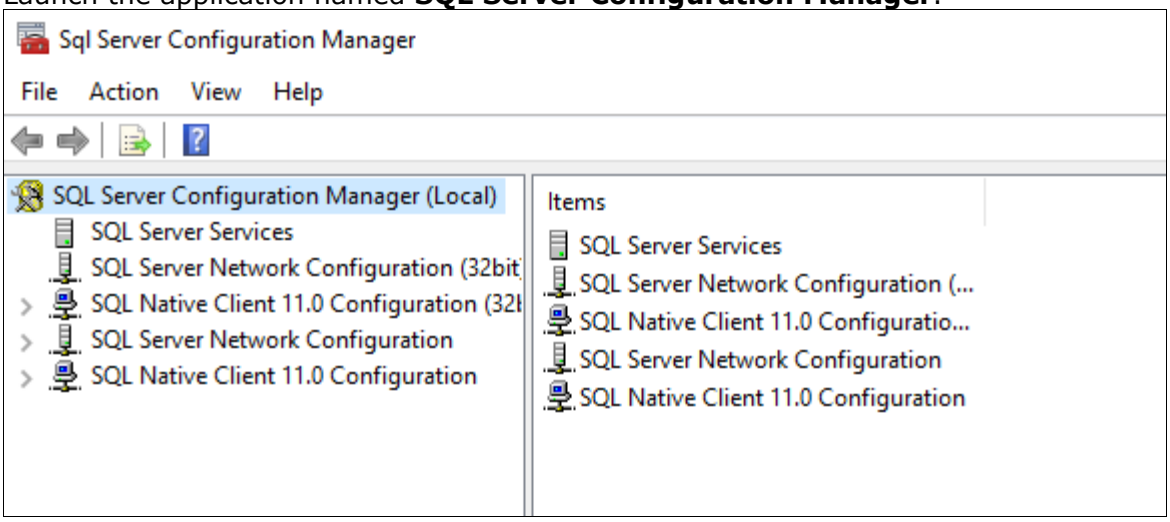
8. Click **Next** until the installation begins. The installation should run and complete automatically.



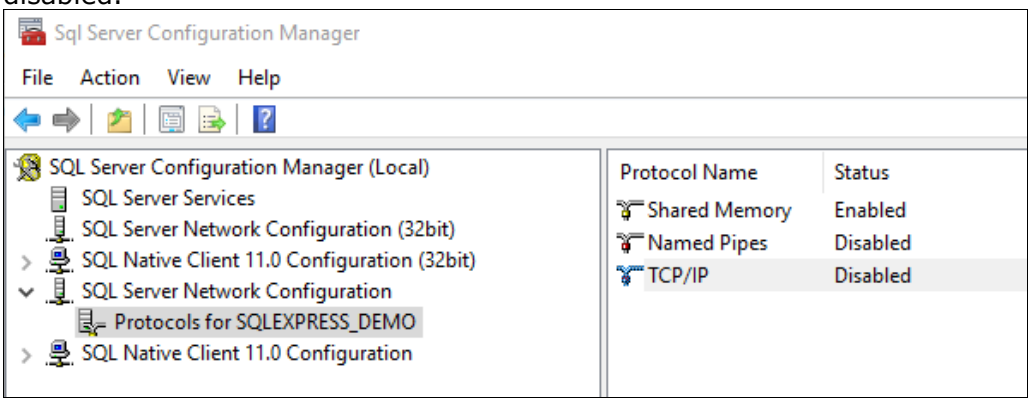
9. Once it is complete, click **Close** then confirm that the TCP/IP ports are enabled by using the following steps.

Confirm TCP/IP Ports are Enabled

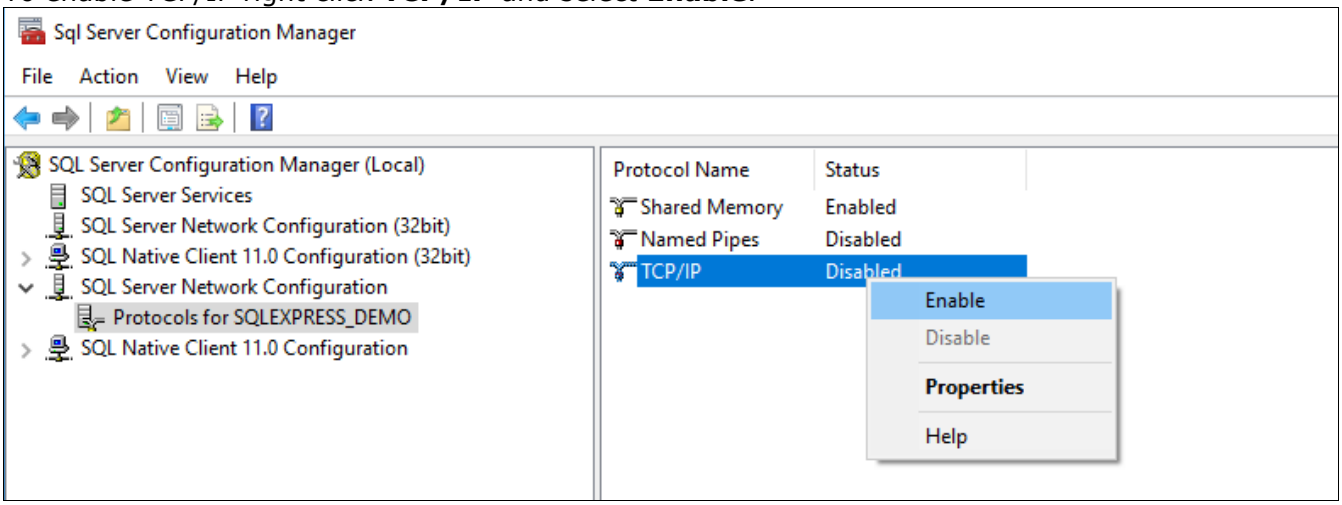
1. Launch the application named **SQL Server Configuration Manager**.



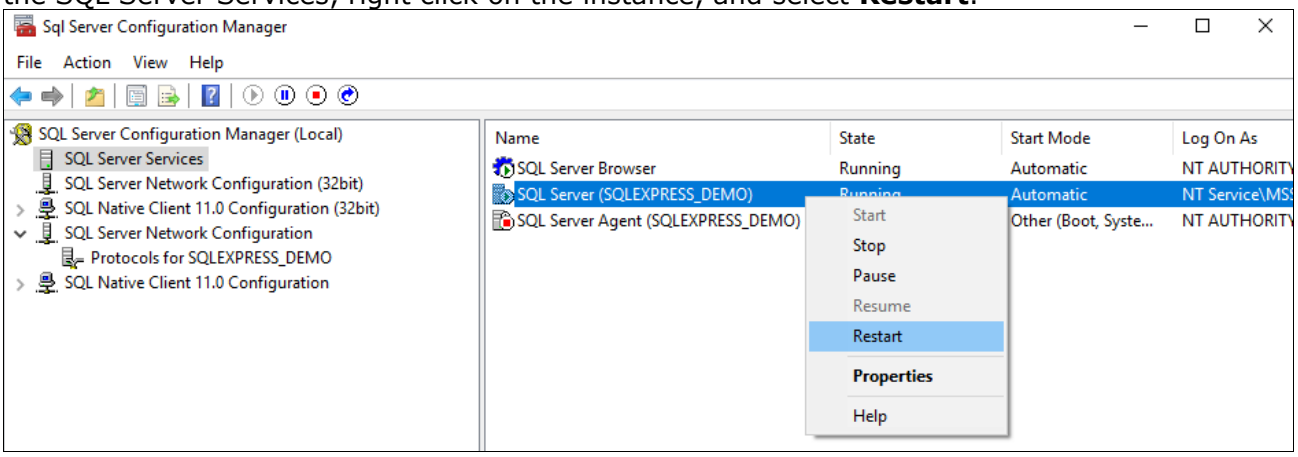
2. Highlight the protocols for the instance name which is located under the **SQL Server** Network **Configuration**. In this example it is named, **Protocols for SQLEXPRESS_DEMO** and it is disabled.



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**.



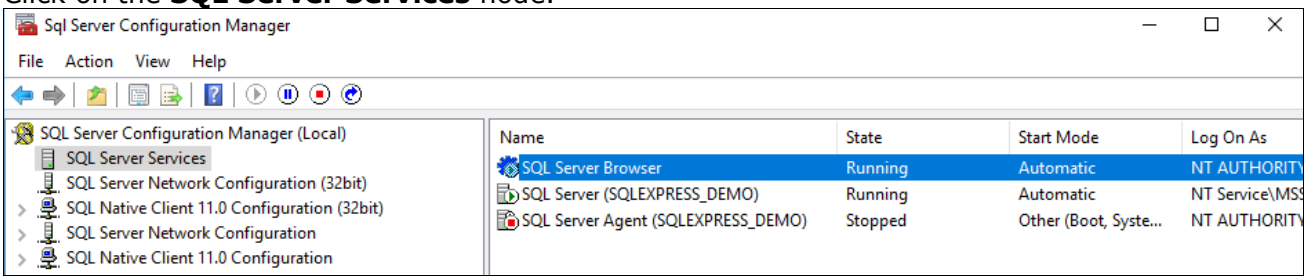
5. After it restarts, configure the SQLBrowser service to run automatically by using the following steps.

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 SP3, complete the following steps.

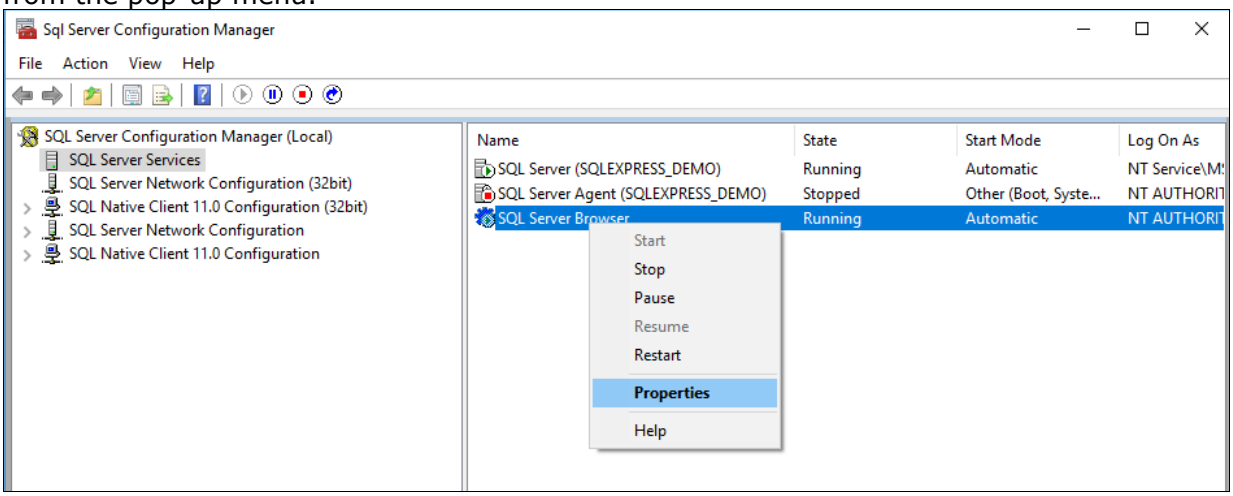
1. Open the **SQL Server Configuration Manager**. The path to this program in the Windows menu may differ slightly on different versions of Windows.

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

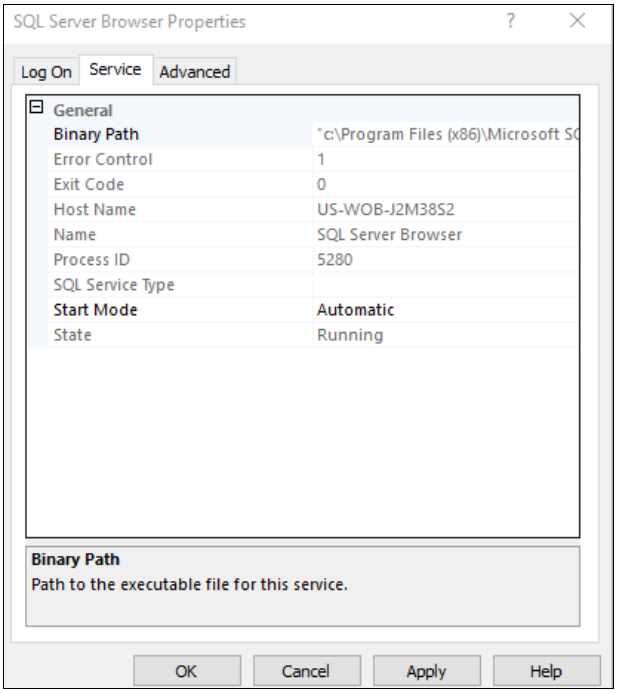


In this example, SQL SERVER Browser is set to start automatically already. If yours is not, use the following steps.

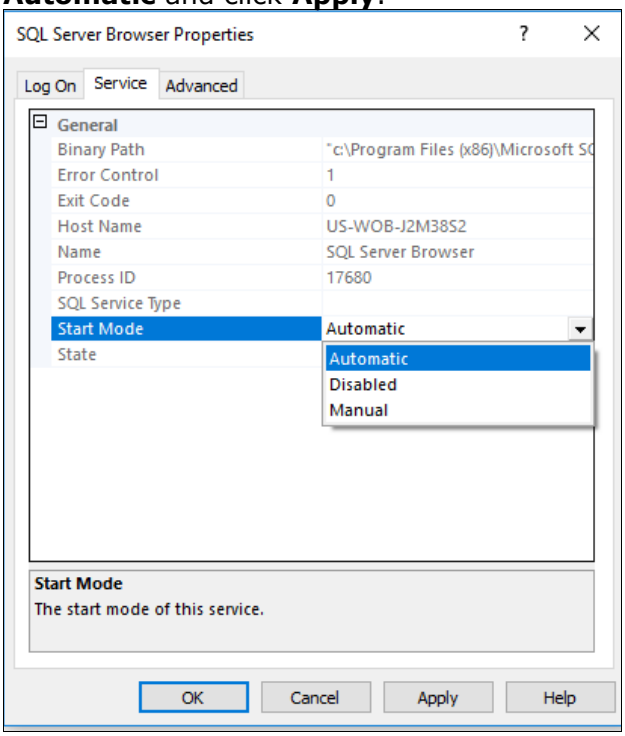
3. Right-click on the **SQL Server Browser** in the right side of the display and select **Properties** from the pop-up menu.



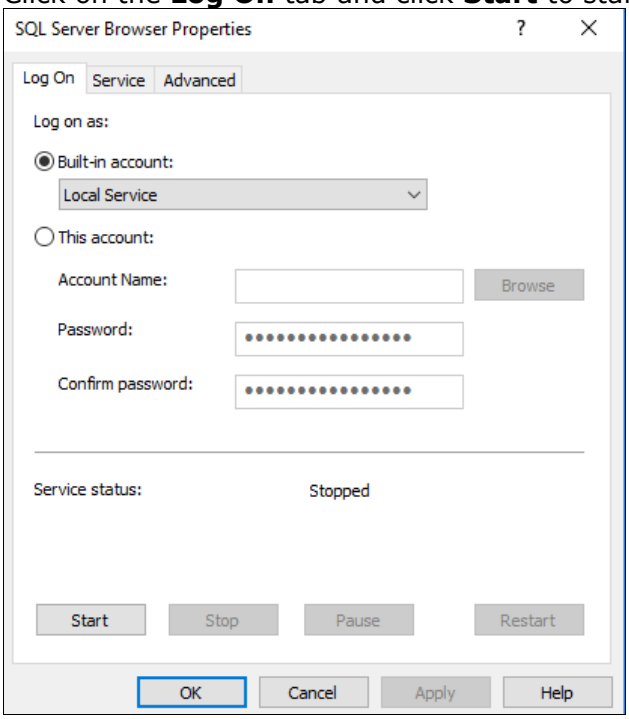
4. Click on the **Service** tab.



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.

CI Computer Install or Upgrade

Perform the following installation by first logging into the CI 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 CI, 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/generic/CI_releasenotes.pdf*.

1. Open an Internet Explorer browser window and enter the CI 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 **Othe**r 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 CI 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 CI Application on the CI 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 CI in the default location with all documentation. Click **Next**.
    c. In the **Database** screen you will see that the default name for the CI 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 CI 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 CI Application will be installed and the CI database will be created within the target Microsoft SQL Server. In **InstallShield Wizard Completed** click **Finish**.

Configure Custodial Integrator

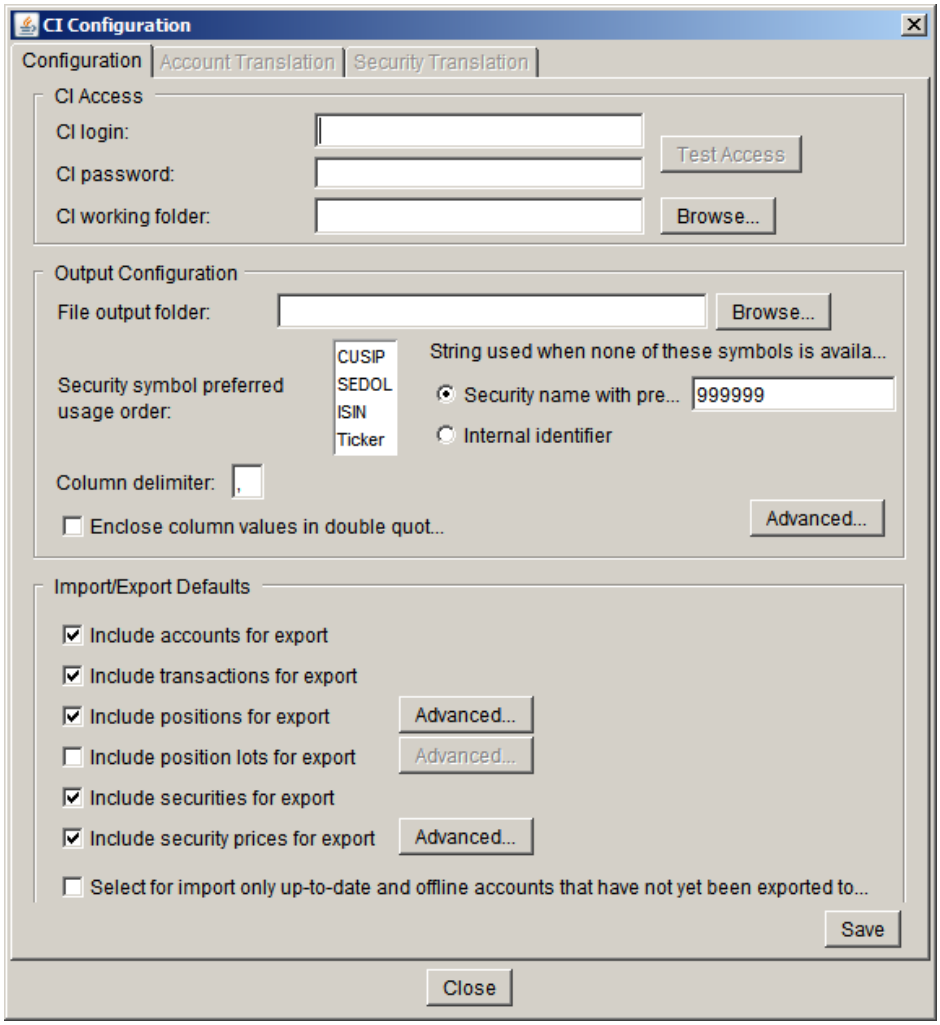
The file \Program Files\Custodial Integrator\CI.ini provided by the installation procedure represents a typical configuration with very few customizations. Determine which additional CI parameters you must use by reviewing the CI Parameters section on page 29.

1. Edit the \Program Files\Custodial Integrator\CI.ini file to add any required CI Parameters.
2. Run CI 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 25. If the problem continues to occur contact ByAllAccounts Technical Support for assistance.

After you run CI without error, the CI 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 CI additional configuration information required to access your financial information in WebPortfolio, and troubleshooting any issues you encounter in your environment.

1. Run CI by double-clicking the Desktop Shortcut.
2. The main view of CI displays, showing the steps to move data from WebPortfolio to files.

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



4. In the *CI Access* section, enter the following

   CI login: CITEST

   CI password: WEEPINGWILLOW1

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

   CI working folder:

   Create a working folder for CI 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.

5. Verify that CI can successfully use the CI 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.

6. In the *Output Configuration* section, enter the folder to which you want CI to write output files.

7. In the *Import/Export Defaults* section, the following options are available to specify which types of files should be generated 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. You may review these settings now:
    a. *Include accounts for export* – this option causes CI to create an account file in the output folder during the Import/Export cycle
    b. *Include transactions for export* – this option causes CI to create a transaction file in the output folder during the Import/Export cycle
    c. *Include positions for export* – this option causes CI to create a reconciliation file in the output folder during the Import/Export cycle
    d. *Include securities for export* – this option causes CI to create a price file in the output folder during the Import/Export cycle
    e. *Include security prices for export* – this option causes CI to create a price file in the output folder during the Import/Export cycle
    f. *Select for import only up-to-date and offline accounts that have not yet been exported today* – this controls which accounts are included in the import/export cycle. If you plan to run multiple downloads in CI each day you will want to leave this option checked.
8. Click **Save** to save your configuration settings.
9. Click the **Account Translation** tab. CI will load the test accounts and display them in the **Untranslated WebPortfolio (WP) accounts** section. Verify that at least one account appears in this section.
10. You have now confirmed that CI can contact ByAllAccounts to obtain data. Click **Close** to close the CI Configuration tab and return to CI's main view and exit CI.

Consider password encryption models

As of version 3.7, CI 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 CI access the same database, described in Setup for Multiple CIs that access the same database.

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 34.

Install multiple instances that access the same database

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

Note: If you are upgrading CI, 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/generic/CI_releasenotes.pdf*.

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

Setup for Multiple CIs that access 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 CI that access the same database. If you are using that model, you can skip the rest of this section.

Enhanced encryption model

However, because the enhanced encryption model is more complicated, it does require special instructions as described here.

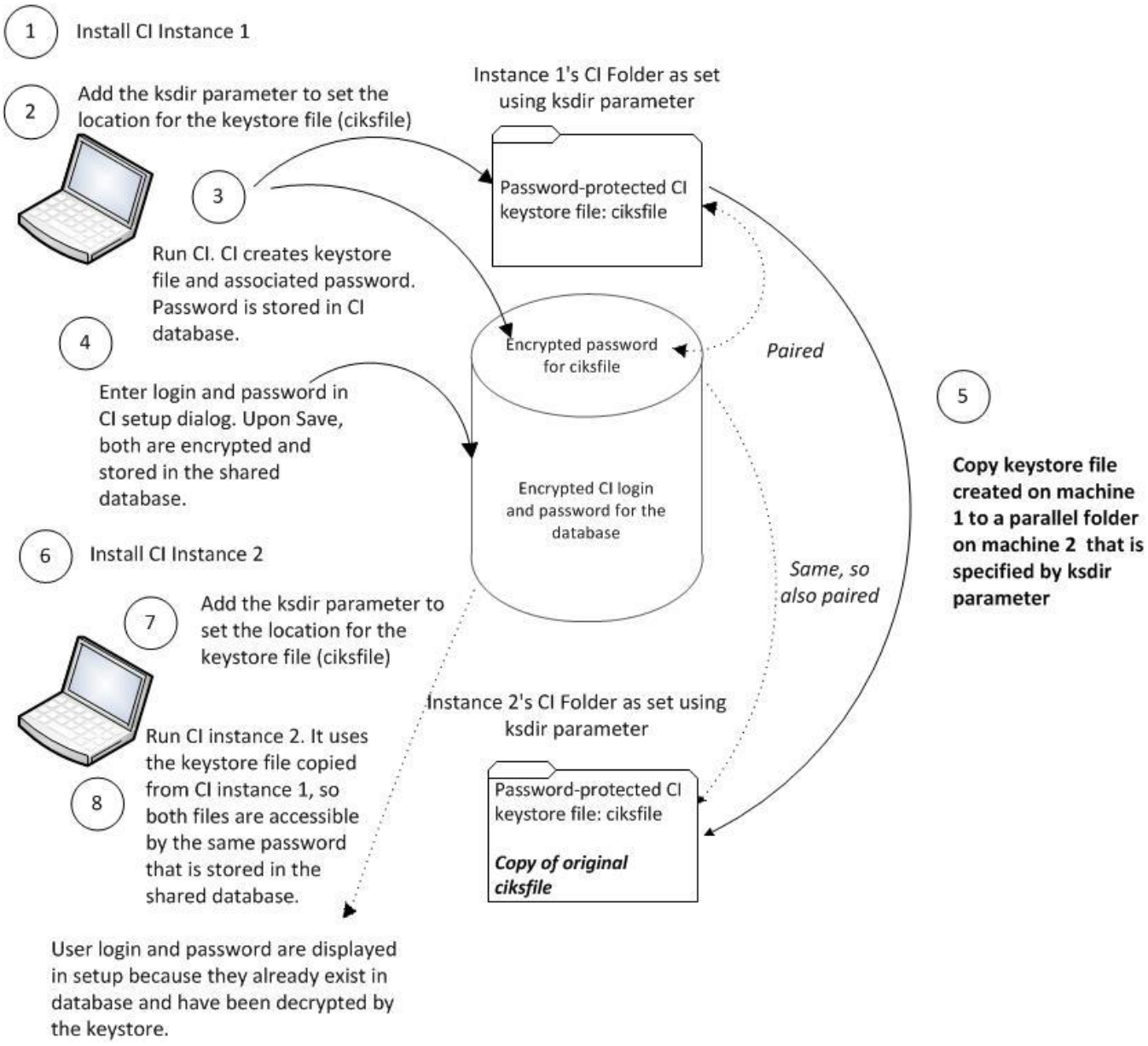
Note: If you are upgrading CI from version 3.15.001 or earlier, you may need to edit each runCI.bat instance as described in the release notes for 3.16.001. Refer to *http://www.byallaccounts.net/Manuals/Custodial_Integrator/generic/CI_releasenotes.pdf*.

The password for the keystore file is stored in the database and is uniquely paired to the keystore file, and each instance of CI 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 CI 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 CI instance 1 for the first time after install. That creates the keystore file and its associated password which is stored in the CI 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 CI setup dialog. Upon saving, the user login and password are encrypted and stored in the shared database.
5. **Before** installing CI instance 2, copy the keystore file created on machine 1 to a parallel folder on machine 2.
6. Install CI instance 2 on machine 2.
7. As you did for CI instance 1, set the location for the keystore file using the ksdir parameter.
8. Run CI instance 2. CI instance uses the keystore file that was copied from CI 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)
Instance 1's CI Folder as set using ksdir parameter
3
Password-protected CI keystore file: ciksfile
Run CI. CI creates keystore file and associated password. Password is stored in CI database.
4
Encrypted password for ciksfile
Paired
5
Copy keystore file created on machine 1 to a parallel folder on machine 2 that is specified by ksdir parameter
Enter login and password in CI setup dialog. Upon Save, both are encrypted and stored in the shared database.
Encrypted CI login and password for the database
6
Install CI Instance 2
Same, so also paired
7
Add the ksdir parameter to set the location for the keystore file (ciksfile)
Instance 2's CI Folder as set using ksdir parameter
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.
8
Password-protected CI keystore file: ciksfile
Copy of original ciksfile
User login and password are displayed in setup because they already exist in database and have been decrypted by the keystore.

Install multiple instances of CI on same machine with different databases

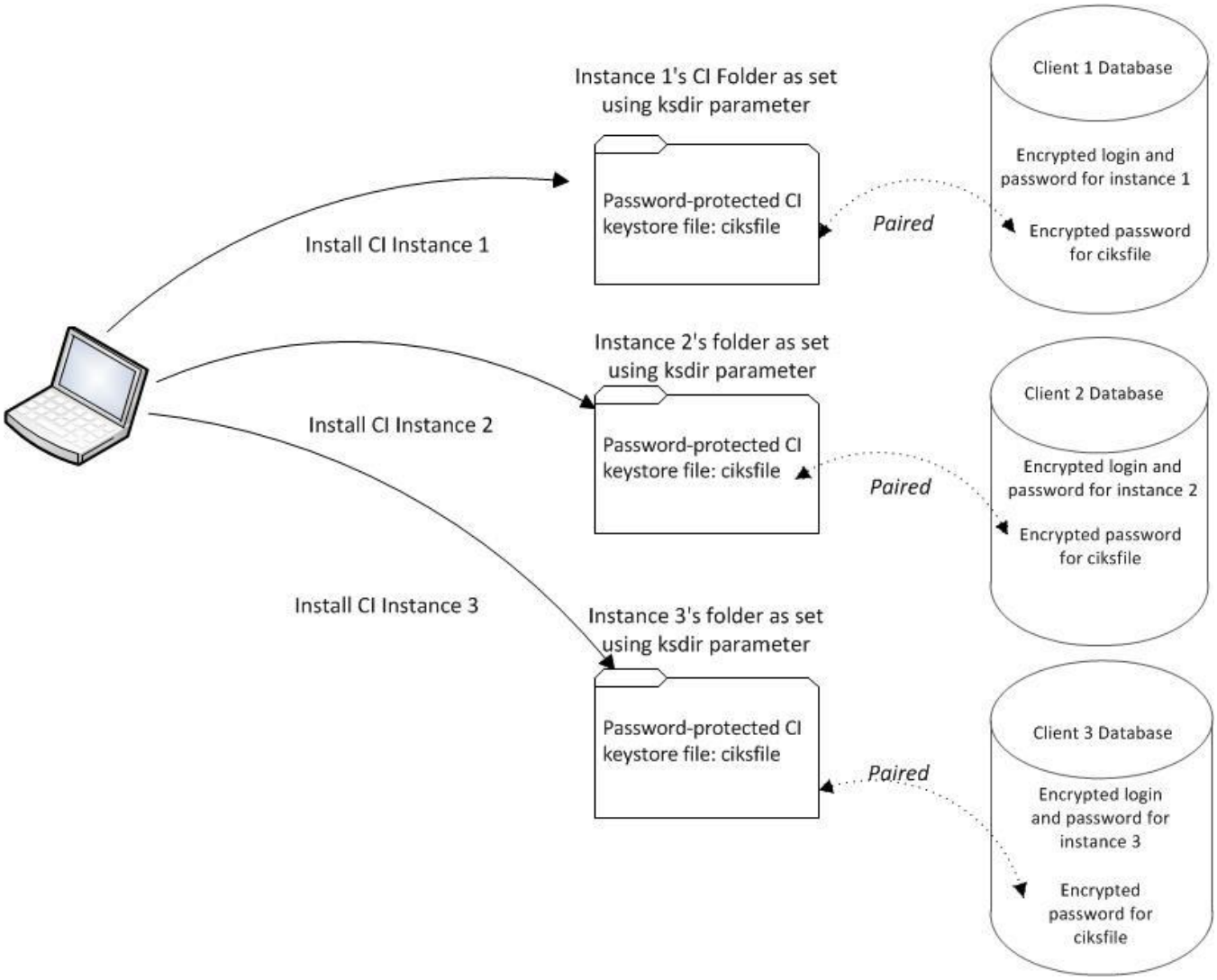
This section describes how to install multiple instances of CI on the same machine that access different databases. The instructions vary a bit depending on whether you use the default or enhanced encryption model for the logins and passwords. For information about the encryption methods, refer to Consider password encryption models, page 21.

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

Use the following procedure to get all CI 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 CI 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/generic/CI_releasenotes.pdf*
2. When setting up CI for each client also configure a separate location for their CI working and output folders.
3. Run CI for one of the customers.
4. Go to the **Setup > Configuration** tab. If this is a CI upgrade from a version prior to 3.5.002, you will be prompted to re-enter the username and password.
5. Make sure the CI working folder is set to the correct folder for the instance.
6. Repeat steps 3- 5 for the additional CI 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: ciksfile) in the that folder for each customer, and should allow each instance of CI 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.



Troubleshoot Install Problems

General Techniques

CI logs information about its activities to a log file. If CI encounters an error, the error usually appears with full detail in the log file. CI log files can be found in the log subfolder of your CI Working folder. The CI working folder is specified by you in the CI Setup area, Configuration tab. If CI cannot contact the database and is therefore unable to determine the working folder, it will store log files in the Windows temp area for the current windows user. To obtain the CI log file associated with the most recent CI session, sort the CI log files by date in the Windows Explorer. CI 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 CI to provide additional information for troubleshooting. Use the **debugall=**1 parameter in the CI.ini file to direct CI to log more detailed information about what it is doing. This parameter is described in the **CI Parameters** section on page 29.

Known Issues

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

General Errors

Symptom	Possible Cause	Action
Configuration **Test Access** reports 'Error loading WebPortfolio Data'	Incorrect CI proxy configuration for Internet access	Check the CI log file for more details. Contact ByAllAccounts Technical Support for assistance.
CryptoException error	Password encryption problem. CI database has a stored keystore password in the database and a keystore file exists in the ksdir, but they do not match.	1. Close CI. 2. Delete the ciksfile in the ksdir folder 3. Restart CI 4. Go to Setup and reenter your user id and password (you may be prompted to do that as soon as CI 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 CI 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 CI instance that was just recovered to each ksdir folder (one per instance) before running these other CI instances. See Consider password encryption models As of version 3.7, CI 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 CI access the same database, described in Setup for Multiple CIs that access the same database. **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 34. Install multiple instances that access the same database on page 21 for more information.

Database Errors

In the case of a database error, CI 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 CI 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.

CUSTODIAL INTEGRATOR CONFIGURATION REFERENCE

CI 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 proxy host as "proxy.myserver" and your SQL Server password as "mypassword":

proxyhost=proxy.myserver

sqlpw=mypassword

CI Customizations

These parameters customize the behavior of your CI installation.

Parameter label	Parameter value(s)	Default	Description	Example
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=useMV

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.	For example, if CI startup directory is C:\CI, and the custom translations folder is C:CI\data\customtranslations, then this parameter would be set to: inputFolder=C:\CI\data\customtranslations
defaultAccountIdentifier				
	WPAccountNumber WPInternalID WPAccountName	WPAccountNumber	Used to set the default account identifier for untranslated accounts to one of three values: ▪ WPAccountNumber sets it to the Account Number from WebPortfolio (default). ▪ 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. Setting this to either WPAccountNumber or WPAccountName will enable a CI feature that in the case of duplicates offers the user the option to allow CI to append the internal account identifier to make any new identifiers unique. Refer to the *Custodial Integrator User Guide for* more. Note that maximum account identifier length may be adjusted using the `AccountIdentifierMaxLen` parameter.	defaultAccountIdentifier=WPInternalID
AccountIdentifierMaxLen				
	19-128	128	Specifies the maximum number of characters for the CI account identifier. The specified max length cannot be less than 19.	AccountIdentifierMaxLen=101

			Applies to both CI GUI and CI Autorun.	
translateFeedEarliestTxDate				
	Valid date in format: YYYYmmDD	(Today's date)	Specifies the transaction date to be used for the initial import data for accounts at FIs designated as "feeds". When used for CI interactive, this parameter affects the position export for failed accounts at FIs designated as "feeds". Positions will NOT be exported for feed accounts that failed aggregation. The setting for failed accounts in the "advanced" positions-export dialog will not be used in this case, but will still apply to non-feed accounts and the setting for Stale accounts will still apply to both feed and non-feed accounts. Applies to both CI GUI and CI Autorun.	translateFeedEarliestTxDate=20201209
identifierAllowedCharacters				
	(special characters) none all	all	Specifies which special (non-alphanumeric) characters are allowed in the account identifier when requesting automatic account translations. Can be set to allow specified non-alphanumeric characters, none, or all. Applies to both CI interactive (GUI) and CI Autorun. When used in interactive mode, removal of non-allowed characters is enforced only when multiple accounts are selected in bulk for automatic translation. Any characters are allowed when the user manually types the identifier for a single account. Can be set to: ▪ **none** - only alphanumeric characters will be kept in	identifierAllowedCharacters=none

			the account identifier. ▪ **all** - when set to all or the parameter is not present, then all special characters are allowed in the account identifier. ▪ One or more of the following characters with no separators: ._\\/+?!;:,%!@*()\"{}^|<>#$=&'`~; - See notes below. **Notes for special characters:** Can be specified in .bat scripts (such as runCI.bat or runciauto.bat), but we recommend specifying it in CI.ini for more flexibility setting special characters. The character list cannot contain square brackets (neither [nor]). If specified in CI.ini, the list of characters should be entered with no surrounding double quotes, including the case when the list contains or starts with a space. A double quote can be specified as a special character in the CI.ini file. Example for CI.ini: `identifierAllowedCharacters=._ \%;"-/+!` Conversely, for.bat scripts (for example runCI.bat or runciauto.bat), the list of characters must be contained within double quotes, and a double quote (") cannot be specified as an allowed character. To specify a back slash (\) or percent (%) in a .bat file it must be preceded by another incident of itself (**** or %%) to take effect. Example of runCI.bat: `identifierAllowedCharacters="._ \\%%;-/+!"` Applies to both CI GUI and CI Autorun.	

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

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 21. For installing or upgrading multiple instances of CI on same machine with different databases, see page 24.	ksdir=<full path of keystore folder>
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 21.	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 21.	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