ETL Validator Upgrade Guide

Table of contents

Introduction 3
ETL Validator Topology 3
Preparing to Upgrade 4
Upgrading ETL Validator Complete 5
Upgrading ETL Validator Multi-user 6
    Upgrading ETL Validator Server 6
    Upgrading ETL Validator Client, Repository and Work Schema 7
Troubleshooting 8
Reverting to pre-upgrade version 8

Introduction

The key components of ETL Validator are :

ETL Validator Client
ETL Validator Client provides the User Interface for configuration, creating and managing Test Cases, scheduling Test Plans and administration

ETL Validator Server
ETL Validator server is a J2EE application for running of Test Cases and Test Plans. It also includes Web Based reporting

ETL Validator Repository
ETL Validator Repository requires an Oracle database or a Postgres database. It stores the metadata related to test cases, execution plans, connections and administration.

ETL Validator Work Schema
ETL Validator Work Schema requires an Oracle database or a Postgres database. It stores the test results, benchmark data and Validation results.

ETL Validator CommandLine
ETL Validator CommandLine is a java CLI tool for invoking ETL Validator test cases and test plans remotely from command line.

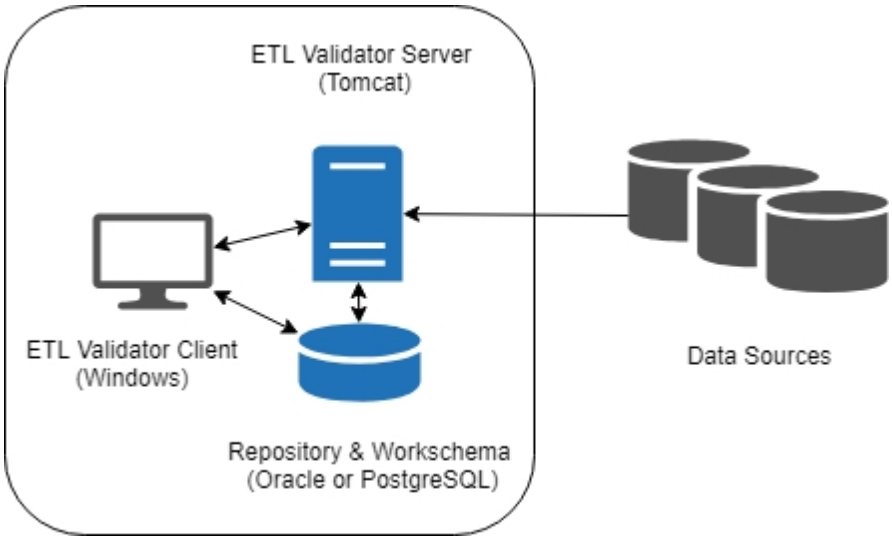
Upgrading ETL Validator typically involves upgrading all of the above component to same version. The following sections provide more details on how to do these tasks.

ETL Validator Topology

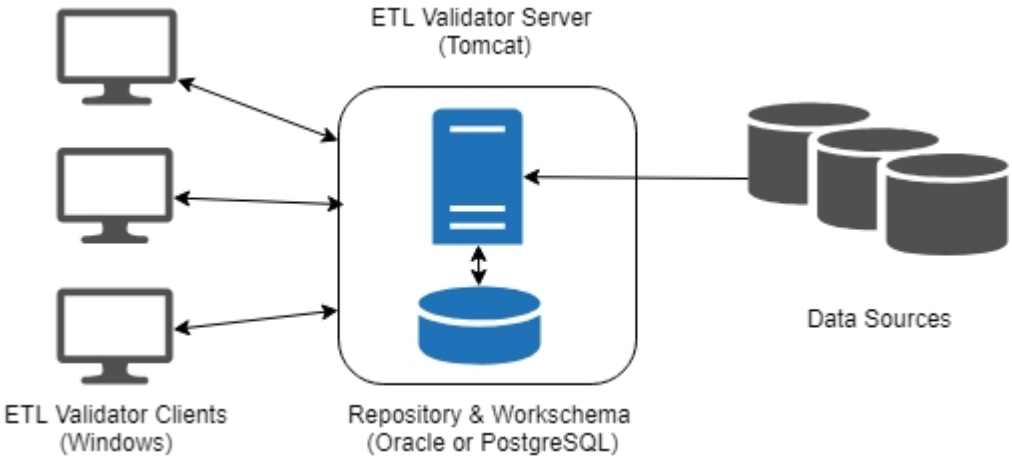
ETL Validator Topology

ETL Validator is available in the following two topologies:

- **ETL Validator Complete (Single user)** - This topology consists of ETL Validator Client and Server running on the same windows Server or windows 10 computer with a Postgres database. ETL Validator Complete can be setup by running the installer available in the "ETLValidator_Complete" download file. The installer automatically installs the ETL Validator Client, a repository/work schema using Postgres database and the server running on tomcat using a default port of 6050. This configuration comes with an embedded JRE. and prompts you for installing .Net Framework Full.



- **ETL Validator multi-user :** In this topology, the ETL Validator Client and Server are installed separately on the different computers using "ETL Validator Client Only" and "ETL Validator Server" install files. ETL Validator Workschema/Repository can be co-located in on the same server as the ETL Validator Server (example Postgres) or installed on another server (example Oracle). ETL Validator server can be deployed in Tomcat version 8.5 or later.



Preparing to Upgrade

This section lists the recommended steps to be followed prior to starting the upgrade:

- Take a backup of your ETL Validator Repository and Work Schema databases. This will enable you to revert back to your existing version in case of any issue.
- Keep a backup of the ETL Validator software install files (eg. ETL Validator Complete, ETL Validator Client, ETL Validator Server war file, License file) that were used for setting up of your current version. You can use them to revert back to your existing version in case of any issues.
- Make sure that ETL Validator Server is not running any Test Plans are Test Cases

prior to bringing it down for the upgrade.

- All the components of ETL Validator - Client, Repository, Work Schema, Server and CommandLine need to be on the same version. This means that all the components need to be upgraded to the same version so that they can be functional again. **In a multiuser environment, all the client software needs to be upgraded before they can connect to the upgraded Repository and Work Schema.** So plan your upgrade such that all the ETL Validator Clients are upgraded at the same time so as to avoid any downtime for the users in using ETL Validator.

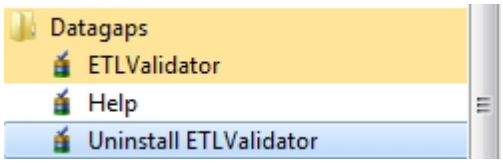
- If you have ETL Validator set up in a backup and recovery configuration, it is recommended that you upgrade both the primary and backup at the same time.

Upgrading ETL Validator Complete

ETL Validator Complete is typically used in a single user setup where the ETL Validator Client and Server are both locally installed as part of the ETL Validator Complete install. Follow these steps to upgrade ETL Validator Complete:

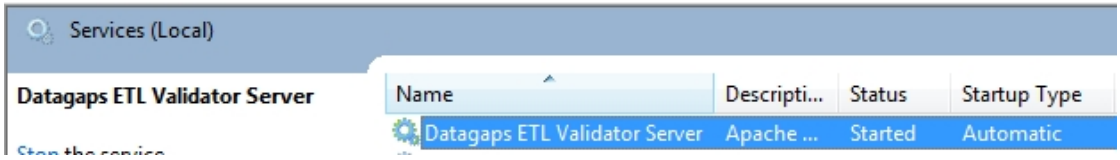
- Ensure that you have gone over and completed the steps outlined in the Upgrade Preparation section. We have tried to make the upgrade as seamless as possible but it is important to take a backup of your Repository and Work Schema just in case you need to revert back to your existing version.

- Uninstall ETL Validator by clicking on Program Menu -> Datagaps -> Uninstall ETL Validator



- Install the latest version of ETL Validator Complete by clicking on the "ETLValidator_Complete.exe" file

- Verify that the ETL Validator Server is running by navigating to the Windows Services. If it is not started, please start the server



- Launch ETL Validator Client by going to Programs Menu -> Datagaps -> ETL Validator

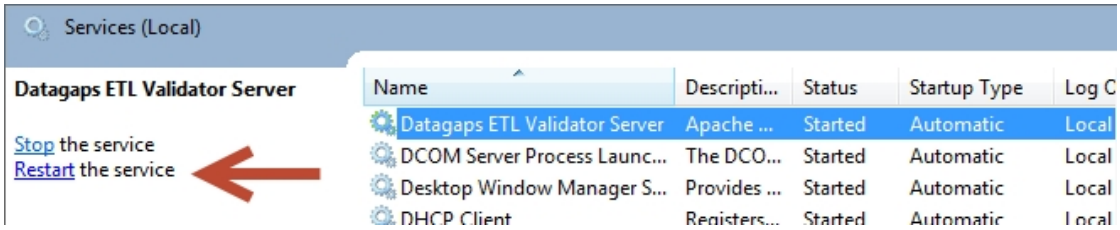
- ETL Validator Client automatically checks for existing repository connections and prompts for login to the repository. Click on Login.

- During Login, ETL Validator automatically checks the Repository version and prompts a message asking for upgrade of the repository. Click 'Ok' to upgrade the repository.

- ETL Validator next checks to see if the ETL Validator Server is configured properly. Since the server has been re-installed as part of ETL Validator Complete, a message is shown about the server not able to connect. Click on 'Edit' and 'Save' so that the

Server is configured with the correct repository information. For ETL Validator Complete, the server URL should be : http://localhost:6050/ETLValidatorServer/Execute?wsdl

- Click on 'Test' and then 'Save' once it has been successfully tested. Restart the ETL Validator Server process so that the settings take effect. This step ensures that the ETL Validator Server knows which repository database to connect to and tells the ETL Validator Client the location of the ETL Validator Server.



- The ETL Validator Client will next try to upgrade the Work Schema. If there are more than one Work Schema each of them is automatically upgraded.

This completes the upgrade process for ETL Validator Complete. See the troubleshooting section or contact datagaps if you encounter any issues during the upgrade.

Upgrading ETL Validator Multi-user

In a multi-user setup, there are typically multiple ETL Validator Clients using a common Repository, Work Schema and Server. Please refer to the ETL Validator Topology for more details.

Upgrading ETL Validator in a multi-user setup involves the following steps :

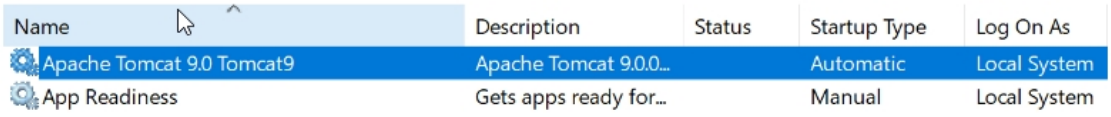
- Upgrade ETL Validator Server
- Upgrade ETL Validator Client, Repository and Work Schema

Please review the steps in upgrade preparation prior to starting the upgrade process. **Also note that in a multiuser environment, all the ETL Validator Clients needs to be upgraded before they can connect to the upgraded Repository and Work Schema.** So plan your upgrade such that all the Clients are upgraded at the same time so as to avoid any downtime for the users in using ETL Validator again.

Upgrading ETL Validator Server

Upgrading ETL Validator Server involves re-deploying the ETLValidatorServer.war and ETLValidatorService.war file in the Application where the ETL Validator Server process is running. The recommended J2EE container for ETL Validator Server is Tomcat 8.5. This upgrade guide assumes that you have ETL Validator Server running in Tomcat 8.5.

- Open Windows Services and stop the Apache Tomcat Service where ETL Validator Server is deployed :

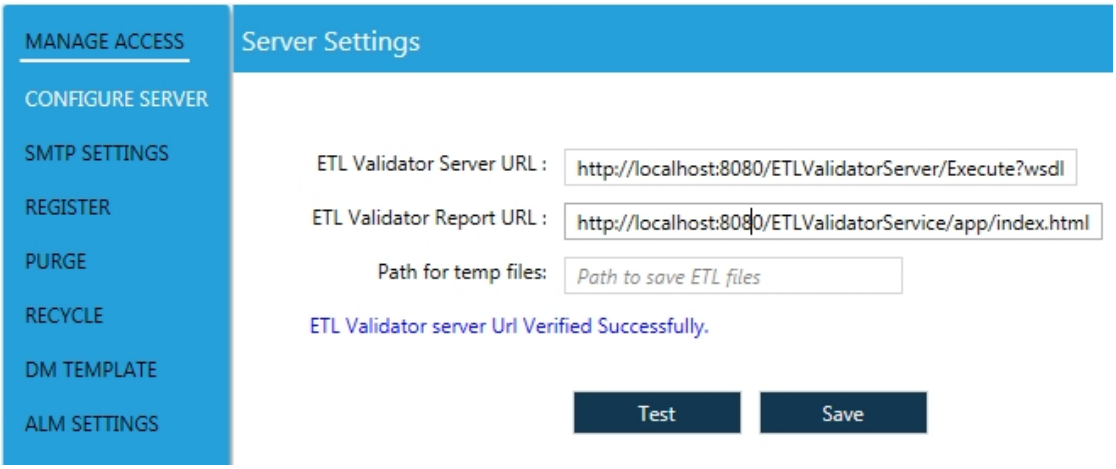


- Navigate to <catalina home>/webapps and delete the folders 'ETLValidatorServer' and 'ETLValidatorService'. Also, delete the two war files 'ETLValidatorServer.war' and 'ETLValidatorService.war' from the <catalina home>/webapps folder

- Copy the latest versions of the two war files 'ETLValidatorServer.war' and 'ETLValidatorService.war' in the <catalina home>/webapps folder

- Restart the Apache Tomcat Windows Service

- The next set of steps are for ensuring that the re-deployed ETL Validator Server has the necessary repository connection information. This can be done from the ETL Validator Client by navigating to the Admin Options->Server menu.



- Click on 'Test' and then 'Save' once it has been successfully tested. Restart the tomcat so that the settings take effect. This step ensures that the ETL Validator Server and the ETL Validator Service (reporting) knows which repository database to connect to and tells the ETL Validator Client the location of the ETL Validator Server.

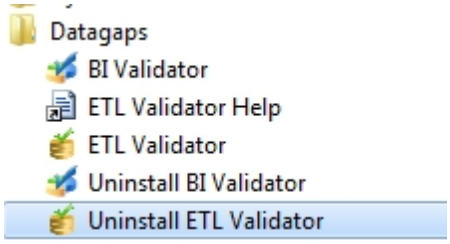
This completes the Server Upgrade steps. See the troubleshooting section or contact datagaps if you encounter any issues during the upgrade.

Upgrading ETL Validator Client, Repository and Work Schema

Upgrade ETL Validator Client, Repository and Work Schema involves re-installing of the ETL Validator Client and a one time upgrade of the ETL Validator Repository and Work Schema when the ETL Validator Client connects to the repository after the client upgrade. The steps are outlined below:

- Ensure that you have gone over and completed the steps outlined in the Upgrade Preparation section.

- Uninstall ETL Validator Client by clicking on Program Menu -> Datagaps -> Uninstall ETL Validator



- Verify that the ETL Validator Server was already upgraded and running. If it is not started, please start the server.
- Install the latest version of ETL Validator Client by clicking on the "ETLValidator_Client.exe" file
- Launch ETL Validator Client by going to Programs Menu -> Datagaps -> ETL Validator
- ETL Validator Client automatically checks for existing repository connections and prompts for login to the repository. Click on Login.
- During Login, ETL Validator Client checks the Repository version and automatically prompts a message if a Repository Upgrade is needed.
- ETL Validator Client next checks to see if the ETL Validator Server is configured properly. If the server was already upgraded and configured as the Upgrading ETL Validator Server section, there should be any need for additional configuration. If these steps were not performed, the Server Configuration window will be prompted automatically.
- The ETL Validator Client also checks for the upgrade of the Work Schema. If there are more than one Work Schema each of them is automatically upgraded.

This completes the upgrade process for ETL Validator Client, Repository and Work Schema. See the troubleshooting section or contact datagaps if you encounter any issues during the upgrade.

Troubleshooting

Please refer to the ETL Validator Support site below for trouble shooting tips.

http://datagaps.freshdesk.com/solution/articles/1000235862-troubleshoot-login-issues

Reverting to pre-upgrade version

Reverting to your previous version requires that you have a backup of the ETL Validator Repository and Work Schema prior to the upgrade. Assuming that you have the backups, please follow the steps below:

For ETL Validator Complete :

- Uninstall ETL Validator by clicking on Program Menu -> Datagaps -> Uninstall ETL Validator
- Restore the ETL Validator Repository and Work Schema databases. Ensure that the Repository and Work Schema users have the appropriate privileges as they did prior

to the upgrade.

- Re-install the pre-upgrade version of the ETL Validator Complete software
- Configure the server connection information upon login

For ETL Validator Multi-user :

- Uninstall ETL Validator Client by clicking on Program Menu -> Datagaps -> Uninstall ETL Validator
- Restore the ETL Validator Repository and Work Schema databases. Ensure that the Repository and Work Schema users have the appropriate privileges as they did prior to the upgrade
- Undeploy the ETL Validator Server and redeploy the previous version of the ETLValidatorServer.war file
- Re-install the pre-upgrade version of the ETL Validator Client software
- Configure the server connection information upon login using the ETL Validator Client