

I D E R A

SQL Admin Toolset

Version 1.9.5



Table of Contents

Simplify day-to-day SQL administration tasks 1

SQL Admin Toolset Release Notes 2

New features and fixed issues 3

Previous features and fixed issues 11

Known issues..... 16

Welcome to SQL Admin Toolset 17

What is IDERA SQL Admin Toolset? 18

How to use the SQL Admin Toolset Launchpad..... 21

Get started 22

Product requirements 23

How to install IDERA SQL Admin Toolset 25

Upgrade to this release 26

Configure your deployment 27

Diagnostic tools..... 33

Backup Status..... 33

Connection Check 33

Object Search 33

Password Checker 33

SQL Discovery..... 33

SQL Search..... 33

Backup Status..... 34

SQL Search..... 36

Object Search 38

SQL Discovery..... 40

Password Checker 43

Connection Check 46

Administration tools..... 48

Connection String 48

Database configuration..... 48

Database Mover 48

Job Editor	48
Job Mover.....	48
Linked Server Copy	48
Login Copy	48
Multi Query.....	48
Partition Generator	48
Quick Reindex.....	49
Server Configuration	49
Table Pin	49
User Clone	49
Multi Query.....	50
Quick Reindex.....	52
Database Mover	53
Server Configuration	55
Login Copy	58
Database Configuration	59
User Clone	61
Partition Generator	63
Job Mover.....	65
Connection String Generator	66
Job Editor	69
Linked Server Copy.....	71
Table Pin	73
Analytic tools	75
Index Analyzer.....	75
Inventory Reporter	75
Patch Analyzer.....	75
Server Ping.....	75
Server Statistics.....	75
Space Analyzer	75
Inventory Reporter	76

Patch Analyzer	78
Index Analyzer	80
Space Analyzer	84
Server Statistics	85
Server Ping.....	86
SQL Admin Toolset PDF	88

Simplify day-to-day SQL administration tasks

- 24 essential desktop tools for your everyday SQL Server needs
- Run every day diagnostics across multiple SQL Servers
- Complete administrative tasks in minutes, not hours
- Perform analysis on your SQL data

SQL Admin Toolset Release Notes

IDERA SQL Admin Toolset includes 24 essential desktop tools for monitoring, troubleshooting, administering and reporting on your SQL Servers. These tools help reduce administrative tasks from hours to minutes and eliminate the time spent developing, testing and maintaining custom scripts.

To get a quick glimpse into the newest features, fixed issues, and known issues in this release of IDERA SQL Admin Toolset, review the following sections of the Release Notes:

- [Learn about key new features in this release](#)
- [Review issues fixed by this release](#)
- [Review previous features and fixed issues](#)
- [See known issues](#)

New features and fixed issues

IDERA SQL Admin Toolset provides the following new features and fixed issues.

1.9.5 New features

Quick Reindex allows designation of SORT_IN_TEMPDB

Users now can designate the SORT_IN_TEMPDB option when using the Quick Reindex tool. For more information about this tool, see [Quick Reindex](#).

Space Analyzer allows users to disable WMI credential requirement

Space Analyzer users now can check or clear the **Use WMI Credentials checkbox** to toggle whether to require WMI credentials when running an analysis. For more information about this tool, see [Space Analyzer](#).

19.5 Fixed Issues

Issue Number	Description
SQLADMI-1803	User Clone. Corrects an issue causing some users to receive the error message, "No user information found" when attempting to use the tool.
SQLADMI-1801	Index Analyzer & Quick Reindex. Corrected an issue preventing spatial indexes from appearing.
SQLADMI-1799	Backup Status. Corrected an issue preventing the backup file size from updating upon a second backup.
SQLADMI-1791	Corrected an issue preventing server groups from import if the servers are grouped under Microsoft SQL Servers in SSMS.
SQLADMI-1776	Backup Status. Corrected an issue causing tempdb to not appear in the Backup Status report after clearing the Skip System Database option.
SQLADMI-1738	Patch Analyzer. Corrected an issue causing SQL Server 2012 updates to appear as new updates.

1.9.4 New features

Supports SQL Server 2019

The tools available in the IDERA SQL Admin Toolset now support SQL Server 2019. For more information about minimum requirements for IDERA SQL Admin Toolset, see [Product requirements](#).

Supports Microsoft Windows Server 2019

IDERA SQL Admin Toolset now supports Microsoft Windows Server 2019. For more information about minimum requirements for IDERA SQL Admin Toolset, see [Product requirements](#).

Backup Status Recovery Mode

The Backup Status tool now includes a column that displays the recovery mode type for each displayed database. The **Recovery Mode** column displays **Simple** or **Full**. For more information about Backup Status, see [Backup Status](#).

Database Mover Windows Cluster

The Database Mover tool now supports the destination SQL Server instance on a Microsoft Windows cluster with cluster-shared volumes (CSV). The drive must be a member of the cluster group in order to have the files attached to the database and the user performing the move must have all permissions to perform database operations. For more information about Database Mover, see [Database Mover](#).

User Clone supports multiple users

The User Clone tool now supports cloning of multiple users. For more information, see [User Clone](#).

1.9.4 Fixed issues

Issue Number	Description
SQLADMI-1783	Test Cloud support for current Azure and AWS EC2
SQLADMI-1781	Adds support for the Windows Server 2019 platform.
SQLADMI-1780	Adds support for the SQL Server 2019 platform.
SQLADMI-1777, SQLADMI-1778	Quick Reindex and Index Analyzer . Corrects an issue causing an exception when the user clicks Browse in the database list.
SQLADMI-1776	Corrects an issue preventing tempdb from displaying in the backup status.
SQLADMI-1775	Patch Analyzer . Corrects an issue that caused some users to see an error message after clicking Check for New List .
SQLADMI-1772	Updated the copyright info in the tools and Launchpad.
SQLADMI-1770	Database Mover . Now supports a clustered shared volume disk.

Issue Number	Description
SQLADMI-1768	Installer no longer fails as a result of requiring .NET 2.0 or higher message.
SQLADMI-1761	Login Copy. Now includes permissions when copying server login detail.
SQLADMI-1759	Separates installer into Trial and Production versions.
SQLADMI-1757	Database Mover. Corrects an issue preventing some properties from being copied or moved to target.
SQLADMI-1755	ATS Tools now are organized into the Tools folder on Microsoft Windows 10.
SQLADMI-1748	Job Mover. Corrects an issue that prevented copying of job step properties to the destination.
SQLADMI-1741	Backup Status. Now presents the Recovery Mode for Backup Status.
SQLADMI-1582	Launchpad. Corrects an issue causing the following error message: Launchpad error: The source was not found. Inaccessible logs: security.
SQLADMI-45	User Clone. Now allows scripting of multiple users.

1.9.3 New features

There are no new features in this release.

1.9.3 Fixed issues

Issue Number	Description
SQLADMI-1764	Corrects an issue with Backup status for SQL Server 2012 and older.
SQLADMI-1762	Users with a large number of instances no longer experience a delay in results running Patch Analyzer in serial rather than parallel.

1.9.2 New features

General

Supports SQL Server 2017

The tools available in the IDERA SQL Admin Toolset now support SQL Server 2017. For more information about minimum requirements for IDERA SQL Admin Toolset, see [Product requirements](#).

New licensing

This product requires Microsoft Visual C++ 2015 Update 3 Redistributable. If the installer does not detect the correct redistributable in your environment, it will attempt to install or update to the correct version.

Please refer to the following Microsoft Support document for information about VC++ 2015 redistributable prerequisites: <https://support.microsoft.com/en-us/help/3118401/update-for-universal-c-runtime-in-windows>.

Diagnostic Tools

Backup Status tool now provides timestamps for backups on Availability Group nodes

The Backup Status tool now provides timestamps for Copy-Only backups performed on Availability Group nodes. For more information about this tool, see [Backup Status](#).

Administration Tools

Database Mover now allows moves from higher to lower versions and editions

The Database Mover tool allows all moves including moves from higher versions to lower versions and higher editions to lower editions. Note that some features available in the higher versions/editions may not be available in the lower versions/editions. Users receive a warning message before moving from the higher to lower version/edition, and must confirm the move. For more information about this tool, see [Database Mover](#).

Improves Multi-Query navigation

The Multi-Query tool now includes a vertical scroll bar to quickly search through results.

Allows database searches in Multi-Query tool

Users now can search targeted databases in the Multi-Query tool using a wildcard search. The Add Query Target dialog includes fields that allow a user to enter a search term or expression to filter for a list of matching databases. For more information about this feature, see [Multi Query](#).

Includes new configuration options in Job Editor

The Job Editor tool now allows users to set the time since last run alert. Users can set the days, hours, or minutes in the **Jobs not run in the last** field. For more information about this feature, see [Job Editor](#).

Enable or disable copied jobs when using Job Mover

While using Job Mover, the user can select whether the jobs on the destination server are enabled or disabled. For more information about this feature, see [Job Mover](#).

Job Mover updates job steps to point to the new server for SSIS packages

The Job Mover tool now updates the job steps for SSIS packages to point to the new server rather than the old server. For more information about this tool, see [Job Mover](#).

Partition Generator now supports datetime2 datatype

The Partition Generator tool now supports the **datetime2** data type. For more information about this tool, see [Partition Generator](#).

1.9.2 Fixed issues

Issue Number	Description
SQLADMI-1728	Partition Generator evaluates partitions on all editions of SQL Server 2016 SP1 and above. SQL Server 2016 RTM is not supported because partition evaluation is not allowed on that version.
SQLADMI-1721	Backup Status report now properly sorts by Date DB Created column when Only include full database backups option is selected.
SQLADMI-1712	Includes more descriptive information in the Level column of the Patch Analyzer.
SQLADMI-1708	SQL Search now properly displays the last line of search results.
SQLADMI-1707	Quick Reindex tool no longer reloads all indexes when performing a rebuild or reorg defrag on one or more indexes.
SQLADMI-1705	SQL Index Analyzer now displays the list of columns in the correct order.
SQLADMI-1692	Updated the banner image in the printed reports.
SQLADMI-1691	Removed the WebHelp folder as it is no longer in use.

Issue Number	Description
SQLADMI-1608	Reserved1 no longer displays as a FILESTREAM database File Type in the Space Analyzer.
SQLADMI-407	The User Clone tool now properly prevents access to the source users list when connection credentials are changed.
SQLADMI-399	The Index Analyzer tool now properly supports long server names.
SQLADMI-376	Space Analyzer now displays Auto Growth and Maximum Potential Size units in correct measurements.
SQLADMI-1	Removed the 65535 character limit in Multi Query.

1.9.1 New features

There are no new features in this release. Please see the Fixed Issues section for information about what is included in IDERA SQL Admin Toolset 1.9.1.

1.9.1 Fixed issues

Resolved license failure issue

IDERA SQL Admin Toolset resolves an issue occurring with some licenses that prevented users from registering a license key on certain Windows operating systems. This issue mainly affected Windows 10 and Windows 8/Windows SQL Server 2012 users, however was also seen on other versions.

1.9 New features

Improved IDERA SQL Admin Toolset user interface

IDERA SQL Admin Toolset features a new look and feel. These changes include new colors to reflect updated IDERA branding, installer changes for fast installation, and navigation menu updates for a more user-friendly experience.

Supports SQL Server 2016

The tools available in the IDERA SQL Admin Toolset now support SQL Server 2016. For more information about minimum requirements for IDERA SQL Admin Toolset, see [Product requirements](#).

New options for the Backup Status tool

The Backup Status tool now allows you to skip offline databases in your backups by checking the **Exclude Offline Databases** checkbox. For more information about this tool, see [Backup Status](#).

1.9 Fixed issues

General

Issue Number	Description
SQLADMI-1589	Removed SQL Server 7 from all selection lists within IDERA SQL Admin Toolset as it is not a supported version.
SQLADMI-1588	Fixes an issue preventing users from activating a license on a VM with multiple NICs and constantly changing machines codes.
SQLADMI-1587	Fixes an issue preventing new installations of IDERA SQL Admin Toolset from importing registered SQL Server instances on SQL Server 2014.
SQLADMI-1582	Fixes an issue with IDERA SQL Admin Toolset 1.8.2 causing some users to receive a Launchpad error when attempting to open the product.

Diagnostics tools

Issue Number	Description
SQLADMI-1594	CONNECTION CHECK: Fixes an issue causing the Test Network connections check of the Connection Check tool to look for only the default SQL Server instance name even when the actual instance name is provided in the SQL Server field.

Administration tools

Issue Number	Description
SQLADMI-1603	DATABASE MOVER: Database Mover now includes a version compatibility check before starting to move data.
SQLADMI-1602	USER CLONE: Fixes an issue causing User Clone to display the message, "You cannot keep the original password when downgrading from SQL 2014, do you want to proceed with a blank password," message even when both instances are SQL Server 2014.
SQLADMI-12	JOB EDITOR: Job Editor now saves column setting changes to the Last Outcome column, such as column location and width.

Issue Number	Description
SQLADMI-350	DATABASE CONFIGURATION: Fixes an issue preventing the database option READ_COMMITTED_SNAPSHOT from being displayed in the Comparison Report available in the Database Configuration tool.

Analytics tools

Issue Number	Description
SQLADMI-370	INVENTORY REPORTER: Fixes an issue causing Inventory Reporter to display error messages regarding data being in the incorrect format.
SQLADMI-1605	PATCH ANALYZER: Fixes an issue causing Patch Analyzer to display SQL Server 2014 CTP1 build 11.09120 in the SQL Server 2012 section.
SQLADMI-1585	PATCH ANALYZER: Fixes an issue causing the Patch Analyzer build list to display UNKNOWN for SQL Server 2014 and SQL Server 2016.
SQLADMI-1580	PATCH ANALYZER: Fixes an issue causing Patch Analyzer to display incorrect details for SQL Server 12.0.2560.
SQLADMI-1600	SERVER STATISTICS: Fixes an issue causing Server Statistics to not display data files for SQL Server 2012 and SQL Server 2014 instances.

Previous features and fixed issues

This build of IDERA SQL Admin Toolset includes many fixed issues, including the following previously-released updates.

1.8 New features

SQL Server 2014 support

The tools available in the IDERA SQL Admin Toolset now support SQL Server 2014.

1.8 Fixed issues

There are no fixed issues in this release.

1.7 New features

There are no new features in this release.

1.7 Fixed issues

Backup Status tool

This release corrects an issue causing the Backup Status tool to unnecessarily increase memory consumption.

SQL Search tool

This release corrects an issue causing the SQL Search tool to include excerpts that did not contain the search string.

SQL Discovery tool

- The SQL Discovery tool now displays only a single error message when a user scans a list of computers and one or more of the servers is unreachable.
- The SQL Discovery tool no longer crashes when opened in a Windows 7 or Windows 2008 R2 environment.

Connection Check tool

The Server Health Check function of the Connection Check tool now properly connects with SQL Server authentication credentials.

Multi Query tool

This release fixes a possible handle leak in the Multi Query tool.

Database Mover tool

This release corrects an issue causing the Database Mover tool to lock when copying a large database.

User Clone tool

The User Clone tool now properly clones explicit permissions for a database.

Job Mover tool

This release corrects an issue preventing the Job Mover tool from copying the job output file settings.

Quick Reindex tool

- Users who run an edition of SQL Server that supports online index rebuilding now can use that feature in the Quick Reindex tool.
- The Quick Reindex tool now includes the **Fill Factor** column in the SQL Server 2000 version.
- If you want to rebuild multiple indexes with varying fill factor settings, the Quick Reindex tool allows you to retain the current setting of each index by selecting the **Current Fill Factor** option in the **Index Settings** area. Note that the index list also includes the **Fill Factor** column for you to quickly view the fill factor settings of each index.
- This release corrects an issue causing the Quick Reindex tool to use the old **Fill Factor** entry when the entry is manually changed on the back end without first loading the index statistics.
- This release corrects an issue causing the Quick Reindex tool to display an incorrect fragmentation percentage.

Partition Generator tool

The Partition Generator tool no longer displays an error message when a user attempts to use the tool on servers running SQL Server 2008 R2 Data Center Edition.

Inventory Reporter tool

- This release corrects an issue causing the Inventory Reporter tool to show an exception error when you have the **Show Servers as Rows** option turned on, and then click **Copy**.
- This release corrects an issue causing the Inventory Reporter tool to switch the results of the **Minimum SQL Server Memory** and **Maximum SQL Server Memory** columns.

Index Analyzer tool

- The Index Analyzer tool now properly displays the **Fill Factor** column to display statistics for analyzed indexes.
- This release corrects an issue causing the Index Analyzer tool to show an exception error when you type an alphabetic character into the **Hide Indexes under — X rows** field.
- Corrected an issue preventing the proper function of the **Hide Non-Clustered Indexes** check box.
- The Index Analyzer tool now displays an accurate **Selectivity** percentage and the correct total number of records displayed.

Space Analyzer tool

The new **Auto Unit** field in the Space Analyzer tool allows you to select the unit of measurement that best displays your results.

Server Statistics tool

The Server Statistics tool now returns all the correct data when connected to a SQL Server 2012 database.

Server Ping tool

This release corrects an issue causing the Server Ping tool to show an exception to SQL Server 2000 Enterprise Manager users with a local server.

1.6 New features

SQL Server 2012 support

The tools available in the IDERA SQL Admin Toolset now support SQL Server 2012.

1.5 New features

New Linked Server Copy tool

The new Linked Server Copy tool lets you quickly and easily copy our Linked Servers from one SQL Server instance to another along with the associated settings, logins, passwords, and permissions.

New options for the Quick Re-Index tool

The Quick Re-Index tool now allows you to:

- Specify which fill factor you want to use when the selected index is rebuilt.
- Enable online processing during rebuild.
- Use the Maxdop setting to specify the maximum degree of parallelism to use during the rebuild.

1.5 Fixed issues

Index Analyzer

You can now view a list of columns of an index in the order that they were defined.

Patch Analyzer

Patch Analyzer now includes the latest SQL Server service packs.

Multi Query

- Multi Query now correctly sorts the returned results.
- You can now use multi-line strings when performing a query.

Space Analyzer

The Space Analyzer tool now correctly reports the value for the **Maximum Potential Size** metric.

1.3 Fixed issues

Job Editor

You can now successfully retrieve job information from a server group that contains duplicate SQL Server instances.

Job Mover

You can now successfully move SQL Server jobs whose names contain single quote marks (').

Multi Query

You can now successfully clear the returned results of your query from the Query Execution and Results pane.

Server Groups

Special characters are no longer supported when naming new SQL Server groups. Support does continue for existing server groups which were created by IDERA SQL Admin Toolset version 1.2 or earlier, that contain special characters.

SQL Search

You can successfully search for text strings in a SQL Server job even if the MSDB system database is not listed in your search criteria.

Known issues

IDERA strives to ensure our products provide quality solutions for your SQL Server needs. The following known issues are described in this section. If you need further assistance with any issue, please contact [Support \(idera.force.com\)](https://idera.force.com).

Known issues in this version

User Interface

Users who have the Microsoft Windows **Appearance Text and Object Size** setting configured to 120% or 125% may notice some of the information is cut off on the right side. While this issue is scheduled as a fix for the next release, you can avoid the cut off text by changing the setting to 100%.

Login Copy

The Copy Logins function does not work when copying from SQL Server 2000 to SQL Server 2014. Note that if you attempt this procedure, the tool does not copy the login and does not display any error message.

Previous known issues

Linked Server Copy

Local computer accounts are not supported. If you create a linked server and map the security locally, the linked server is created, but you receive an error stating that it is not a valid login.

Database Mover

When copying a large database (10GB or greater) to another server, Database Mover may experience performance issues.

Server Configuration

Offers limited keyboard browsing. When browsing multiple servers, using the directional arrow keys will highlight a server, but not update the server details. Press **Enter** to refresh the screen and display the correct server information.

Welcome to SQL Admin Toolset

IDERA SQL Admin Toolset lets you use many different tools to diagnose, administer, and analyze SQL Server instances across your enterprise.

This wiki helps you understand IDERA SQL Admin Toolset characteristics and provides step-by-step help on how to use all its features.

Need help using IDERA SQL Admin Toolset? See the following sections:

- [Create server groups](#)
- [Set options](#)
- [Start using tools](#)

What is IDERA SQL Admin Toolset?

IDERA SQL Admin Toolset includes 25 essential desktop tools to simplify your day-to-day administration tasks. You can run your every day diagnostics across multiple SQL Servers in minutes instead of hours.

IDERA SQL Admin Toolset diagnostic tools

SQL Backup Status

Check whether your databases are backed up properly.

SQL Search

Search for text anywhere you have SQL code, such as stored procedures, functions, and triggers.

SQL Object Search

Find objects by name anywhere they are hiding in your SQL Servers.

SQL Discovery

Find the SQL Servers lurking on your network.

SQL Password Checker

Check the strength of your SQL Server passwords.

SQL Connection Check

Diagnose connections problems, quickly determine why users cannot reach your SQL Servers.

IDERA SQL Admin Toolset administration tools

SQL Multi Query

Execute and manage queries across multiple SQL Servers.

SQL Database Mover

Move or copy a database and associated logins across servers or relocate data files.

SQL Login Copy

Copy server logins across SQL Servers.

SQL User Clone

Create a new user using an existing user as a template.

SQL Job Mover

Move and copy between servers.

SQL Job Editor

View and edit jobs across multiple SQL Servers.

SQL Quick Reindex

Quickly review and rebuild the indexes on your SQL Servers.

SQL Server Configuration

Compare and edit server configuration for one or more SQL Servers.

SQL Database Configuration

Compare and edit database configuration for one or more databases.

SQL Partition Generator

View existing partition information and generate new partitions.

SQL Connection String Generator

Wade the multitude of options for connection strings and create your own in seconds.

SQL Server Linked Copy Tool

Quickly and easily copy linked servers from one server to another.

SQL Table Pin

View and manage the pinned status of your tables (*SQL Server 2000 and before only*).

IDERA SQL Admin Toolset analytic tools

SQL Inventory Reporter

Get a list of everything important about your SQL Servers, including version, hardware, and more.

SQL Patch Analyzer

Report and analyze your SQL Server version levels.

SQL Index Analyzer

Analyze your indexes for usage patterns and sizes, and determine if you need more or less.

SQL Space Analyzer

Analyze space utilization of your SQL Server by drive or database.

SQL Server Statistics

View loads of statistics about your SQL Server.

SQL Server Ping

Keep a constant eye on the status of your SQL Servers.

How to use the SQL Admin Toolset Launchpad

The IDERA SQL Admin Toolset Launchpad provides a central point of access. From this window, you can see a list of all the tools provided with IDERA SQL Admin Toolset. The Launchpad also provides a brief description of each tool so you can easily find the right tools for your current needs.

Launch an IDERA SQL Admin Toolset tool

To launch a tool from the IDERA SQL Admin Toolset Launchpad, click the button corresponding to the tool you want to use. The tool launches using a separate process, and opens in a new window.

Change your tool options

You can change the options available for IDERA SQL Admin Toolset by accessing the launch pad, and then clicking **Edit SQL Admin Toolset Options**. The [Options](#) window opens, allowing you to manage the connection and command timeouts, and whether you want Launchpad to close after you launch a tool. These options apply to all tools across the toolset.

Access the window to help you manage your server groups

IDERA SQL Admin Toolset lets you manage your server groups through the launch pad window by clicking **Manage Server Groups**. The [Manage Server Groups](#) window opens, allowing you to create, rename, and delete server groups as well as manage the SQL Server instances within these groups.

Use the online IDERA SQL Admin Toolset Help

Help topics are available for all of the features and windows in IDERA SQL Admin Toolset by clicking **SQL Admin Toolset Help**. Help is also available by pressing the F1 key on your keyboard or by clicking **Help** in the Launchpad menu.

Purchase a new license key

To purchase a new license key using Internet access, open the IDERA SQL Admin Toolset Launchpad, and then click **Purchase the SQL Admin Toolset**. The IDERA product purchase page opens, allowing you to select the product(s) you want to purchase.

Get started

IDERA SQL Admin Toolset includes 24 essential desktop tools to simplify your day-to-day administration tasks. You can run your every day diagnostics across multiple SQL Servers in minutes instead of hours.

- [Product requirements](#)
- [How to install IDERA SQL Admin Toolset](#)
- [Upgrade to this release](#)
- [Configure your deployment](#)

Product requirements

IDERA SQL Admin Toolset requires the following hardware, software, and permissions in order to successfully install and use the individual tools.

IDERA SQL Admin Toolset installation requirements

Each computer on which you install one or all of the IDERA SQL Admin Toolset tools must meet these recommended requirements.

Type	Requirements
CPU	2.0 GHz or higher
Memory	1 GB
32-bit or 64-bit Operating System	<ul style="list-style-type: none"> • Windows Server 2019 • Windows Server 2016 • Windows Server 2012 • Windows Server 2012 R2 • Windows 10 • Windows Server 2008 • Windows Server 2008 R2 • Windows 7
Software applications	.NET Framework 4.5.1

Supported Microsoft SQL Server versions

Each computer that hosts the SQL Server instances you want to manage must meet these recommended requirements.

Type	Requirement
Microsoft SQL Server Standard or Enterprise Edition	<ul style="list-style-type: none"> • SQL Server 2019 • SQL Server 2017 • SQL Server 2016 • SQL Server 2014 • SQL Server 2012 • SQL Server 2008 R2 • SQL Server 2008 • SQL Server 2005 SP1 or later • SQL Server 2000 SP4 or later

Partition Generator tool supported SQL Server versions

Partition Generator evaluates partitions on all editions of SQL Server 2016 SP1 and above. SQL Server 2016 RTM is not supported because partition evaluation is not allowed on that version.

IDERA License requirement

This product requires Microsoft Visual C++ 2015 Update 3 Redistributable. If the installer does not detect the correct redistributable in your environment, it will attempt to install or update to the correct version.

Please refer to the following Microsoft Support document for information about VC++ 2015 redistributable prerequisites: <https://support.microsoft.com/en-us/help/3118401/update-for-universal-c-runtime-in-windows>.

IDERA SQL Admin Toolset permission requirements

Some tools require administrative privileges on the managed SQL Server instance in order to perform the desired function. ***If you receive an access error***, check whether you are currently logged on Windows account or the configured authentication credentials belong to the sysadmin fixed role.

Cloud

IDERA SQL Admin Toolset runs on cloud virtual machines, such as Microsoft Azure VM and AWS EC2, with Microsoft Windows. It supports Microsoft SQL Server instances on cloud virtual machines, such as Microsoft Azure VM and AWS EC2.

How to install IDERA SQL Admin Toolset

Install IDERA SQL Admin Toolset on your computer or workstation. You can also install IDERA SQL Admin Toolset on the SQL Server computer hosting the databases you want to manage with these tools.

To install IDERA SQL Admin Toolset:

1. Review the [product requirements](#).
2. Log in to the target computer using an account that has administrator permissions.
3. [Download and install IDERA SQL Admin Toolset](#).
4. From the **Tools available for install** list, select **SQL Admin Toolset**.
5. Click **Install**.

Upgrade to this release

You can easily and quickly upgrade to the latest version of IDERA SQL Admin Toolset. For more information about this release, see the [Release Notes](#).

To upgrade IDERA SQL Admin Toolset to the most recent release:

1. Use an administrator account to log onto the target computer.
2. Start SQL toolbox.
3. Click the Updates tab.
4. Select **SQL Admin Toolset**, and then click **install**. IDERA SQL Admin Toolbox silently installs the latest version.
5. Start the IDERA SQL Admin Toolset launch pad to access the tools you want to use.

Configure your deployment

After initially installing and setting up IDERA SQL Admin Toolset, there are several tasks you might want to do in order to further customize and streamline your deployment including:

- [Changing the account IDERA SQL Admin Toolset uses to connect to SQL Server](#)
- [Creating server groups or changing server group membership](#)
- [Managing your product license](#)
- [Setting global options that affect all tools](#)

Configure authentication

Each tool requires a valid Windows user account or SQL Server login credentials to connect to your SQL Server instances and databases. By default, the IDERA SQL Admin Toolset uses the credentials of your currently logged-on Windows account.

You can configure the authentication method and credentials in the following ways:

- When using any tool, click the user access icon.
- When [managing a server group](#), click **Set Server Credentials**.

These options let you enforce authentication standards and corporate security policies across your server groups while providing the ability to use alternate credentials on an ad-hoc basis to address specific connection needs.

Manage license

IDERA SQL Admin Toolset requires a license for each computer on which it is installed. By default, IDERA SQL Admin Toolset installs with a limited-time trial license key.

Each time you start the tool, IDERA SQL Admin Toolset checks for a valid license. A valid license is a trial or production key that is registered for the computer on which you installed IDERA SQL Admin Toolset.

If your license expires, such as when the trial period ends, or if your license is deactivated, you can no longer use the tool. You can re-install the tool and reuse a previously deactivated production key.

You can view and update your current license key using the licensing console. From the licensing console, you can:

- Verify current status of your license
- Extend your trial period
- Register and activate your license
- Deactivate your license

For more information about using the licensing console, see the [licensing online Help](#).

Access the licensing console

The Licensing Console automatically displays during your trial period to help you register and activate your production license. Once the production license is active, you can use the full functionality of the IDERA SQL Admin Toolset on any SQL Server database hosted by the local computer.

To deactivate a license currently in use, click **Deactivate License** on the Help menu.

Get a new or updated license

You should update your license when:

- You decide to move from a trial implementation to a production implementation.
- You want to deploy the IDERA SQL Admin Toolset to more users.

You can purchase a new license online from the IDERA Website at www.idera.com.

Manage server groups

The Manage Server Groups window helps you organize access to your SQL Server instances so you can fully leverage the scope and power of the IDERA SQL Admin Toolset tools.


A server group can contain multiple instances, whether these instances are hosted on the same computer or on different computers. When you create a server group, you specify a single set of credentials that the IDERA SQL Admin Toolset uses to connect to each instance in the group.

When you use one of the IDERA SQL Admin Toolset tools, you can specify which SQL Server instances you want to act on. Choosing a server group allows you to act on multiple instances that have a logical relationship.

For example, you could create a server group that contains all the SQL Server instances that host accounting databases. Then, when you launch the Backup Status tool, you can simply specify the Accounting server group to check the backup status of all the instances in this group, no matter where they are located in your enterprise.

Manage your server groups to:

- Create a server group
- Change group memberships by adding or removing instances
- Set the SQL Server connection credentials to use when a tool accesses an instance in this group
- Rename an existing server group
- Delete a server group

 When you first launch the IDERA SQL Admin Toolset, it loads any server groups previously defined through SQL Server Management Studio or Enterprise Manager (2000). After the initial import, you can use the Manage Server Groups tool to perform all management of the server groups.

Access Manage Server Groups

You can access the Manage Server Groups window from the Tools menu on any tool window. You can also launch this window from the IDERA SQL Admin Toolset Launchpad by clicking **Manage Server Groups**.

Nest your server groups

You can nest server groups to better organize your SQL Server instances. Nesting is beneficial when you need to organize several small server groups or when you need to sort instances by multiple properties, such as local and data type. For example, nesting HR databases in Chicago, Houston, and London.

If you select a nested server group when running one of IDERA SQL Admin Toolset tools, the tool acts on all the instances belonging to the selected server group as well

as any instances belonging to the child server groups. Using our previous example, when you select the HR server group, the tool acts on the instances belonging to the Chicago, Houston, and London server groups.

Set SQL Admin Toolset options

You can configure options that control the behavior of all tools in IDERA SQL Admin Toolset. You can edit these options from any tool in the toolset, and immediately apply them every tool. The following options are available:

Connection Timeout

Controls the amount of time in seconds before a connection attempt to a targeted SQL Server instance fails with a timeout error. ***If you are experiencing timeout errors while using the tools***, you may have a slow network connection or a busy SQL Server, and may need to increase this value. By default, IDERA SQL Admin Toolset uses a connection timeout of 30 seconds.

Command Timeout

Controls how long a tool should wait in seconds for a command against a targeted SQL Server instance to complete. ***If you are experiencing timeout errors while using the tools***, you may have a slow network connection or a busy SQL Server, and may need to increase this value. By default, IDERA SQL Admin Toolset uses a connection timeout of 30 seconds

Close launch pad after launching a tool

Controls the behavior of IDERA SQL Admin Toolset Launch Pad window after you launch a tool. You can close the launch pad or leave it open after a tool is launched. The first time you launch a tool from IDERA SQL Admin Toolset, you are prompted for your preference.

Access SQL Admin Toolset Options

You can access IDERA SQL Admin Toolset Options from the Tools menu by clicking **SQL Admin Toolset Options**. To access options from IDERA SQL Admin Toolset Launch Pad window, click **Edit SQL Admin Toolset Options**.

Diagnostic tools

The IDERA SQL Admin Toolset contains the following diagnostic tools:

Backup Status

Check whether your databases are backed up, and how much time has passed since the last backup.

Connection Check

Diagnose connection problems on your SQL Server.

Object Search

Find objects anywhere they are hiding in your SQL Server.

Password Checker

Check the strength of your SQL Server login passwords.

SQL Discovery

Find SQL Servers lurking on your network.

SQL Search

Search for a specific word or phrase anywhere you have SQL code, such as stored procedures, functions, or triggers.

Backup Status

The Backup Status tool monitors your SQL Server backup status, helping to make sure that you back up your SQL Servers on a regular basis. By default, Backup Status [connects](#) to the specified SQL Server instance using the credentials of your currently logged on Windows account, and then collects information about backups performed on the hosted databases.

The Backup Status tool helps you:

- Identify any non-backed up databases.
- Determine databases not backed up within the last x days.
- Determine which databases are in simple or full recovery mode.
- View the last backup time stamp for each database.
- Drill into the complete backup history for a specific database.
- View backup history across multiple SQL Server instances or a server group.
- Save the backup history.

Importance of backups and the Backup Status tool

Performing regular back ups of all databases on your SQL Server instances allows you to keep sensitive and mission-critical data safe. It is important that you determine your normal backup schedule. For example, if you want to back up all databases every week, you might want to skip databases backed up within the past seven days, allowing you to focus on databases with backup sets older than eight days. The Backup Status tool offers the following options when checking the backup status of an instance:

Skip databases with a backup in the last x days

Check this box for the Backup Status tool to exclude databases that include a backup performed in the past x number of days.

Skip system databases

Check this box for the Backup Status tool to exclude system database backups.

Only include full database backups

Check this box for the Backup Status tool to include only those databases that include a full database backup.

Exclude Offline Databases

Check this box for the Backup Status tool to exclude all offline databases from the backup status check.

Include all nodes

Check this box for the Backup Status tool to include all nodes in the backup status check.

Use the Backup Status tool

To check the backup status of one or more servers using the Backup Status tool:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Backup Status**.
2. Specify whether you want to back up a SQL Server instance or a server group.
3. Search for and then select the SQL Server instance or server group whose backup status you want to check. To specify multiple SQL Servers, separate each instance name with a semicolon.
4. *Optional*. Check the options you want to use in this status check.
5. Click **Get Backup Status**. The Backup Status tool checks the backup status of the appropriate instance or group.
6. ***If you want to view the backup history for a specific database***, select the database from the status list, and then click **Show Backup History for Database**.
7. ***If the tool returns unexpected results***, check the groom settings on the SQL Server instance. For example, if the backup operation status is recently groomed, the tool may return a partial history, such as a transaction log backup record with no corresponding full backup record.
8. ***If you want to save the results***, click **Copy Results To Clipboard**, or right-click to save the results in a data table, or as an XML or CSV file.

SQL Search

The SQL Search tool helps you find specific words or text phrases in SQL Server schema objects including table columns, T-SQL scripts, or other SQL code in use on an instance. With this global search, you can quickly and easily complete tasks such as updating copyright notices or changing all usages of a stored procedure call.

The SQL Search tool helps you:

- Perform case-sensitive searches.
- Search across all SQL on a SQL Server instance.
- Search SQL used on specific databases or by specific objects.
- Search SQL on system databases.
- Search for an exact word or text phrase, or use SQL wild cards to broaden your results.
- See how many times the specified word or phrase appears, which objects contain this string, and the full SQL source.
- Save your search results.

How to SQL Search tool works

By default, SQL Search [connects](#) to the specified SQL Server instance using the credentials of your currently logged on Windows account. SQL Search then looks at all the SQL code used by stored procedures, functions, triggers, and views to determine when a specific word or phrase is being used on a SQL Server instance. The SQL Search tool offers the following options when searching for specific text:

Limit to first x matches

Before using SQL Search, be sure to decide how broad a search you actually need. Consider limiting your search to only the necessary databases and objects. Keep in mind that large searches impact performance. By default, the SQL Search tool returns a maximum of 500 matches.

Perform a case-sensitive search

Limit your results to only text matching the case of the search text.

Use SQL wildcards

Allow SQL wildcard characters to retrieve more results. To search for text variations using SQL wildcards, append or prefix the word with an underscore character. For example, "*lab*" or "*lab*".

Include system databases

Include system databases in your search. Leave this box unchecked to exclude system databases from your search.

Use the SQL Search tool

To search for specific text using the SQL Search tool:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **SQL Search**.
2. Search for and then select the SQL Server instance whose schema you want to search.
3. Search for and then select the database on this instance you want to search. To specify multiple databases, separate each database name with a semicolon. Leave this field empty if you want to search all user databases.
4. Type the word or phrase you want to find. To search for text variations using SQL wildcards, append or prefix the word with an underscore character. For example, "*lab*" or "*lab*".
5. *Optional*. Check the options you want to use in this SQL text search.
6. Click **Perform Search**.
7. ***If you want to see the SQL source*** , select the SQL excerpt, and then click **View Source**.
8. ***If you want to save the results*** , click **Copy Results To Clipboard**, or right-click to save the results as an XML or CSV file.

Object Search

The Object Search tool helps you find tables and objects across your SQL Server environment by searching for specific words or text phrases used in object names. With this global search, you can quickly and easily determine where particular objects, such as a table, reside or how many times the same object, such as a login, was created on different instances.

The Object Search tool helps you:

- Perform case-sensitive searches.
- Search across all objects on a SQL Server instance or server group.
- Search for objects in specific databases.
- Search for objects in system databases.
- Search for an exact word or text phrase, or use SQL wild cards to broaden your results.
- Save your search results.

How the Object Search tool works

By default, Object Search [connects](#) to the specified SQL Server instances using the credentials of your currently logged on Windows account, and then collects data for any object that matches the word or phrase you specified.

Limit to first x matches

Before using Object Search, be sure to decide how broad a search you actually need. Consider limiting your search to only the necessary databases. Keep in mind that large searches impact performance. By default, the Object Search tool returns a maximum of 500 matches.

Use SQL wildcards

Allow SQL wildcard characters to retrieve more results. To search for text variations using SQL wildcards, append or prefix the word with an underscore character. For example, "*lab*" or "*lab*".

Case-sensitive search

Limit your results to only text matching the case of the search text.

Exclude system objects

Exclude system objects from your search. Leave this box unchecked to include system databases in your search.

Database

Type the name of one or more databases that you want to search. Leave this field blank to include all databases in your search.

Use the Object Search tool

To search for specific objects using the Object Search tool:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Object Search**.
2. Specify whether you want to back up a SQL Server instance or a server group.
3. Search for and then select the SQL Server instance or server group that you want to search. To specify multiple SQL Servers, separate each instance name with a semicolon.
4. Type the word or phrase you want to search for. To search for text variations using SQL wildcards, append or prefix the word with an underscore character (for example, "*lab*" or "*lab*").
5. *Optional*. Check the options you want to use in this object text search.
6. Click **Perform Search**.
7. ***If you want to save the results***, click **Copy Results To Clipboard**, or right-click to save the results as an XML or CSV file.

SQL Discovery

The SQL Discovery tool helps you find new or rogue SQL Server instances that are online in your network. This tool allows you to locate SQL Server instances and create a server group for the SQL Server instances you locate. Once you identify and group newly-discovered SQL Server instances, you can properly lock down these instances to prevent any security issues.

SQL Discovery searches computers hosting SQL Server instances by specifying IP ranges, a list of computers, or servers registered with Active Directory or Network Browser Service. You can also use a number of different scans or probes to increase the effectiveness of your search. The following table displays a list of available probes and their purposes.

Type of probe	What it does
Active Directory Probe	Checks for SQL Server instances registered with Windows Active Directory.
Browser Service Probe	Checks for SQL Server instances registered with Network Browser Service.
Service Control Manager Probe	Checks with Service Control Manager for installed SQL Server instances.
SQL Server Resolution Service Probe	Checks port 1434 to determine whether SQL Server instances are registered with the SQL Server Resolution Service.
TCP Probe	Checks the standard ports SQL Server uses to listen. The standard ports are 1433 and 2433.
Windows Registry Probe	Checks the registry on each computer to detect whether SQL Server services are installed.
WMI Enumeration Probe	Checks for installed SQL Server instances by using WMI for the search.

Once the results are available, you can review the list of discovered SQL Servers, see which probe found each instance, and check the configuration details of each instance.

Considerations before using the SQL discovery tool

One factor to consider before using SQL Discovery is to check the firewall settings your local computer, such as the native Windows Firewall or any third-party firewall application. SQL Discovery uses ports 1433, 1434, and 2433 to scan your network for SQL Server instances. Make sure these ports are open and enabled for communication.

You can use Inventory Reporter, Patch Analyzer, and Password Checker with SQL Discovery to further enhance your validation. These other tools work with the SQL Discovery tool in the following ways:

Inventory Reporter

Save the found SQL Server instances as a new server group, and then run [Inventory Reporter](#) against this server group to capture configuration details.

Patch Analyzer

Use [Patch Analyzer](#) to verify that the found SQL Server instances are properly patched and are running supported versions of Microsoft SQL Server software.

Password Checker

Use [Password Checker](#) to test the security of the found SQL Server instances.

Use the SQL Discovery tool

To find new or rogue SQL Server instances using SQL Discovery:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **SQL Discovery**.
2. On the Welcome window, click **Next**.
3. Select which scan technique you want to use to locate SQL Server instances on your network, and then click **Next**.
4. **If you selected IP Range in the previous window**, specify one or more IP addresses that you want to scan.
5. **If you selected SQL Server(s) in the previous window**, specify one or more SQL Servers or the server group on whose host computers you want to perform a scan. To specify multiple SQL Servers, separate each name with a semicolon.
6. **If you selected Computer(s) in the previous window**, specify one or more computers to scan for SQL Servers. To specify multiple computers, separate each name with a semicolon.
7. Click **Next**.
8. Choose which probes you want SQL Discovery to run as part of the discovery process, and then click **Finish**.
9. Review the scan results. To see more details about a discovered instance, select that instance from the Scan Results list, and then view the details displayed below.
10. **If you want to create a server group that contains these instances**, click **Save as Server Group**.
11. **If you want to save the results**, click **Copy Results To Clipboard**, or export the list as an XML or CSV file.

Password Checker

The Password Checker tool helps you evaluate and strengthen your SQL Server password policy, such as finding weak passwords that can make your SQL Servers vulnerable to outside attack. You can see exactly which logins are at risk so you can replace the weak passwords with strong passwords.

The Password Checker tool allows you to:

- Use proven approaches to test password strength, including use of built-in password lists.
- Use a custom password list that enforces your specific corporate password policy.
- Choose which logins you want to check.
- Drill down to identify logins associated with weak passwords.
- Save your test results.

By default, Password Checker [connects](#) to the specified SQL Server instances using the credentials of your currently logged on Windows account.

Password Checker then finds bad passwords by performing a dictionary scan against one of the built-in password lists provided with the Password Checker tool or against your own custom list.

Considerations before using the Password Checker tool

The Password Checker tool uses a list of common words to test for bad passwords. As a best practice, users should not use blank passwords, common words, or passwords that match a login name. The Password Checker tool uses lists of common passwords compiled by industry experts.

There are a number of factors you should consider before using the Password Checker. It is important that you:

- Decide whether you need to check specific SQL Server instances or a server group.
- Ensure the [authentication account](#) has CONTROL SERVER or sysadmin privileges on each instance you intend to check.
- Choose which password list you want to use for the check. You can check against your own password, a custom password file, or one of the built-in password lists. You can also check for blank passwords.

You can use Manage Server Groups and SQL Discovery with the Password Checker tool to further enhance your validation. These other tools work with the Password Checker tool in the following ways:

Manage Server Groups

Use [Manage Server Groups](#) to create logical groups of SQL Server instances. By creating a server group, you can easily check passwords on instances that are mission-critical or host sensitive data.

SQL Discovery

Use [SQL Discovery](#) to find SQL Server instances on your network, and then use the Password Checker tool to test the security of those instances.

Use the Password Checker tool

To check a password using the Password Checker tool:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Password Checker**.
2. Specify whether you want to check passwords on a SQL Server instance or a server group.
3. Search for and then select the SQL Server instance or server group whose backup status you want to check. To specify multiple SQL Servers, separate each instance name with a semicolon.
4. Check the options of passwords you want to check against. You can use:
 - **Top 10 List**. Contains the most popular passwords as defined by *PC Magazine* and actually contains 11 passwords as a blank password is in the list.
 - **Blank password and passwords matching login**. Contains a blank password plus any passwords that match the user account used to log in to the system.
 - **Nifty Fifty**. Contains 50 of the most commonly used passwords including a blank password and the user name as a password.
 - **Other**. Allows you to specify specific passwords to look for.
 - **800 Common Passwords**. Contains 800 of the most commonly used passwords including a blank password and the user name as a password.
 - **Custom List**. Allows you to attach a file to check passwords against.
 - **The Big List (2400+ Passwords)**. Contains over 2400 of the most commonly used passwords including a blank password and the user name as a password.
 - **Test Common Variations of Selected Passwords**. Checks different variations of passwords already selected using one of the previous check boxes.
5. Specify which logins you want to check on the selected SQL Servers, separating each login name with a semicolon. Always check logins containing elevated privileges, such as members of the built-in server roles.

6. **Check Passwords.**
7. View the results.
8. ***If you want to view login details for a specific SQL Server instance*** , click the appropriate login listed in the **Test Results** right pane.
9. ***If you want to save the results*** , right-click to save the results as a TXT, XML, or CSV file.

Create a custom password list

The Password Checker tool allows you to attach a document containing a custom list of passwords to check against. You can attach this list using the **Custom List** check box, and then searching for and selecting the appropriate file.

To create a custom password list using the Password Checker list:

1. Open a text editor program, such as Notepad.
2. Type each word or phrase on individual lines. This list should represent strings that should not be used in SQL Server login passwords.
3. Save the file in a secured folder.

Connection Check

The Connection Check tool helps you check your SQL Server connections and diagnose why a database user cannot connect to a specific instance. Connection problems are difficult to identify because there are many causes, such as a service not restarting after routine maintenance or an incorrect network configuration setting on the client computer. With this tool, database administrators and Help Desk technicians can quickly and easily assess connection problems.

How the Connection Check tool works

The Connection Check tool allows you to:

- Check whether you can connect to a SQL Server.
- Validate the network connection between the client computer and the SQL Server computer including proper network and DNS configuration.
- Check the status of the SQL Server service.
- Receive best-practice recommendations for your next steps.
- Specify alternate SQL login or Windows user credentials to test the database access from the end-user viewpoint.

By default, Connection Checker attempts to [connect](#) to the specified SQL Server instance using the credentials of your currently logged on Windows account. Connection Check then performs a series of tests that diagnose what is working on your SQL Server computer. These tests validate the SQL Server configuration and the network connectivity. After Connection Check completes these tests, it provides best-practice guidance regarding the steps required to confirm the diagnoses and repair the problem.

Use the Connection Check tool

To check the connection on a SQL Server instance using the Connection Check tool:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Connection Check**.
2. Search for and then select the SQL Server instance whose connection you want to check.
3. *Optional*. Type the name of the database you want to check.
4. Type which port you want to use during the test. By default, the Connection Check tool uses port 1433.
5. Specify the account credentials you want to use to perform the test.
6. Click the appropriate button for the type of test you want to run.
7. Use the Test Results pane to review the individual test results as each test runs, and then use the Recommendation pane to review your next steps.

Administration tools

The IDERA SQL Admin Toolset contains the following administration tools:

Connection String

Create connection strings in seconds by picking and choosing from the available options.

Database configuration

Compare and edit configurations for one or more SQL Server databases.

Database Mover

Move or copy databases and their associated logins within or between SQL Server instances.

Job Editor

View and edit jobs across multiple SQL Server instances.

Job Mover

Copy or move your jobs between SQL Servers.

Linked Server Copy

Copy linked servers from one SQL Server instance to another.

Login Copy

Synchronize logins across SQL Server instances by copying logins from one instance to another.

Multi Query

Execute and manage queries across multiple SQL Server instances.

Partition Generator

View existing partition information and generate new data partitions quickly and easily.

Quick Reindex

Review index fragmentation and rebuild indexes on your SQL Server instances.

Server Configuration

Compare and edit configurations for one or more SQL Server instances.

Table Pin

View and manage the pinned status of your tables (*SQL Server 2000 and before only*).

User Clone

Create a new SQL Server login using an existing login as a template.

Multi Query

The Multi Query tool helps you easily query multiple SQL Server instances and databases across your enterprise. With this tool, you can run T-SQL scripts to perform any command supported by SQL Server. For example, you can compare data points across instances or deploy a schema change across databases.

The Multi Query tool helps you:

- Run ad-hoc queries to make quick changes on multiple instances and databases.
- Run existing queries to perform routine tasks or schema maintenance.
- Use queries to compare data points from multiple instances and databases.
- Create and reuse target lists of the instances and databases on which a query should act.
- Save the returned result sets to track statistics or object properties over time.

How the Multi Query tool works

Run the Multi Query tool on your instances and select a database. If you do not specify a database, the Multi Query tool connects to and runs the query on the default database configured in SQL Server.

Use the Query Options to narrow the scope of the returned result set and improve performance when running queries that return more results than you expected, are run against many different targets, or contain complex functions. You can search targeted databases using a wildcard. The Add Query Target dialog includes fields that allow a user to enter a search term or expression to filter for a list of matching databases.

By default, Multi Query **connects** to the specified SQL Server instances using the credentials of your currently logged on Windows account, and then runs the specified query against those instances.

Use the Multi Query tool

To create a query target list using the Multi Query tool:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Multi Query**.
2. On the Query Targets task bar, click **Add**.
3. Search for and then select the SQL Server instance or server group that you want to query. To specify multiple SQL Servers or server groups, separate each name with a semicolon.
4. Select the authentication type you want the Multi Query tool to use to connect to these instances.
5. *Optional*. Click **Test Connection** to verify that you can connect using the entered authentication specifics.

6. Indicate whether you want the Multi Query tool to act on the default database (as defined in SQL Server) or a specific database. You can filter the list of databases using a wildcard.
7. Click **OK**.
8. On the Query Targets task bar, click **Save**. Consider specifying a name that correlates with the queries you intend to run against this list of targets.

To run a query using the Multi Query tool:

1. In the Query Targets pane, specify which instances and databases that you want the query to target. You can specify new targets by clicking **Add**, or use an existing target by clicking **Open**, and then searching for and selecting the appropriate target.
2. In the Query pane, enter the query you want to run. You can type an ad-hoc query in the provided text pane, or run an existing script by clicking **Open**, and then searching for and selecting the appropriate script.
3. ***If you want to run your query against all the listed targets*** , click **Execute Query**. ***If you want to run your query against specific targets*** , click **Execute Against Selected Query Targets**.
4. View the query status and results sets. You can:
 - View the query status using the **Summary** tab.
 - Combine result sets that contain the same columns using the **Combined Results** tab.
 - View individual result sets for a specific target.
 - Maximize the pane by clicking **Full Screen**.
5. ***If you want to save the results*** , copy the list to clipboard, or export the list as an XML or CSV file.

Quick Reindex

The Quick Reindex tool helps you quickly rebuild and reorganize SQL Server indexes. You can easily view the fragmentation level for all of the indexes on a SQL Server instance and then decide your course of action. You can filter the results by server, database, or table and sort indexes based on the fragmentation level, size, or name. To further help you analyze results, you can hide small or disabled indexes.

How the Quick Reindex tool works

By default, Quick Reindex **connects** to the specified SQL Server instance using the credentials of your currently logged on Windows account, and then collects detailed information about the database indexes.

If an index shows a high fragmentation value after it is organized, verify the size of the index before continuing as small indexes often do not benefit from reorganization because fragmentation on small indexes is difficult to control. For more information, see Microsoft SQL Server Books Online.

Use the Quick Reindex tool

To rebuild your SQL Server indexes using the Quick Reindex tool:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Quick Reindex**.
2. Specify the SQL Server whose indexes you want to check.
3. Click **Load Index Statistics**.
4. ***If you want to filter the index view by database object***, select a database or table that contains the indexes you want view. Click the column headers to sort the indexes.
5. ***If you want to resolve high fragmentation***, select one or more fragmented indexes, and then click either **Reorganize Indexes** or **Rebuild Indexes**.
6. ***If you want to reduce the strain on your system***, enter the **Maximum Degree of Parallelism** in the Quick Reindex warning window. Maximum degrees of parallelism refers to the number of processor cores to be used in the reindex. The default value of 0 allows SQL Server to use all the available processors up to 64.
7. ***If you want to perform index rebuilds using the Sort in tempdb***, select **On** in the Quick Reindex warning window..
8. Select **Online Rebuild** if you would like to perform the operation without taking the selected indexes offline.

 The **Online Rebuild** and **Max Degree of Parallelism** options are only available on SQL Server Enterprise and Developer versions.

Database Mover

The Database Mover tool helps you move databases from one SQL Server instance to another, preserving the associated data files and the SQL Server logins. With this tool, you can move or copy databases between SQL Server instances or on the same SQL Server instance. You can also relocate the data files for a database to a different storage device while keeping the database on the same SQL Server. For example, you can move a performance intensive database to a faster hard disk to speed up performance. You can also rename a database while moving or copying to another location.

i The Database Mover tool allows all moves including moves from higher versions to lower versions and higher editions to lower editions. Note that some features available in the higher versions/editions may not be available in the lower versions/editions. Users receive a warning message before moving from the higher to lower version/edition, and must confirm the move.

Windows Cluster support

The Database Mover tool supports the destination SQL Server instance on a Microsoft Windows cluster with cluster-shared volumes (CSV). The drive must be a member of the cluster group in order to have the files attached to the database and the user performing the move must have all permissions to perform database operations.

How the Database Mover tool works

The Database Mover tool requires your Windows user account to have administrative access to the target SQL Server instance because the tool uses administrative shares to perform the file copy. By default, the Database Mover tool **connects** to the specified SQL Server instance using the credentials of your currently logged on Windows account, and then collects information about the specified database.

To perform the move, the Database Mover tool detaches the source database to copy the data files for the specified database to the target SQL Server instance. The tool then attaches the database at the source and target instances, and copies the SQL Server logins to the target instance. The copy database and move data files operation leaves the source database intact. The move database operation drops or deletes the source database.

Use the Database Mover tool

To move a database from one SQL Server instance to another:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Database Mover**.
2. On the Welcome window, click **Next**.

3. Select which action you want to perform, and then click **Next**.
4. ***If you are copying or moving a database*** , select the source and destination SQL Server instances and databases. To rename the database during the copy or move operation, specify a new name for the destination database. Click **Next**.
5. ***If you are moving data files*** , select the source SQL Server instance and database, and then click **Next**.
6. Specify the appropriate copy or move options, and then click **Next**.
7. Specify the target folder to which the data files should be copied or moved, and then click **Next**.
8. Select which SQL Server logins you want to copy or move to the target instance, and then click **Next**.
9. Review the summary, and then click **Finish**.

Server Configuration

The Server Configuration tool helps you review, compare, and edit the configuration settings of one or more SQL Server instances. With this tool, you can:

- Diagnose problems and identify unwanted differences by comparing the configuration values of an instance to its historical snapshot or another instance
- Roll out changes to one or more instances across your enterprise
- Report on configuration settings by instance or by server group

How the Server Configuration tool works

The Server Configuration tool allows you to manage the configuration of a single instance by displaying the current value and description of each configuration setting. You can edit these settings using the Server configuration tool, and then save and compare past and present snapshots of these settings. You can also reset the configuration to match a previous version, if necessary.

When comparing values between multiple instances, the Server Configuration tool lets you edit any of these settings, and then roll out a value across the selected instances. You can designate a configuration as a baseline using live data or a snapshot and make a set of instances match this new baseline.

By default, Server Configuration **connects** to the specified SQL Server instances using the credentials of your currently logged on Windows account, and then collects configuration information about each instance.

Use the Server Configuration tool

To view and report on your configuration settings:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Server Configuration**.
2. Specify the SQL Server instance or server group that contains the configuration settings you want to check. To specify multiple SQL Servers, separate each instance name with a semicolon.
3. Click **Get Configuration**.
4. On the Configuration tab, select the instance whose configuration you want to view, and then select the setting in the right pane to view details.
5. **If you want to save this configuration for future reference**, click **Save Snapshot**.
6. **If you want to save the results as a report**, copy the list to clipboard, or save the results as an XML or CSV file.

To change your configuration settings:

1. Get the current configuration settings for the SQL Server instances you want to change.
2. ***If you want to change one setting at a time*** , complete the following steps on each instance:
 - a. On the Configuration tab, select the configuration setting you want to change.
 - b. In the Details pane, click **Edit**.
 - c. Specify the new value, and then click **OK**.
3. ***If you want to perform a bulk update*** , complete the following steps:
 - a. On the Comparison tab, right-click the configuration setting you want to change, and then click **Bulk update** on the context menu.
 - b. Specify the new value, select which instances you want to update, and then click **OK**.
4. ***If you want to change a configuration setting to a baseline value*** , complete the following steps:
 - a. On the Comparison tab, right-click the configuration setting you want to change, and then click **Set baseline value to all servers** on the context menu.
 - b. When prompted to confirm, click **OK**.
5. Close the Server Configuration window, and restart each instance you updated.

To compare your configuration against a baseline:

1. Get the current configuration settings for the SQL Server instances you want to compare.
2. On the Configuration tab, select the instance you want to use as the baseline, and then click **Set as Baseline** on the context menu.
3. ***If you want to change a configuration setting to the baseline value*** , click **Fix Differences**.
4. On the Baseline Configuration options window, verify the settings that you want to update, and then click **OK**.

To compare your configuration against a snapshot:

1. Save a snapshot of a SQL Server instance configuration you would like to use as your standard, or template.
2. Get the current configuration settings for the SQL Server instances you want to compare.
3. On the Configuration tab, click **Open Snapshot**.
4. ***If you want to change a configuration setting to the snapshot value*** , click **Fix Differences**.
5. On the Snapshot Configuration options window, verify the settings that you want to update, and then click **OK**.

Login Copy

The Login Copy tool helps you keep SQL Server logins synchronized between instances by letting you transfer logins using a simple, easy-to-use interface. Easy login transfer is important because it is difficult administering parallel SQL Server environments such as test and production systems. Use the Login Copy tool to preview the missing SQL Server logins on a secondary SQL Server or move the SQL Server Logins to the secondary SQL Server.

How the Login Copy tool works

With this tool, you can ensure SQL Server logins required for database access on multiple SQL Servers are up to date and Supplement existing backup and recovery scripts by capturing the SQL Server logins stored outside the databases at the server level.

By default, the Login Copy tool **connects** to the specified SQL Server instance using the credentials of your currently logged on Windows account. The Login Copy tool then copies the appropriate login information and recreates the SQL Server logins on the target instance.

Use the Login Copy tool

To sync your SQL Server instances using Login Copy:

1. Open the IDERA SQL admin toolset Launchpad, and then click **Login Copy**.
2. Specify the source SQL Server instance for the move login operation. The source SQL Server instance contains the SQL Server logins you want to move.
3. Specify the destination SQL Server.
4. Click **Preview Missing Logins** to see the list of the logins that exist on the source SQL Server but do not exist on the target SQL Server.
5. Check which logins you want to move to the target SQL Server.
6. Click **Copy Logins** to create the selected logins on the target SQL Server.

Database Configuration

The Database Configuration tool helps you review, compare, and edit the configuration settings of one or more SQL Server databases. With this tool, you can diagnose problems and identify unwanted differences by comparing the configuration values of a database to its historical snapshot or another instance.

This action allows you to roll out changes to one or more databases across your enterprise. These results allow you to report on your database configuration settings by instance or by server group.

How the Database Configuration tool works

The Database Configuration tool allows you to manage the configuration of a single database or monitor and change the configuration of multiple databases. When viewing the value and description of each configuration setting for a single database. You can edit any configuration setting, and then save your changes. Use this tool to compare values between configuration snapshots. You can designate a configuration as a baseline using live data or a snapshot. If necessary, you can always reset a database configuration to match a previous version.

When monitoring and changing the configuration of multiple databases it is important that you compare the values between the databases beforehand. Make any necessary changes before rolling out a value across a set of databases. The Database Configuration tool also allows you to designate a configuration as a baseline, and then making a set of databases match the baseline database.

By default, Database Configuration [connects](#) to the specified SQL Server instances using the credentials of your currently logged on Windows account, and then collects configuration information about each database.

Use the Database Configuration tool

To view and report on configuration settings:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Database Configuration**.
2. Specify the SQL Server instance or server group that hosts the databases whose configurations you want to check. To specify multiple SQL Servers, separate each instance name with a semicolon.
3. Click **Get Configuration**.
4. On the Configuration tab, expand the host instance, and then select the database to view the details in the right pane.
5. ***If you want to save this configuration for future reference***, click **Save Snapshot**.

6. ***If you want to save the results as a report*** , copy the list to clipboard, or export the list as an XML or CSV file.

To change your configuration settings:

1. Get the current configuration settings for the SQL Server databases you want to change.
2. On the Configuration tab, select the configuration setting you want to change.
3. On the right-click context menu, click **Edit**.
4. Specify the new value, and then click **OK**.
5. Close the Server Configuration window, and restart each host instance for the databases you updated.

To compare results against a baseline:

1. Get the current configuration settings for the databases you want to compare.
2. On the Configuration tab, select the database you want to use as the baseline, and then click **Set as Baseline**.
3. ***If you want to change a configuration setting to the baseline value*** , click **Fix Differences**.

To compare results against a snapshot:

1. Save a snapshot of a database configuration you would like to use as your standard or template.
2. Get the current configuration settings for the databases you want to compare.
3. On the Configuration tab, click **Open Snapshot**.
4. ***If you want to change a configuration setting to the snapshot value*** , click **Fix Differences**.

User Clone

- ❗ You must run the following SQL Admin Toolset tools under the Administrator account for proper functionality:
- User Clone
 - [Patch Analyzer](#)
 - [Partition Generator](#)

The User Clone tool helps you create new logins by copying one or more existing SQL Server users to use as templates. You can immediately perform the clone operation through the tool, or generate a T-SQL script that you can preview and edit.

How the User Clone tool works

Use the User Clone tool to create new user accounts from one or more existing accounts. You can copy a SQL Server login across SQL Servers and keep the same access settings and specify a new default database or password during the clone. After creating a new account, you can set the new SQL Server login as enabled or disabled.

You also can generate a T-SQL script that you can integrate into your existing work flows.

By default, User Clone [connects](#) to the specified SQL Server instance using the credentials of your currently logged on Windows account. The User Clone tool then copies the specified source SQL Server login and recreates it on the target SQL Server instance and databases. By default, the new login shares the same password, server and database roles, permissions and database access of the source login.

Use the User Clone tool

To use User Clone:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **User Clone**.
2. Specify the SQL Server login(s) you want to clone and the SQL Server instance where these accounts are located. By default, User Clone lists the available logins on the specified instance.
3. Specify the SQL Server instance and database where you want to create the new login(s). The new login name is the same as the source login name.
4. Select which security properties you would like to clone.
5. To perform the clone operation immediately, click **Clone User**.
6. ***If you want to create a T-SQL script based on your settings***, choose the appropriate action based on the version of SQL Server you are running.

7. ***If you want to save the script***, copy the code to clipboard, or save the code to a file.

Partition Generator

- i** You must run the following SQL Admin Toolset tools under the Administrator account for proper functionality:
- Partition Generator
 - [Patch Analyzer](#)
 - [User Clone](#)

The Partition Generator tool helps you create and manage SQL Server table partitions, which provides better data management for large tables. A large, unpartitioned table can cause performance issues, blocking issues, costly backups, and overall scalability issues as the table continues to grow. Partitions can help solve these problems by splitting the data into smaller, more manageable segments.

Supported SQL Server versions

Partition Generator evaluates partitions on all editions of SQL Server 2016 SP1 and above. SQL Server 2016 RTM is not supported because partition evaluation is not allowed on that version.

How the Partition Generator works

The Partition Generator tool allows you to quickly determine whether a table is partitioned or needs partitioning. You can create the necessary partitions and specify the appropriate data ranges or change existing partitions by merging or splitting data ranges.

Before using the Partition Generator tool, it is important to know that table partitioning is supported by SQL Server 2005 or later. For more information, see Microsoft SQL Server Books Online.

By default, the Partition Generator tool [connects](#) to the specified SQL Server instances using the credentials of your currently logged on Windows account, and then collects partition information about each table. The Partition Generator tool uses partition functions and partition schemes to define how the data should be split.

Use the Partition Generator tool

To use the Partition Generator:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Partition Generator**.

2. Specify the SQL Server instance or server group that contains the database you want to partition. To specify multiple SQL Servers, separate each instance name with a semicolon.
3. Click **Get Partition Information**.
4. To see only partitioned tables, select that option on the **Table** pane.
5. ***If you want to create a partition*** , complete these steps:
 - a. In the **Partition Details** column, click **Create** for the table you want to partition.
 - b. On the Partition Information window, specify the appropriate partition settings.
 - c. To add a new data range, click **Add new**, and then type the range values in the new grid row. Ensure the range values are in ascending order and can be converted to the SQL data type of the partition column.
 - d. Click **OK**.
6. ***If you want to edit a partition*** , complete these steps:
 - a. In the **Partition Details** column, click **Edit** for the partitioned table you want to update.
 - b. On the Partition Information window, change the appropriate partition settings.
 - c. To add a new data range, click **Add new**, and then type the range values in the new grid row. Ensure the range values are in ascending order and can be converted to the SQL data type of the partition column.
 - d. Click **OK**.
7. Click **OK**.

Job Mover

The Job Mover tool helps you move or copy SQL Server jobs from one SQL Server instance or database to another, preserving the associated actions as well as the assigned user privileges. You can also rename jobs during the move or copy operation. The Job Mover tool helps you achieve a faster result than script-based solutions.

How the Job Mover works

By default, the Job Mover tool **connects** to the specified SQL Server instance using the credentials of your currently logged on Windows account, and then collects information about the SQL Server jobs. Depending on your selection, Job Mover then copies or moves the specified jobs. ***If you are moving a job from one instance to another***, Job Mover deletes the original job from the source instance.

Before using Job Mover, it is important that you understand that Job Mover requires your Windows user account to have administrative access to the target SQL Server instance. Job Mover uses administrative shares to perform the job copy.

Use the Job Mover tool

To move jobs using Job Mover:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Job Mover**.
2. On the Welcome window, click **Next**.
3. Select which action you want to perform, and then click **Next**.
4. ***If you are copying or moving a job to another instance***, select the source and target SQL Server instance, and then click **Next**.
5. ***If you are copying a job to another database on the same instance***, select the source SQL Server instance, and then click **Next**.
6. Select which jobs you want to move, and then click **Next**.
7. Indicate whether you want to keep the existing job schedule and native SQL Server alerts and notification settings, and then click **Next**.
8. Configure options for the new job, including whether the jobs are enabled or disabled, and then click **Next**.
9. Review the summary, and then click **Finish**.

Connection String Generator

The Connection String Generator tool helps you build a SQL connection string, eliminating the need to research and memorize all the available options. Connection String Generator creates a ready-to-use connection string that allows a third-party application to connect to a SQL Server instance. Each generated connection string contains only those options you selected, arranged in the correct order with the correct syntax.

The Connection String Generator puts all the connection string options at your fingertips. Simply specify the desired options and the tool automatically builds the corresponding connection string. Once you build the connection string, you can test it within the tool and then copy it to your code.

How the Connection String Generator tool works

The Connection String Generator offers over 40 editable connection string options to build your custom string. The following table includes the basic connection string options available when using the Connection String Generator tool.

Property	Description
SQL Server	<i>Required.</i> Specifies the name or network address of the SQL Server instance to which you want to connect. To specify a local unnamed instance, type "(local)".
TCP/IP Port	<i>Required when the target SQL Server does not use the default port of 1433.</i> Specifies the port on which the SQL Server listens.
Database or Initial Catalog	Specifies the database or catalog to which you want to connect.
Application Name	Specifies the name of the application that you want to connect to the SQL Server instance. Use this option to identify your application in audit trails. <i>If you are using ADO and do not specify an application name ,</i> Connection String Generator uses the default string '.NET SQLClient Data Provider'.
Connection Timeout	Specifies the amount of time in seconds you want to wait before the connection times out due to a failure or error. This option is set to 15 seconds by default.
Encrypt Data	Enables SSL encryption for all data sent between client and server. This option requires a certificate installed at the server.
Connect Using	Specifies the credentials to use when connecting to the SQL Server instance. Choosing Windows authentication use the current Windows login credentials whereas choosing SQL authentication lets you specify a SQL login and password.

The following table includes the connection pooling options available when using the Connection String Generator tool.

Property	Description
Use Connection Pooling	Enables connection pooling. When enabled, the connection is drawn from the pool or created and added to the pool. This option is enabled by default.
Max Pool Size	Specifies the maximum number of connections available in the connection pool. This option is set to 100 by default.
Min Pool Size	Specifies the minimum number of connections available in the connection pool. This option is set to 0 by default.
Connection Lifetime	Specifies the lifetime of a connection in seconds. This option indicates whether the connection is destroyed or kept alive when returned to the connection pool. Specifying 0 gives the connection the maximum allowed timeout.
Reset Database Connection	Specifies whether the connection is reset when drawn from the connection pool.

The following table includes the client network library options available when using the Connection String Generator.

Property	Description
Client Network Library	Specifies alternate network protocols to use during the connection. Named pipes is used by default.

The following table includes the advanced connection string options available when using the Connection String Generator.

Property	Description
Attach DB File Name	Specifies the full path for the primary database file of the database. This option ensures that the database is attached before making the connection.
Current Language	Specifies the SQL Server Language record name.
Driver	Specifies the driver to be used for the connection.
Packet Size	Specifies the size in bytes of network packets sent between the client and server. This option is set to 8k by default.

Property	Description
Network Library	Specifies the network library used to establish the connection.
Workstation ID	Specifies the name of the workstation connecting to the SQL Server instance.
Database Mirroring Fail Over Partner	Specifies the name of the fail over partner server used for database mirroring.
Asynchronous Processing	Enables asynchronous support. This option is disabled by default.
MARS Connection	Enables the ability to an application to maintain multiple active result sets. This option is available for .NET version 2.0 and above only.
Persist Security Info	Enables returning security-sensitive information, such as a password, returned as part of the connection if the connection is open or is ever open.
Enlist	Enables the SQL Server connection pooler to automatically enlist the connection in the current transaction of the creation thread.

Use the Connection String Generator tool

To generate a connection string:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Connection String**.
2. Specify the SQL Server instance to which you want to connect. This field is the only required field to generate a connection string.
3. Specify any other options you want to include in your string, such as the default database, alternate credentials, or your application name.
4. Click **Test Connection** to ensure you specified the correct options for the connection string.
5. ***If the test connection is successful***, copy the connection string to paste into your own code by clicking **Copy to Clipboard**.

Job Editor

The Job Editor tool helps you easily manage your SQL Server jobs by performing bulk updates across your environment. For example, you can sort jobs by status and property, such as whether the job is enabled and who is the owner. You can then bulk update the owners to a new login. The Job Editor tool allows you to:

- Check job settings and last run status across multiple SQL Server instances.
- View summary of key job statistics.
- Sort jobs by status or property, such as category, owner, or last run time.
- Bulk update job properties on multiple jobs across servers.
- Save job information.

How the Job Editor tool works

When you edit jobs that exist on different SQL Server instances, you can choose from the values currently available on the selected instances.

This restriction affects the general job properties, such as the job owner, and the notification settings. Be aware that SQL Server has dependencies between the notification types, levels, and operators. For example, you cannot change the notification level if an operator was not previously assigned to the job. For more information about managing notifications, see Microsoft SQL Server Books Online.

By default, Job Editor [connects](#) to the specified SQL Server instances using the credentials of your currently logged on Windows account, and then collects job statistics for all jobs deployed on these instances.

Use the Job Editor tool

To use the Job Editor to view jobs:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Job Editor**.
2. Specify the SQL Server instance or server group that contains the jobs you want to edit. To specify multiple SQL Servers, separate each instance name with a semicolon.
3. *Optional.* Filter based on a timeframe by selecting the number of days, hours, and minutes in the **Jobs not run in the last** field.
4. Click **Get Job Information**.
5. View the results.
6. **If you want to save the results**, copy the list to clipboard, or export the list as an XML or CSV file.

To use Job Editor to edit jobs:

1. Get the job information from the target SQL Server instances.

2. Select the jobs you want to edit. To select all jobs that match a specific search criteria, click **Check Jobs by Property Values**.
3. Click **Update Checked Jobs**.
4. On the Update Properties window, change the appropriate properties, and then click **Update**.


Linked Server Copy

The Linked Server Copy tool allows you to copy your linked servers from one SQL Server instance to another along with the associated settings and permissions, saving you the time it would take to move linked servers manually. With this tool you can:

- View all the linked servers that exist on the source server and not on the destination SQL Server instance.
- Copy all or specific linked servers from your source SQL Server instance to a destination SQL Server instance.

How the Linked Server Copy tool works

You can copy all linked SQL Server instances. If the remote authentication on a SQL Server instance is SQL Server authentication, must provide the password. You can copy all other linked servers without any additional steps.

 If the type of provider is not compatible with the host, the copied linked server does not work correctly. For example, a jet provider (32-bit only) copied to a 64-bit system does not work.

Use the Linked Server Copy tool

To copy over a list of linked servers:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Linked Server Copy**.
2. Select the source and destination SQL Server instances.
3. Click **Preview Missing Linked Servers**.
4. Either select the individual linked servers you want to copy or check the box at the top of the list to select all linked servers. All the linked servers that include a green Y in the **All Passwords Moveable** column are copied without any additional steps. Linked servers that include a red N require you to supply a password to copy.
5. Click **Copy Linked Servers**.

To use the Specify Passwords window:

1. Select a SQL Server instance from the **Linked Servers** column.
2. Select the logins you want to copy from the list and enter the associated passwords.
3. Repeat this for every SQL Server instance in the **Linked Servers** column.
4. Click **OK** to copy the linked servers.


 Logins that exist for the Linked Server on the source SQL Server instance are copied to the destination SQL Server instance.

Table Pin

i This tool only works for SQL Server 2000 instances since the Pinned Table feature was removed in SQL Server 2005 and later.

The Table Pin tool allows you to manage pinned SQL Server tables. Pinning tables into SQL Server memory can provide performance boosts when SQL Server accesses those tables. Despite the performance benefit, this feature can become dangerous and cause serious performance issues when misused. With this tool, you can:

- Monitor exactly which tables are pinned and how much memory they consume from the SQL Server memory pool
- Pin or unpin tables and easily set up the proper configuration

How the Table Pin tool works

By default, Table Pin [connects](#) to the specified SQL Server instances using the credentials of your currently logged on Windows account, and then determines which database tables are pinned to the SQL Server memory.

Use the Table Pin tool

To pin a table:

1. Specify the SQL Server whose table status you want to view.
2. Clear the **Hide tables that are not pinned** option.
3. Click **Get Pinned Status**.
4. Select the table you want to pin, and then click **Pin Table**. The action occurs immediately.

To view a pinned status:

1. Specify the SQL Server whose table status you want to view. ***If you want to see the status of all tables on this SQL Server instance***, clear the **Hide tables that are not pinned** option.
2. Click **Get Pinned Status**.
3. ***If you want to save the results***, copy the list to clipboard, or export the list as an XML or CSV file.

To unpin a table:

1. Specify the SQL Server whose table status you want to view.
2. Click **Get Pinned Status**.
3. Select the table you want to unpin, and then click **Unpin Table**. The action occurs immediately.

Analytic tools

The IDERA SQL Admin Toolset contains the following analytic tools:

Index Analyzer

Analyze the usage patterns of your indexes, including those used most often used and those never used.

Inventory Reporter

Show inventory and configuration details for your SQL Server instances and their host computers.

Patch Analyzer

Find out if your SQL Servers are up to date with the latest available patches.

Server Ping

Continuously monitor the status and availability of your SQL Servers.

Server Statistics

View statistics and counts about your SQL Servers.

Space Analyzer

Analyze the space usage of your SQL Server by drive or database.

Inventory Reporter

The Inventory Reporter tool helps you report on inventory and configuration details about your SQL Server instances and the host computers. With this tool, you can view and compare inventory data more easily using inventory reports that include:

- a comprehensive inventory
- SQL Server properties
- SQL Server version details
- computer properties

You can also capture inventory data for one or more SQL Server instances. Use the Inventory Reporter tool to compare configuration across multiple SQL Server instances or a server group and highlight the differences in each inventory. After capturing the data, you can save the data as a baseline snapshot to compare against future inventories or other instances.

How the Inventory Reporter tool works

Inventory Reporter also features various export options so you can use this data in other programs.

Before using this tool, it is important that you know that Inventory Reporter uses the credentials of currently logged on Windows account to make a WMI connection to the computers hosting your SQL Server instances. To successfully complete the connection, your user account must have permission to use the WMI protocol.

By default, Inventory Reporter **connects** to the specified SQL Server instances and the host computers using the credentials of your currently logged on Windows account, collecting configuration information and then generating the inventory report.

Use the Inventory Reporter tool

To generate an inventory report:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Inventory Reporter**.
2. ***If you want to discard any previously collected data***, click **Clear**.
3. Click **Capture Live Data**.
4. Specify the SQL Server instance or server group that contains the logins whose passwords you want to check, or click the browse button to select them. To specify multiple SQL Servers, separate each instance name with a semicolon.
5. Click **OK**.
6. To view the data in a grid format, click **Show Servers As Rows** on the View menu.

7. ***If you want to create a baseline snapshot using this inventory data*** , click **Save Baseline**.
8. ***If you want to save the results*** , copy the list to clipboard, or save the results as an XML or CSV file.

To compare the inventory of multiple instances:

1. Generate an inventory report, specifying which instances you want to compare against each other.
2. Click **Highlight Differences**.
3. Review the properties and settings highlighted with a yellow bar.

To compare the current inventory against a baseline snapshot:

1. Generate an inventory report, specifying which instances you want to compare against the baseline snapshot.
2. Click **Open Snapshot**.
3. Select the snapshot you want to compare against, and then click **OK**.
4. Click **Highlight Differences**.
5. Review the properties and settings highlighted with a yellow bar.

Patch Analyzer

- i** You must run the following SQL Admin Toolset tools under the Administrator account for proper functionality:
- Patch Analyzer
 - [User Clone](#)
 - [Partition Generator](#)

The Patch Analyzer tool helps you manage SQL Server patches and keep your environment properly patched. Specify your managed SQL Servers and Patch Analyzer tells you if they are running a supported version of SQL Server. In addition, you can get information about the patches running on your SQL Server and whether additional hotfixes are available for your SQL Server level.

How the Patch Analyzer tool works

Patch Analyzer allows you to:

- Verify that your SQL Servers instances are patched to the same level and/or the appropriate level.
- Identify whether any SQL Server instances are running unsupported versions of SQL Server (or versions whose support is about to expire).
- Use quick links to Microsoft Knowledge Base articles to get more information about applied hotfixes and new hotfixes.
- Keep up with the latest patches by downloading the most recent list compiled by IDERA.

By default, Patch Analyzer [connects](#) to the specified SQL Server instances using the credentials of your currently logged on Windows account.

Use the Patch Analyzer tool

To use Patch Analyzer:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Patch Analyzer**.
2. Specify the SQL Server instance or server group whose SQL Server versions you want to check. To specify multiple SQL Servers, separate each instance name with a semicolon.
3. Click **Load Version Information**. Review the results and note the highlighted versions that are no longer supported or are soon becoming unsupported.
4. **If you want to group the results by SQL Server version**, click **Show Groups** on the View menu.
5. **If you want to see more information about a hotfix that was applied to your SQL Server instance**, select the instance from the results list, and then click **View the Microsoft Knowledge Base Article**.

6. ***If you want to see which new hotfixes are available***, select the SQL Server instance you want to update from the results list, and then click **View Available Updates**.
7. ***If you want to save the results***, copy the list to clipboard, or save the results as an XML or CSV file.

To use find SQL Server downloads:

1. Click **SQL Server Downloads / Support Lifecycle**.
2. Find the download you need, and then click the link to proceed to the corresponding Microsoft Web site page.

To review which patches are checked:

1. Click the **Display Build List** link. ***If you are not sure you have the latest list***, click **Check for New List**. This link downloads the most recent list of releases, service packs, and hotfixes for SQL Server software.
2. Review the list of all known releases of SQL Server software.
3. To see the Microsoft Knowledge Base article for a specific hotfix, select the hotfix, and then click **View Microsoft Knowledge Base Article**.

Index Analyzer

The Index Analyzer tool helps you evaluate SQL Server indexing. With this tool, you can view indexes contained in a specific table, or view index properties and performance statistics across a SQL Server instance.

By default, Index Analyzer **connects** to the specified SQL Server instances using the credentials of your currently logged on Windows account, and then collects information and performance statistics about each index.

Understand selectivity

Selectivity is a critical statistic for analyzing index performance. However, retrieving selectivity from SQL Server is a time-intensive operation. To analyze selectivity across all indexes on the specified instances, click **Load Selectivity while loading Statistics** before you begin. To view selectivity values on a few indexes, load the other index statistics first, then select the indexes you want to analyze and click **Load Selectivity**.

How the Index Analyzer tool works

The following index statistics are available for viewing in the Index Analyzer tool results:

% Rows Modified

Provides the percentage of rows modified since statistics were last updated.

If this percentage is high, then SQL Server is using outdated information to make optimization decisions. A high percentage can negatively impact system performance.

You can reset this value by updating the index statistics.

% Updates to Total Access

Percent of Updates to Total Accesses. If an index is updated often, then the usefulness as an index is offset by the overhead of keeping the index updated.

Columns

Indicates which columns were affected by the statistic.

Clustered

Indicates whether the index is clustered.

Database

Provides the name of the database containing the index.

Disabled

Indicates whether the index is disabled.

Fill Factor

Provides the Fill Factor used to create the index.

Index

Provides the name of the index.

Index Usefulness

Provides an estimate of how likely it is that SQL Server uses the index based on the following criteria: Selectivity, % Rows Modified, and % Updates to Total Accesses.

- Low usefulness indicates that the index is disabled, or one or more of the criteria are at critical levels
- Medium usefulness indicates that one or more of the criteria are at warning levels
- High usefulness indicates that all criteria are at acceptable values

Last Statistics Update

Indicates when SQL Server last updated statistics for the index.

Lookups

Provides the number of lookups performed on the index.

Pages

Provides the number of 8K pages used by the index.

Rows

Provides the number of rows in the index.

Rows Modified Since Last Update

Provides the number of rows modified since SQL Server last updated these statistics.

Scans

Provides the number of scans performed on the index.

Seeks

Provides the number of seeks performed on the index.

Selectivity

Indicates the statistical uniqueness of each row of the index. Selectivity is calculated from the index density stored by SQL Server (Selectivity = 1 - Density). Low Selectivity may negatively impact system performance.

If no statistics are stored in SQL Server, the value for this column is "No Statistics Available".

Size

Provides the physical size of the index in bytes.

Table

Provides the name of the table containing the index.

Total Accesses

Sums the total number of times the index is accessed (Seeks + Scans + Lookups + Updates).

Updates

Provides the number of updates to the index.

The following column statistics are available:

Columns

Indicates which columns were affected by the statistic.

Database

Provides the name of the database from which the statistic was generated.

Modified Rows

Provides the number of rows that were updated, inserted, or deleted.

Selectivity

Indicates the statistical uniqueness of each row of the statistic. Low Selectivity may negatively impact system performance.

Statistic

Provides the name of the SQL Server statistic calculated for this column.

Table

Provides the name of the table from which the statistic was generated.


Use the Index Analyzer tool

To use the Index Analyzer:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Index Analyzer**.
2. Specify the SQL Server instance whose indexes you want to check.
3. Check **Load Selectivity while loading Statistics** to load selectivity statistics now.
4. Click **Load Statistics**.
5. View the index data displayed on the **Index Statistics** tab. To sort the indexes, use the **View Options** in the right pane.
6. To view details about the index columns, click the **Column Statistics** tab.
7. To view index data for a specific table, expand the instance node in the tree pane, and then click the appropriate table.
8. To recalculate the statistics using a sample percentage of rows, click **Update Statistics (using Sampling)**. To recalculate statistics using all rows, click **Update Statistics (using Full Scan)**.
9. ***If you want to save the results***, copy the list to clipboard, or save the results as an XML or CSV file.

Space Analyzer

The Space Analyzer tool helps you identify the disk space usage for SQL Server instances across your enterprise. With this tool, you can easily and proactively anticipate when you need to bring new instances online, plan for new databases, and authorize hardware upgrades.

 The Space Analyzer tool does not support mirrored databases. You can view disk space usage for your principal database, but not for any mirrored database.

How the Space Analyzer tool works

Space Analyzer allows you to view SQL Server user database storage statistics or view disk space usage statistics, such as the amount of available space on hard drives across computers that are hosting SQL Server instances. You can also discover which computers or SQL Server instances need immediate attention by quickly viewing the summary status.

The Space Analyzer tool displays the following percentages:

- % Used shows how much of a file is being used
- % Disk Used shows how much of the whole drive is being used
- % Used by Database Files shows how much of that drive is being used by that file

By default, Space Analyzer [connects](#) to the specified SQL Server instances using the credentials of your currently logged on Windows account, and then collects size and usage information about each database.

Use the Space Analyzer tool

To use Space Analyzer:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Space Analyzer**.
2. Specify the SQL Server instance or server group whose disk space usage you want to view. To specify multiple SQL Servers, separate each instance name with a semicolon.
3. Click **Load Space Statistics**.
4. View the results for each disk drive and SQL Server instance hosted by the corresponding computer. For more details, select the disk drive, instance, or user database in the tree, and view the statistics grid in the right pane.
5. *Optional.* Clear the **Use WMI Credentials** check box if you do not want the user to present WMI credentials before accessing specific server instances.
6. **If you want to save the results**, click **Copy Results To Clipboard**, or save the results as an XML or CSV file.

Server Statistics

The Server Statistics tool helps you determine the health of your instances by collecting real-time SQL Server statistics, such as the number of jobs assigned to the SQL Server Agent, and performance counters.

How the Server Statistics tool works

The Server Statistics tool searches the selected SQL Servers of server group, and displays a comprehensive list of real-time statistics for many components including your:

- Backup devices
- Databases
- Error Logs
- Locks
- Logins
- Performance counters
- Processes
- SQL Server Agent errors

By default, Server Statistics [connects](#) to the specified SQL Server instances using the credentials of your currently logged on Windows account, and then collects performance and diagnostic statistics about each instance.

Use the Server Statistics tool

To use Server Statistics:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Server Statistics**.
2. Specify the SQL Server instance or server group whose statistics you want to check. To specify multiple SQL Servers, separate each instance name with a semicolon.
3. Click **View Statistics**.
4. View the results. To see details, expand a node in the statistics tree, and select a data point.
5. ***If you want to save the results***, copy the list to clipboard, or save the results as an XML or CSV file.

Server Ping

The Server Ping tool helps you monitor the status of SQL Server instances across your enterprise by routinely pinging specified instances. You can run Server Ping as centralized interface or as system tray service alerts you when instance goes offline or becomes unhealthy.

How the Server Ping tool works

By default, Server Ping **connects** to the specified SQL Server instances using the credentials of your currently logged on Windows account.

Server Ping then uses either a WMI or SQL query to collect information such as whether the SQL Server services are running or when an instance comes online or goes offline.

Server Ping allows you to:

- Monitor a SQL Server instance or a server group
- Run the tool in the system tray to:
 - Use status icons to quickly ascertain whether a server is online
 - Use pop-up alerts to monitor when instances come online or go offline
 - Launch tool when the Windows operating system starts
- Refresh data automatically at specified intervals
- Avoid unnecessary alerts by selecting which instances should be ignored during maintenance periods
- Choose WMI or a SQL query to collect health information

Use the Server Ping tool

To use the Server Ping tool:

1. Open the IDERA SQL Admin Toolset Launchpad, and then click **Server Ping**.
2. Specify the SQL Server instance or server group that contains the logins whose passwords you want to check. Use the following steps to specify more than one instance:
 - a. Click **Add**.
 - b. On the Add Server window, type the name of each instance, separating multiple instances with semicolons.
 - c. Specify the type of authentication you want Server Ping to use to connect to the specified instances. By default, the credentials of your current Windows account are used.
 - d. To verify that the connection settings are valid for the specified instances, click **Test Connection**.
 - e. Indicate whether you want to receive overall status information or alerts and notifications about the specified instance.

- f. Click **OK**.
3. Select the appropriate system tray options.
4. Choose whether you to run a WMI or SQL query to retrieve the status of the specified instances. You can specify which SQL command you want to run in the query.
5. Click **Check Servers Now**.

SQL Admin Toolset PDF

This page contains a direct link to the IDERA SQL Admin Toolset help in PDF format. This format is suitable for printing and for saving on your local PC for further reference. The PDF includes all pages from the relevant help published on wiki.idera.com.

- [IDERA SQL Admin Toolset 1.9.4 PDF](#)
- [IDERA SQL Admin Toolset 1.9.3 PDF](#)
- [IDERA SQL Admin Toolset 1.9.2 PDF](#)
- [IDERA SQL Admin Toolset 1.9.1 PDF](#)
- [IDERA SQL Admin Toolset 1.9 PDF](#)