

The MITRATECH logo consists of the word "MITRATECH" in a white, uppercase, sans-serif font, centered within a solid teal rectangular background.

MITRATECH

TeamConnect® Data Warehouse 5.0

© 2016 Mitrtech

Release Notes

Data Warehouse 5.0 Release Notes

Document ID: dw_5_0_rn_1, published on 12/2/2016

Copyright © 2016, Mitrtech Holdings, Inc. All rights reserved.

Disclaimer of Warranty

Mitrtech Holdings, Inc. (Mitrtech) makes no representations or warranties, either expressed or implied, by or with respect to anything in this document, and shall not be liable for any implied warranties of merchantability or fitness for a particular purpose or for any indirect, special or consequential damages.

Mitrtech reserves the right to not support non-standard or non-default functionality and extended functionality available in third-party software, unless specifically documented as supported or certified in the Mitrtech product documentation. For further information regarding third-party non-standard or non-default functionality, please contact Mitrtech Support.

This document, along with the software that it describes, is furnished under license and may be used or copied only in accordance with the terms of such license. The content of this document is furnished for informational use only, is subject to change without notice, and should not be construed as commitment by Mitrtech.

The following document is for the TeamConnect™ Data Warehouse 5.0 release only. Though every effort was made to ensure that the information in this document is correct and reliable, Mitrtech does not assume any liability for any errors encountered in this document.

If you need support for TeamConnect™ Data Warehouse 5.0, please contact the Mitrtech support team by sending an email to: support@mitrtech.com. For more information about Mitrtech, visit our web site: <http://www.mitrtech.com>.

"Mitrtech", TeamConnect™ Enterprise, TeamConnect™ Legal, TeamConnect™ Legal Matter Management, Collaborati®, TeamConnect™ Collaborati Spend Management®, TeamConnect™ Deadlines, TeamConnect™ AP Link, TeamConnect™ Office Suite, TeamConnect™ Legal Reports, TeamConnect™ SOP Manager, TeamConnect™ Financial Management, TeamConnect™ Screen Designer, and TeamConnect™ Upgrade Toolkit are trademarks and products of Mitrtech Holdings, Inc. All other products or services mentioned in this book are the trademarks or service marks of their respective companies or organizations.

GOVERNMENT RIGHTS LEGEND:

Use, duplication or disclosure by the U.S. Government is subject to restrictions set forth in the applicable Mitrtech license agreement and as provided in DFARS 227.7202-1(a) and 227.7202-3(a) (1995), DFARS 252.227-7013 (Feb 2012) FAR 12.212(a) (1995), FAR 52.227-19 (Dec 2007), or FAR 52.227-14, as applicable.

CONTACT US:

Mitrtech Holdings, Inc.
5001 Plaza on the Lake Suite 111, Austin, TX 78746
Phone: (512) 382-7322

NOTE: Throughout Mitrtech product publications, in addition to using full product names where necessary, we also use familiar and shorter terms to increase your ease of reading. You may find the following aliases for our product names:

TeamConnect for TeamConnect Enterprise
Matter Management for TeamConnect Legal Matter Management
TeamConnect Legal for TeamConnect Legal Matter Management
CSM for TeamConnect Collaborati Spend Management
Collaborati Spend Management for TeamConnect Collaborati Spend Management
SOP or SOP Manager for TeamConnect SOP Manager
Legal Hold for TeamConnect Legal Hold
Legal Reports for TeamConnect Legal Reports
Deadlines for TeamConnect Deadlines
AP Link for TeamConnect AP Link
Office Suite for TeamConnect Office Suite
Financial Management for TeamConnect Financial Management
Screen Designer for TeamConnect Screen Designer
Upgrade Toolkit for TeamConnect Upgrade Toolkit

Acknowledgements

This product includes software developed by the following organizations:

Apache Software Foundation (<http://www.apache.org/>)

OpenSymphony Group (<http://www.opensymphony.com/>).

The license agreements for these and other supplemental software packages can be found in your installation media in subfolder Supplemental_Software_Licenses. That subfolder also contains Open Source Components.pdf, which lists the locations, license types, and specific versions of components that are available on the web.

Table of Contents

| | |
|--|----------|
| Part I Data Warehouse Release Notes | 5 |
| 1 System Requirements | 5 |
| 2 Known Issues | 5 |
| 3 Changes and Enhancements | 7 |
| 4 Upgrade Considerations | 10 |

1 Data Warehouse Release Notes

| |
|--|
| System Requirements |
| Changes and Enhancements |
| Upgrade Considerations |
| Known Issues |

1.1 System Requirements

Data Warehouse uses Pentaho tools (Pan, Kitchen, and Spoon). Pentaho tools formerly were known by the collective name "Kettle". It has the following system requirements for installation and configuration:

- TeamConnect as the source database
- A target database. Data Warehouse supports the same database servers that are supported for TeamConnect (Oracle 11g or Oracle 12c; SQL Server 2008 or SQL Server 2012). The database type and version selected should be the same as what's used for the respective TeamConnect database.
- Refer to the *TeamConnect Release Notes* for your respective version for further details.
- Java Runtime Environment version 1.7.
 - Oracle users will need the ojdbc6.jar file.
 - SQL Server users will need the sqljdbc4.jar file.

Both of these .jar files need to be put in the \libext\JDBC\ directory. These files are available from their respective Oracle and Microsoft web sites.

- Ensure that the kettle.properties file has the WH_COMMIT_SIZE=5000 field present. **Data Warehouse will not function without this field present.**
- Extra disc space requirements for temporary tables. (Extra multiplier).

More information can be found here.

Note: Data Warehouse has been certified only with standard network authentication.

1.2 Known Issues

The following items are known TeamConnect Data Warehouse 5.0 issues and limitations. Each known issue is documented in the following format:

- A description of the issue

- A workaround, if available
- Internal tracking code

Issue: Updating/inserting documents in a large database is very slow.

Workaround: None.

Tracking code: DWE-2484

Issue: Deleting a large number of contact records is slow.

Workaround: None.

Tracking code: DWE-2460

Issue: Records with timestamps equal to maximum modifiedOn timestamp are not included in initial load.

Workaround: Run a Refresh job following initial load to add these records.

Tracking code: DWE-2226

Issue: Initial load fails or is terminated if interrupted during the Workflow entity.

Workaround: Edit the WH_LAST_REFRESH table and set the IS_INITIAL_COMPLETE field to zero for all fields. Initial load will then run from the beginning. This action must also be completed if user wants to re-run initial load after successful completion.

Tracking code: DWE-1670

Issue: Overall audit count is inaccurate if record is created and then deleted between refreshes.

Tracking code: DWE-2303

Workaround: The overall audit count will correct itself after the next refresh. This will purge the record from the WH_REMOVED table that is causing the discrepancy.

Issue: Refresh audit count is inaccurate if record is created and then deleted between refreshes.

Tracking code: DWE-2083

Workaround: The overall audit count will correct itself after the next refresh. This will purge the record from the WH_REMOVED table that is causing the discrepancy.

Issue: Some commit sizes are static, don't obey the Kettle properties COMMIT_SIZE variable.

Workaround: None. Some commit sizes are fixed by Pentaho and cannot be changed.

Tracking code: DWE-2492

Issue: All warehouse components are not deleted by removal script.

Workaround: Manually delete from the database. Deleting is optional. This does not affect the ability to reinstall the application.

Tracking code: DWE-2426

Issue: Installation creates database tables not needed by the app (staging, temp).

Workaround: Manually delete from the database. Deleting is optional. This does not affect the ETL jobs.

Tracking code: DWE-2501

1.3 Changes and Enhancements

Data Warehouse 5.0 includes the following changes and enhancements. For more detail, please see the *Data Warehouse FAQ* and *Data Warehouse Help* guide.

For users accessing this guide online, click the headline for each enhancement to toggle additional details.

▣ **Stability improvements. ETL transformations have been re-sequenced to reduce the potential for SQL deadlocks.**

Previous releases executed many SQL instructions in parallel, which sometimes caused delays, timeouts and deadlocks. The new release removes multiple copies from input steps and separates them, resulting in fewer potential conflicts. The default database commit sizes were increased, replacing batch sizes. In addition, the initial load job was streamlined to remove unnecessary steps it shared in common with the refresh job. The net result of these changes should be fewer job failures, and fewer incidents of jobs running far longer than expected.

▣ **Prevent refresh job collisions. A new refresh job will be prevented from launching if the previous one is still running.**

Past releases did not check if an old refresh job was still running before launching a new one. It was possible to accidentally have two or more refresh jobs running simultaneously, risking lockups and database integrity. The new release creates lock files and also checks the operating system. If it finds that a refresh process is already running, it will skip the current refresh job and terminate. So long as each refresh job is launched from the same machine, the standard wrapper script (.bat or .sh) will ensure that only one job runs at a time.

▣ **Discrete time boundaries. Each refresh job collects only records created or modified between the last refresh and current refresh starting times. Boundaries don't overlap, which improves auditability and avoids redundant data collection for records that haven't changed.**

As in earlier releases, the refresh job scans the modifiedOn timestamps of each TeamConnect record to see if the records are new or newly updated. However, the minimum and maximum time boundaries used to determine what's "new" are computed differently in this release. Each job has boundaries adjacent to the previous successful job. These values are stored in the WH_LAST_REFRESH table. Only records whose modifiedOn times are greater than or equal to the lower boundary or less than the upper boundary are eligible. The lower boundary of the current job is simply the upper boundary of the previous successful refresh job. In other words, the current refresh picks up from where the previous refresh left off. This boundary is uniform across all entities.

The upper boundary is computed from two values. First, it uses the start time of the current job. Then the upper boundary is adjusted by subtracting an offset from the current start time. For example, consider a job scheduled to run at the top of each hour. When it starts at 15:00 with an offset of 300 seconds, it will only collect records modified through 14:55. The user can change the offset by editing the COMMIT_DELAY value in the kettle.properties configuration file.

The purpose of the offset is to ensure that records saved in TeamConnect shortly before the job start time aren't lost even if they aren't committed to the database until after. In this example, a record modified at 14:57 will not be updated in the warehouse during the 15:00 job run. However, it will be included in the 16:00 run. Because the upper boundary for each job becomes the lower boundary for the next, the offset changes the minimum as well as maximum timestamps that qualify for each refresh. The 15:00 job run would include records $\geq 13:55$ and $< 14:55$; the 16:00 run would filter records between 14:55 and 15:55; and so on.

▣ **Refresh audit table. This new audit table maintains an automatic history. Each refresh job logs how many of each main entity were transferred, starting/ending times, and discrepancies between expected and actual counts.**

Every time the ETL Refresh job runs, it adds information about to the AUDIT_TBL database table.

A row is appended for each main entity: User, Group, Contact, Project, Involved, Account, Appointment, Document, Expense, History, Invoice and Task. An entry contains these columns:

- ENTITY: The name of the entity.
- RUN_NUMBER: An incremental integer assigned to each job, starting at 1.
- TC_TABLE: The TeamConnect database table that is the primary data source for the transformation.
- DW_TABLE: The corresponding warehouse target table populated from the source table.
- TC_COUNT_UPDATE: The number of source records collected for this refresh that were updated in TeamConnect since the last refresh. These records will be updated in the warehouse.
- TC_COUNT_NEW: The number of source records that were new in TeamConnect. These records will be added to the warehouse target.
- TC_COUNT_DELETED: The number of source records that were deleted from TeamConnect. These records will be deleted from the warehouse target. The deleted count is obtained from the WH_REMOVED_RECORDS table.
- TC_DELTA: The net increase or decrease in the number of source records, calculated as TC_COUNT_NEW minus TC_COUNT_DELETED.
- DW_COUNT_BEFORE: The count of target records in the warehouse at the start of the refresh job.
- DW_COUNT_AFTER: The count of target records in the warehouse at the conclusion of the refresh job.
- DW_DELTA: The net increase or decrease in the number of target records, calculated as DW_COUNT_AFTER minus TC_COUNT_BEFORE.
- TC_DW_DIFF: The difference (discrepancy) between the net change in the number of TeamConnect records and the net change in the number of warehouse records (TC_DELTA minus DW_DELTA).

- ENTITY_START. The time that the Refresh job began collecting and populating the entity.
- ENTITY_COMPLETE. The time that the Refresh job finished collecting and populating the entity.
- RUN_TIME_MINUTES. The number of elapsed minutes for the entity to populate (ENTITY_COMPLETE minus ENTITY_START).

The refresh audit table will grow over time. The application does not delete or overwrite old records. If you wish, you may safely delete records from the table when the refresh job is not running.

▣ **Overall audit count comparison. The user can run a top-level audit at any time, comparing TeamConnect database to warehouse totals.**

The user can compare counts at any time between the TeamConnect database and the warehouse, by running the audit.bat (Windows) or audit.sh (Linux) script. This command will collect the counts in the AUDIT_COUNT database table, and also export them to a text file. The table includes these columns:

- ENTITY_NAME: The name of the entity.
- TC_COUNT: The number of entity records in TeamConnect.
- TC_REMOVED_COUNT. The number of entity records that were deleted from TeamConnect since the last fresh, obtained from the WH_REMOVED_RECORDS table.
- TC_TOTAL_COUNT: The adjusted TeamConnect entity count for comparison with the warehouse. Records tracked in WH_REMOVED_RECORDS have not yet been deleted from the warehouse. Therefore, this field equals TC_COUNT plus TC_REMOVED_COUNT, the expected warehouse count if the two databases are in balance.
- DW_COUNT: The actual number of entity records in the warehouse.
- DIFF_TC_MINUS_DW. The difference (discrepancy) between the adjusted TeamConnect and warehouse counts.

▣ **Notifications with logs. The log file is now included as an attachment to all e-mails.**

The user will receive the log file as an e-mail attachment, along with any e-mail notification that the job has completed, provided that SMTP e-mail has been configured in the Kettle properties. The log will be included for either initial load or refresh, whether or not the job succeeds. The notification is generated as a separate Pentaho job following the completion of the ETL job. This ensures that the notification is sent so long as the ETL job returns an error code.

▣ **Resume initial load. If the initial load job fails or is interrupted before it finishes, it won't start over at the beginning. When the job is re-launched, it will skip over entities that have already populated successfully.**

If an initial load terminates before it completes -- for example, because of a network interruption -- it can be resumed from near the point of failure. Simply start the command again (TeamConnect_Warehouse_initial.sh or TeamConnect_Warehouse_initial.bat). The job will begin

with the most recent main entity it was processing, skipping over entities that have already finished populating.

The job uses the IS_INITIAL_COMPLETE column in the WH_LAST_REFRESH table as a flag field to track which entities have already succeeded. The value is changed from zero to one once the entity is complete.

▣ **Resume refresh after failure. Likewise, a refresh that is partially complete can be restarted and will skip over entities that are already complete.**

A refresh that fails can also be restarted and will skip over entities that are already complete.

Unlike initial load, the process is not automatic. The user must run a different script to resume the refresh: TeamConnect_Warehouse_Refresh_Resume.bat (Windows) or

TeamConnect_Warehouse_Refresh_Resume.sh (Linux).

The refresh resume feature is provided as an optional command because the standard refresh cycles will eventually correct for failures. Any records that were not populated because of a failed refresh will be collected during the next refresh (the time boundaries only advance when a refresh succeeds). The resume script is helpful when the refresh is expected to take a long time, or the failed refresh was on the verge of completing.

The resume refresh job uses the IS_REFRESH_COMPLETE column in the WH_LAST_REFRESH table as a flag field to track which entities have already succeeded. The value is changed from zero to one once the entity is complete.

1.4 Upgrade Considerations

The Data Warehouse enhancements in this guide are available for 3.4SP1, 4.x, and 5.0

TeamConnect Enterprise users. However, each version has its own respective Data Warehouse version.

- Data Warehouse 4.1 is for TCE 4.x users
- Data Warehouse 3.5 is for TCE 3.4SP1 users
- Data Warehouse 5.0 is for TCE 5.0 users

The only direct upgrade path to version 4.1 is from Data Warehouse version 4.0. If you have a version of Data Warehouse earlier than 4.0, you have must remove your existing Data Warehouse (see Removing the Data Warehouse), then recreate it using the instructions in the New Installations section of the Data Warehouse guide.

Important: Data Warehouse should not be upgraded until TeamConnect is already at your desired version. Ensure that the TeamConnect version is correct before proceeding.

To upgrade from version 4.0, from the installation media, open directory **TeamConnect_DataWarehouse** and copy the contents to a target directory on your hard drive. (For purposes of this documentation, we will presume that your target directory is named **datawarehouse** and it is a subdirectory of your hard drive's root directory.)

1. Ensure that file **kettle.properties** contains the correct connection information for your Data Warehouse database. Details about connection properties are found in step 3 on New Installations.
2. Modify **shared.xml** and **kettle.properties**, as described in step 3 on New Installations. Then return to this procedural.
3. For SQL Server and NTLM authentication only, add two new attributes to each affected **connection** element in file **shared.xml**. See the highlighted text in the example below. This example deals with the source database connection, but NTLM can be used on source and/or target database connections.

```
<connection>
<name>Source_TeamConnect_data</name>
<server>${jdbc.source_TC_hostname}</server>
<type>MSSQL</type>
<access>Native</access>
<database>${jdbc.source_TC_dbname}</database>
<port>${jdbc.source_TC_portnumber}</port>
<username>${jdbc.source_TC_username}</username>
<password>${jdbc.source_TC_userpassword}</password>
<servername/>
<data_tablespace/>
<index_tablespace/>
<attributes>
<attribute><code>EXTRA_OPTION_MSSQL.domain</code>
<attribute>DOMAIN_NAME</attribute>
</attribute>
</attributes>
</connection>
```

For **DOMAIN_NAME**, substitute the actual Windows domain name that will be doing the authentication.

4. Run the upgrade scripts for your version and operating system (.bat for Windows users and .sh for Linux users).

Important: After each upgrade script, you must run the installer. Run the correct **WH_install.bat** or **WH_install.sh** file depending upon your operating system. These files are in the **datawarehouse** target directory.

The respective paths are as follows:

- a. Data Warehouse 4.0:

```
Run WH_upgrade_40U1.bat or WH_upgrade_40U1.sh
Run Installer
Run WH_upgrade_402.bat or WH_upgrade_402.sh
Run Installer
Run WH_upgrade_41.bat or WH_upgrade_41.sh
Run Installer
Run WH_upgrade_50.bat or WH_upgrade_50.sh
Run Installer
```

- b. Data Warehouse 4.0 Update 1:

```
Run WH_upgrade_402.bat or WH_upgrade_402.sh
Run Installer
```

Run WH_upgrade_41.bat or WH_upgrade_41.sh
Run Installer
Run WH_upgrade_50.bat or WH_upgrade_50.sh
Run Installer

c. Data Warehouse 4.0 Update 2:

Run WH_upgrade_41.bat or WH_upgrade_41.sh
Run Installer
Run WH_upgrade_50.bat or WH_upgrade_50.sh
Run Installer