Table of Contents

Using The Site

Server Administration

Getting Started

Validatar's Data Quality Catalog

Data Profiling

Enhancing Metadata & Validatar Objects

Connecting to Data

Creating Validatar Tests

Projects > Tests

Projects > Jobs

Reports

Project Settings

Project Settings > Tools

Project Settings

Global Server Settings

Global Server Settings > Configuration

Quick Start Projects

Resources

Product Changes

Previous Version Documentation

EULA

Using the Validatar Documentation

Documentation Overview

Welcome to the Validatar Documentation site where we hope you can find answers to your questions and problems and enhance your journey toward improved data quality with Validatar.

If this is your first time here let us give you a quick tour so you can make the most of the features available to you.

Navigation

On the left-hand side of the screen, you will find the main navigation. It is organized for you to easily find what you are looking for.

Inside each article, you will also find helpful navigation. On the right-hand side of your screen, you will find the article's table of contents that will let you quickly and easily jump to any section in the article. Below the table of contents, you will find links to related articles in the documentation.

Search

At the top of your screen, you will find the Search bar. This will give you the fastest way to find what you are looking for across the entire site.

Knowledge Base Assistant In Validatar

You may not know it, but starting in Validatar 2021.2 this documentation is available directly inside of the Validatar interface through the Knowledge Base Assistant. It will provide you context-based access to documentation that is relevant to the page you are on in Validatar.

Types of Documentation

On this site, we have several types of documentation to help you on your Data Quality journey.

- **Getting Started With Validatar** - These articles are designed to help you ramp up as quickly as possible so you can begin using Validatar effectively to improve your organization's data quality.
- **Feature Documentation** - These articles provide comprehensive documentation about a specific feature, function, or page in Validatar.
- **Knowledge Base Articles** - These articles walk you through Validatar best practices and helps to optimize your Validatar experience.
- **Concept Guides** - These articles are designed to step back from specific features and provide guidance for how you could approach a particular problem or implement a solution related to Data Quality.
- **Product Changes - If you are looking to find out what is new in each version of Validatar this area will get you up to speed.**
- **EULA** - In case you have questions about our End User License Agreement we have made it easy to access here.

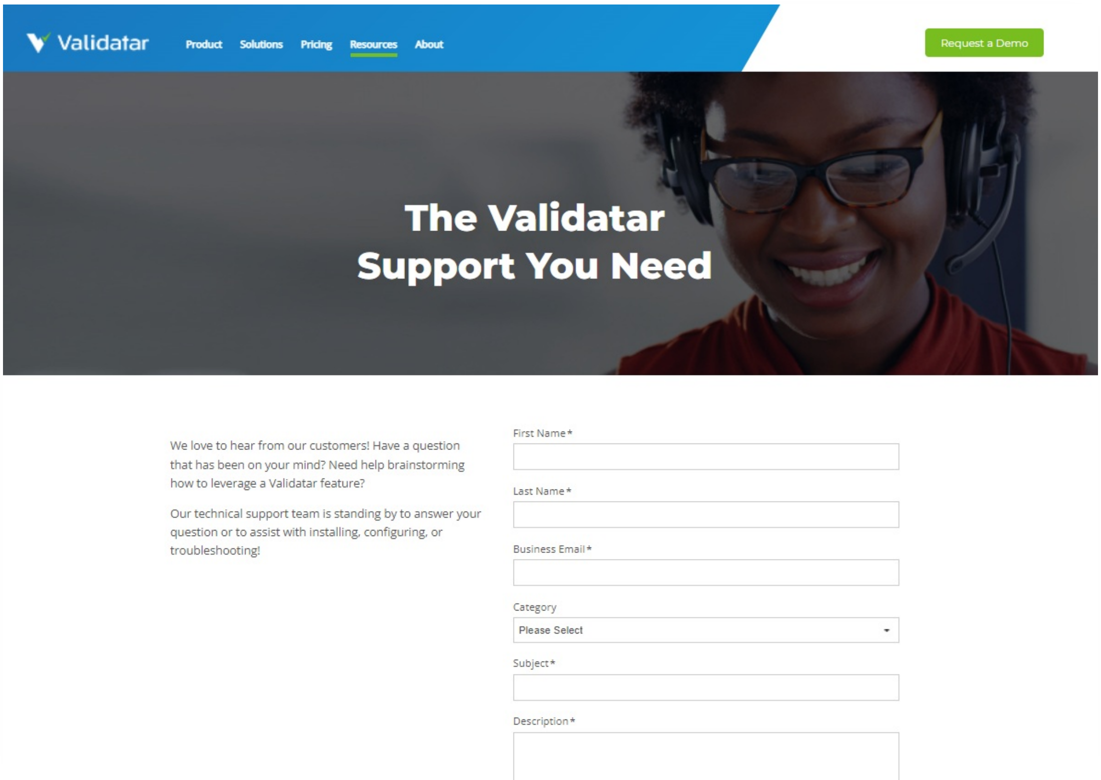
Interactivity and Feedback

At the bottom of each article, you will find options for providing feedback on how helpful you found the article. We appreciate your feedback so we can continually improve the value this documentation is providing you.

Getting Support

Contacting Support

To get in contact with support, please submit a customer support form. https://www.validatar.com/support



We love to hear from our customers! Have a question that has been on your mind? Need help brainstorming how to leverage a Validatar feature?

Our technical support team is standing by to answer your question or to assist with installing, configuring, or troubleshooting!

First Name*

Last Name*

Business Email*

Category

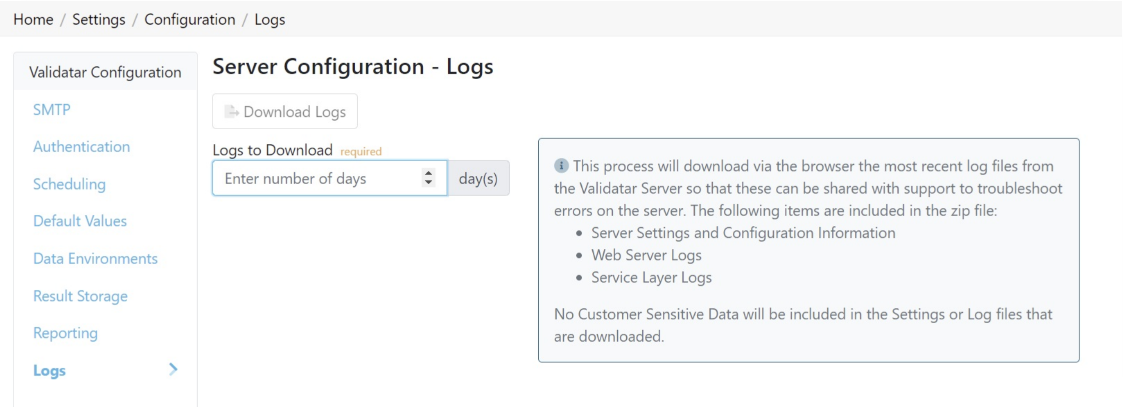
Please Select

Subject*

Description*

Working with Support

There are times when you will need to work with our support team to get them information about what is happening in your Validatar environment. This may require us to review logs. A Validatar Global Admin can download recent logs from the server by navigating to **Global Settings** > **Configuration** > **Logs** if this is required.



Server Configuration - Logs

Logs to Download required

Enter number of days

This process will download via the browser the most recent log files from the Validatar Server so that these can be shared with support to troubleshoot errors on the server. The following items are included in the zip file:

- Server Settings and Configuration Information
- Web Server Logs
- Service Layer Logs

No Customer Sensitive Data will be included in the Settings or Log files that are downloaded.

Install Guide

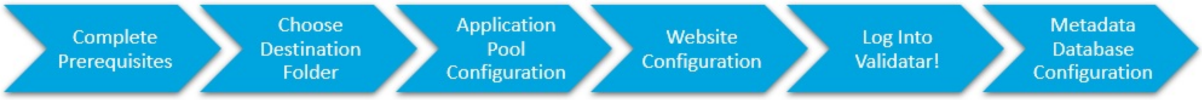
Overview

Installing Validatar consists of two steps:

- Download the installer at https://admin.validatar.com/releases. Choose the most recent release.
- Complete prerequisites relating to the web server and the database server

Completing Installation

Est. Time to Complete: ~ 1 hour



Complete Prerequisites → Choose Destination Folder → Application Pool Configuration → Website Configuration → Log Into Validatar! → Metadata Database Configuration

Prerequisites

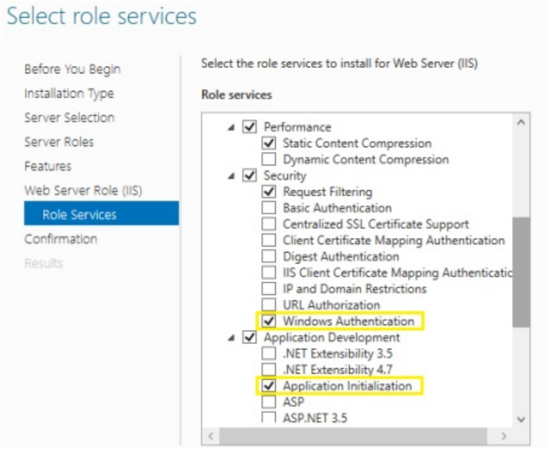
Access needed to complete install

- Admin access to the operating system to run the installer
- Admin access to the web server
- Admin access to the database Validatar will use on the back end

Web server

- HDD: 50 GB (the web application files take up roughly 150 MB)
- RAM: 4-16 GB
- CPU: 2.0 GHz or faster (2-8 cores)

The server must have Internet Information Services (IIS) installed, including the following highlighted options:



Select role services

Select the role services to install for Web Server (IIS)

Role services

- ☑ Performance
  - ☑ Static Content Compression
  - ☐ Dynamic Content Compression
- ☑ Security
  - ☑ Request Filtering
  - ☐ Basic Authentication
  - ☐ Centralized SSL Certificate Support
  - ☐ Client Certificate Mapping Authentication
  - ☐ Digest Authentication
  - ☐ IIS Client Certificate Mapping Authenticatio
  - ☐ IP and Domain Restrictions
  - ☐ URL Authorization
  - ☑ Windows Authentication
- ☑ Application Development
  - ☐ .NET Extensibility 3.5
  - ☐ .NET Extensibility 4.7
  - ☑ Application Initialization
  - ☐ ASP
  - ☐ ASP.NET 3.5

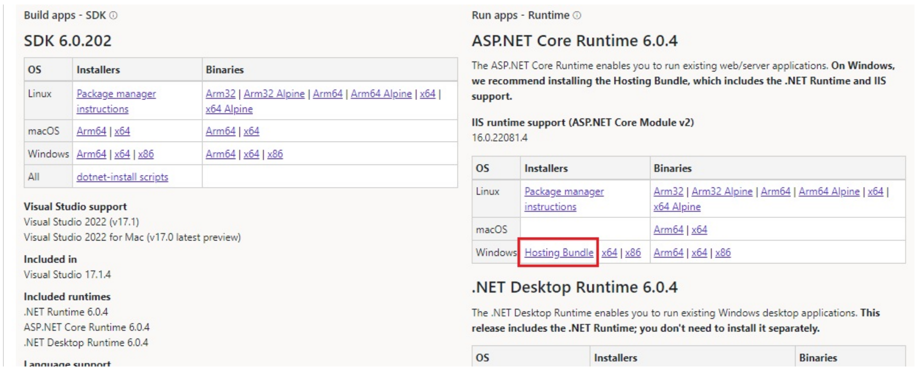
To add and configure IIS:

1. Type **Server Manager** in the Windows search bar
2. Click **Add Roles & Features**
   - Also found under **Manage** in the top-right corner
3. Click **Next** for every page until you reach the **Server Roles** section
4. In Server Roles check **Web Server (IIS)** then **Add Features**
5. Click **Next**
6. In the **Role Services** section
   - Scroll down the web services list, expand **Security**, check **Windows Authentication**
   - Expand **Application Development**, check **Application Initialization**

  - Click **Next**, then **Install**

Install the following required tools if not already installed:

- The web server must have the **IIS URL Rewrite** extension installed: download URL Rewrite here
- The web server must have the **Hosting Bundle** installed for ASP.NET Core Runtime 6.0.x: download .NET Core here



Build apps - SDK ⓘ

SDK 6.0.202

OS	Installers	Binaries
Linux	Package manager instructions	Arm32 | Arm32 Alpine | Arm64 | Arm64 Alpine | x64 | x64 Alpine
macOS	Arm64 | x64	Arm64 | x64
Windows	Arm64 | x64 | x86	Arm64 | x64 | x86
All	dotnet-install scripts	

Visual Studio support
Visual Studio 2022 (v17.1)
Visual Studio 2022 for Mac (v17.0 latest preview)

Included in
Visual Studio 17.1.4

Included runtimes
.NET Runtime 6.0.4
ASP.NET Core Runtime 6.0.4
.NET Desktop Runtime 6.0.4

Language support

Run apps - Runtime ⓘ

ASP.NET Core Runtime 6.0.4

The ASP.NET Core Runtime enables you to run existing web/server applications. **On Windows, we recommend installing the Hosting Bundle, which includes the .NET Runtime and IIS support.**

IIS runtime support (ASP.NET Core Module v2)
16.0.22081.4

OS	Installers	Binaries
Linux	Package manager instructions	Arm32 | Arm32 Alpine | Arm64 | Arm64 Alpine | x64 | x64 Alpine
macOS		Arm64 | x64
Windows	Hosting Bundle | x64 | x86	Arm64 | x64 | x86

.NET Desktop Runtime 6.0.4

The .NET Desktop Runtime enables you to run existing Windows desktop applications. **This release includes the .NET Runtime; you don't need to install it separately.**

OS	Installers	Binaries

Be sure to select the **Hosting Bundle** option.

Additional optional steps:

- If you will be testing Snowflake data, install the **Snowflake ODBC Driver**: download Snowflake ODBC Driver here
- Create a DNS record for the Validatar website so that Validatar can be accessed using something other than the web server name or IP Address.
- Obtain an SSL certificate from a certificate authority to encrypt network traffic to Validatar. After installing Validatar, use IIS to assign the SSL certificate to the Validatar website (see detailed steps below).

Database server (SQL Server or PostgreSQL)

- Supported versions
  - SQL Server 2019 (no older than 2014)
  - PostgreSQL 14.1
  - SQL Server 2019 Express (no older than 2014) - only recommended for proof of concepts or trials
- Common Specifications
  - HDD: 50 GB
  - SQL Drive: allocate 50 GB total for the Validatar metadata data/log files
  - RAM: 16 GB
  - CPU: 2.0 GHz or faster
- Create a new database named "Validatar"
- Minimum security permissions for the Validatar Service Account
  - At least db_owner database role in the Validatar database
- If using Windows Authentication instead of SQL Server authentication for connections created in Validatar to the databases you're testing, your IT will need to create an Active Directory service account and provide the account credentials. That domain account should then be used when configuring the Validatar application pool in IIS during the install process.

SMTP credentials

Configure SMTP settings so that Validatar can send password reset and other notification emails. If the SMTP settings are not known, check with your IT administrator or Network administrator for this information.

Installation

1. Download the installer from https://admin.validatar.com/releases
2. Run the installer. If the user does not have the operating system administrator role, they will need to Run as Administrator.
3. Enter IIS application pool identity information – this will be the account that Validatar runs under. This must be a domain account if database connections use Windows authentication instead of SQL Server Authentication.

4. Enter IIS website information for the Validatar website. The port number should typically be 80 unless you already have another website listening on that port.
5. Finish install.
6. In IIS, verify that the default site is stopped and the Validatar website is running.

Are you running into issues? View common installation issues at the bottom of the page.

Assigning an SSL Certificate

Complete the following steps to assign an SSL certificate to the Validatar site to secure network traffic.

1. Open IIS.
2. In the **Connections** tree, expand the server, then expand **Sites**.
3. Select the **Validatar** site.
4. In the **Actions** pane on the right, under Edit Site, select **Bindings...**
5. In the **Site Bindings** window, select the **https** binding, then click **Edit**.
6. In the SSL certificate drop-down, select your newly installed SSL certificate.
7. Select OK.

Configuration

1. **For New Installations**
    1. In a web browser, navigate to the Validatar URL
    2. Select the database engine and enter the connection string to the Validatar database
    3. Enter the information for the Validatar global administrator user
    4. Click the “Complete Setup” button
    5. Login using the credentials of the administrator user you just created

2. **Activate your license**
    1. Navigate to **Settings > Licenses**
    2. Select “Activate License”
    3. Read the Validatar End-User License Agreement and select “I Agree”
    4. Fill out the license information form.
    5. Click the “Activate” button

3. **Configure SMTP**
    1. Navigate to **Settings > Configuration > SMTP**
    2. Enter required SMTP settings
    3. Click the "Send Test Email" button to test out your changes
    4. Click the "Save" button

Common Installation Issues

Issue	Possible Solution
Installer rolls back and fails	Run the installer via command-line and produce a log file using the following steps: 1. Open the Command Prompt 2. Execute the following command, replacing "C:\tmp" to reflect where the installer is saved: C++ Copy `msiexec /i "C:\tmp\validatar-setup.msi" /L*V "C:\tmp\validatar-setup.log"` 3. If the install is unsuccessful, send the log file created in C:\tmp\validatar-setup.log to **support@validatar.com**.
Installer ended prematurely	1. Verify the installer is run either by a web server admin or Run As Administrator. 2. In IIS, verify that the default site is stopped and the Validatar site is started.
Error 404 appears when attempting to navigate to the Validatar server address in a browser, **or** the user is brought to IIS when navigating to the Validatar server address in a browser	In IIS, verify that the default site is stopped and the Validatar site is started.
Error 500.19 appears when attempting to navigate to the Validatar server address in a browser	1. Verify that the **Hosting Bundle** for ASP.NET Core was installed instead of the **Windows x64** version. 2. Stop and restart application pools in IIS.
After logging into Validatar, the user only sees Project Data Sources and Tools under Project Settings	Verify that the license key is activated.

Upgrading to New Versions

Overview

Upgrading Validatar is a super easy process! Use the following steps to complete the upgrade.

Instructions

1. Backup the Validatar SQL Server metadata database.
2. In Microsoft IIS, record the prior host and port settings so that the same settings can be entered during the new install.
3. In Windows Explorer, open C:\inetpub and create a copy of the Validatar folder just in case you need to roll back to the previous version.
4. Download the installer from https://admin.validatar.com/releases
5. Enter IIS application pool identity information – this will be the account that Validatar runs under. This must be a domain account if database connections will use Windows authentication instead of SQL Server Authentication.
6. Enter the IIS website information for the Validatar website that you recorded in step 2.
7. Complete the upgrade and log in to Validatar.

Command Line Interface Setup

The validatarcli command-line application allows jobs to be executed from an external scheduler. The domain account running the validatarcli application will be used to authenticate the user into Validatar, so the command line does not require any credentials. The following steps can be used to install validatarcli on an application server – this assumes that Validatar has already been installed on a separate web server.

> **Remember**
>
> The user who is launching the job must have access to the application server in order to execute the job via command line.

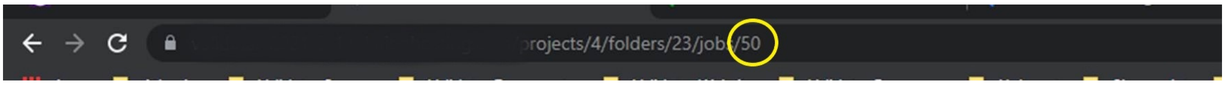
Configuration

1. Make sure .NET Runtime 6.0.x has been installed on the application server. It can be downloaded from here if necessary: https://dotnet.microsoft.com/download/dotnet/6.0. **NOTE:** On that page, scroll past the ASP.NET Core Runtime 6.0.x and .NET Desktop Runtime 6.0.x sections to the .NET Runtime 6.0.x section and download the Windows x64 installer.
2. Download and run the validatarcli-setup.msi installer from here: https://admin.validatar.com/releases.
3. Ensure that the domain account that will be used for authentication has been added to a Validatar user profile.

Testing The Command Line

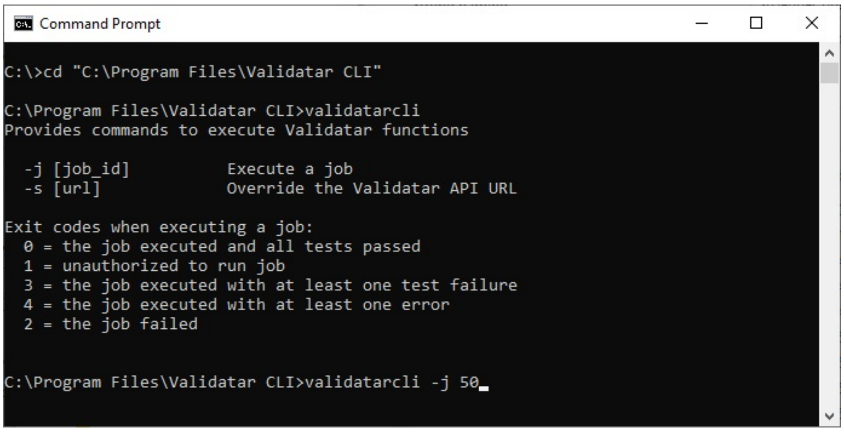
> **Additional Help**
>
> Refer to the screenshot at the bottom of this section for additional assistance.

1. In Validatar, under **Global Settings > Users**, add the Windows AD or domain account to the user's profile who will execute the job if it hasn't been added already.
2. Navigate to the job you want to execute.
3. Grab the job ID which is the last number in the URL.



projects/4/folders/23/jobs/50

4. Using the application server that you want to externally run the job on, open up a command prompt.
5. Navigate to the folder that you installed validatarcli to in step 2 of the configuration. You may have to change directories using the cd command. We're using the Z:\ drive in the example below.
6. After navigating down to the folder type **validatarcli**
7. Press Enter.
8. The Validatar commands should appear on the next two lines.
9. Next type **validatarcli -j [insert job ID]**. Be sure to include a space between -j and the job ID.
10. Press Enter. The job will now execute.
11. The exit code that appears refers to the result of the job.
12. Navigate back to Validatar, under Results, to be sure the job executed correctly and displayed the proper exit code in the command prompt.



```
C:\>cd "C:\Program Files\Validatar CLI"

C:\Program Files\Validatar CLI>validatarcli
Provides commands to execute Validatar functions

  -j [job_id]          Execute a job
  -s [url]             Override the Validatar API URL

Exit codes when executing a job:
  0 = the job executed and all tests passed
  1 = unauthorized to run job
  3 = the job executed with at least one test failure
  4 = the job executed with at least one error
  2 = the job failed


C:\Program Files\Validatar CLI>validatarcli -j 50
```

Integrating Validatar with an External Tool

After you confirm that you're able to execute jobs via command line, you can include this into your external scheduler workflow. You can either kick off a Validatar job externally or based on the result of the Validatar job, kick off an external job.

A way to do this is to create a Powershell host script to call the Validatar CLI. If you want to kick off an external job based on the result of the Validatar job, you can configure your script to release the appropriate external job based on the return code of the CLI. You may need to refer to the external tool's documentation to get the syntax for a job-release.

When building out the process, it's highly recommended to use parameters or extended properties as much as possible. That will make the process more flexible and easier to change.

Installing WhereScape Test Pack

Introduction

WhereScape RED is a data infrastructure automation tool that helps simplify and standardize the development of data warehouses. The purpose of the WhereScape RED Metadata Test Pack for Validatar is to deliver a collection of test scripts that will audit and validate data warehousing structures using WhereScape RED's metadata repository. Through the use of the WhereScape RED Metadata Test Pack, users will be able to discover, assess, and resolve issues related to their data structures while optimizing time, cost, and human resources.

The rest of this document will refer to the WhereScape RED Metadata Test Pack as simply the Test Pack.

Overview

This guide is for Validatar users who have built their data infrastructure using WhereScape
RED's automation tool. Users are looking to validate their development process to ensure their data structures follow best practices as well as meet the standards of their organization.
Key topics covered are:

- Creating the required project
- Setting up data sources
- Importing the test pack

Installation

Creating the Test Pack Project

It is recommended that a project is created specifically to store the Test Pack to keep its contents separate from other tests that users have created. To create the project:

1. Log into Validatar as a Global Administrator
2. Choose **Settings** in the main menu bar
3. Choose Projects in the navigation tree
4. Select New Project
5. Enter **WhereScape RED Test Pack** as the project name
6. Enter a project description (optional)

Getting the Right Permissions

For the Test Pack to be imported, the user must have the Import user role enabled. The following steps will enable the Import user role:

1. Log into Validatar as a Project Administrator
2. In the**WhereScape RED Test Pack** project, go to **Project Settings** > **Project Permissions** in the navigation pane
3. Grant the **Import** role to the appropriate user or user group.
4. Click **Save**

Once a user has the Import user role enabled, they will see Import under their lists of tools.

Setting Up Data Source

Before you can import the Test Pack, at least one Data Source must be configured and enabled in your project that you plan to use with the Test Pack.
If you haven't set up a WhereScape Repository data source yet:

1. Navigate to **Settings** > **Data Sources**.
2. Choose **New Data Source**.
3. Enter **WhereScape Repository**, or something similar, as the name.
4. Choose the Environment the repository belongs to.
5. Under **Connection**, choose **+Add**.

6. Configure the connection to the WhereScape Repository database. If you need assistance with creating connection strings, please visit Connection Strings
7. Test the connection.
8. Choose **Create Connection**.
9. Choose **Save** at the top of the page.

Repeat for each WhereScape Repository you want to add. You are now ready to import the Test Pack.

Importing the Test Pack

The following steps will allow you to import the Test Pack:

1. Make sure you are logged into Validatar as a user with the Import role.
2. In the appropriate project, go to Tools in the navigation pane
3. Choose Import
4. Step 1 of 3: Upload File
   a. Click Select File
   b. Choose the 'WhereScape_RED_Metadata_Test_Pack.xml' file
   c. Click Open
5. Step 2 of 3: Select Destination
   1. Map each source connection from the import file to a target connection in your project.

      - **Connection 1** is used in every repository audit test for both the source and target connections. Connection 1 is also used in every repository comparison test as the source connection, or the connection that serves as a baseline.
      - **Connection 2** is only used in repository comparison tests as the target connection, or the connection that warrants the comparison.
        Note: You can change the connections for each test after the import is complete.

   2. Under Project Test Folders, choose **Tests** for the Import Target folder
   3. Under Project Job Folders, choose **Jobs** for the Import Target folder
      Note: If you ever need to return to a page, select Previous to go to the previous page or select Restart which brings you back to Step 1: Upload File.

6. Step 3 of 3: Review
   1. The last step is merging the Test Pack with the other tests in the target folder. Ideally, all tests, jobs, and folders should be highlighted in a green color to symbolize their addition to the folder.
   2. Choose **Import**

Installing Validatar Data Agent - Windows

Prerequisites

- ASP.NET Core Runtime 6.0 - (Microsoft Download Site with Options) Recommended installation link for the complete Hosting Bundle (Direct download link)

Installation Instructions

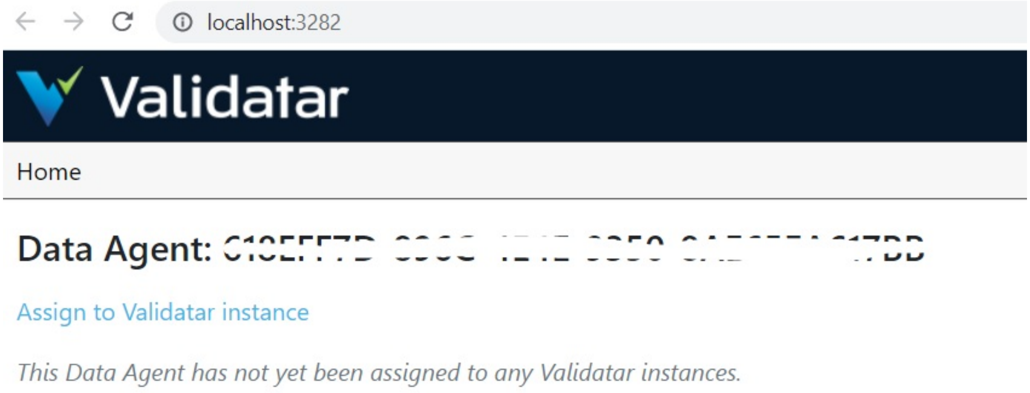
- Download the latest version of the Validatar Data Agent installer from Validatar Releases

- Run the .msi installer file and go through the individual prompts

    - Recommended to install to the Default location
    - Use the default port 3282 unless otherwise required
    - Check the box to configure the data agent

Configuration Instructions

- A browser window should pop up once the install is complete. If not, open a browser and navigate to: http://localhost:3282/

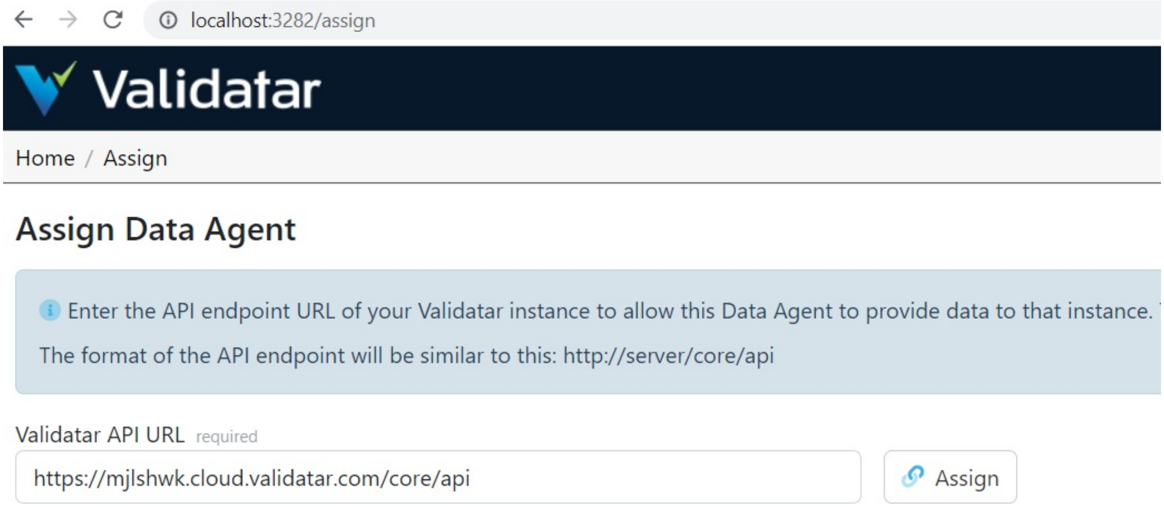


Data Agent: [redacted]

Assign to Validatar instance

This Data Agent has not yet been assigned to any Validatar instances.

- Click the Assign to Validatar Instance and then enter the API URL associated with your Validatar Cloud instance. It should be in this form: **https://[hosting_prefix].cloud.validatar.com/core/api** for Validatar Cloud instances or **https://[hosting_url]/core/api** for Validatar Server instances.



localhost:3282/assign

Home / Assign

Assign Data Agent

Enter the API endpoint URL of your Validatar instance to allow this Data Agent to provide data to that instance.

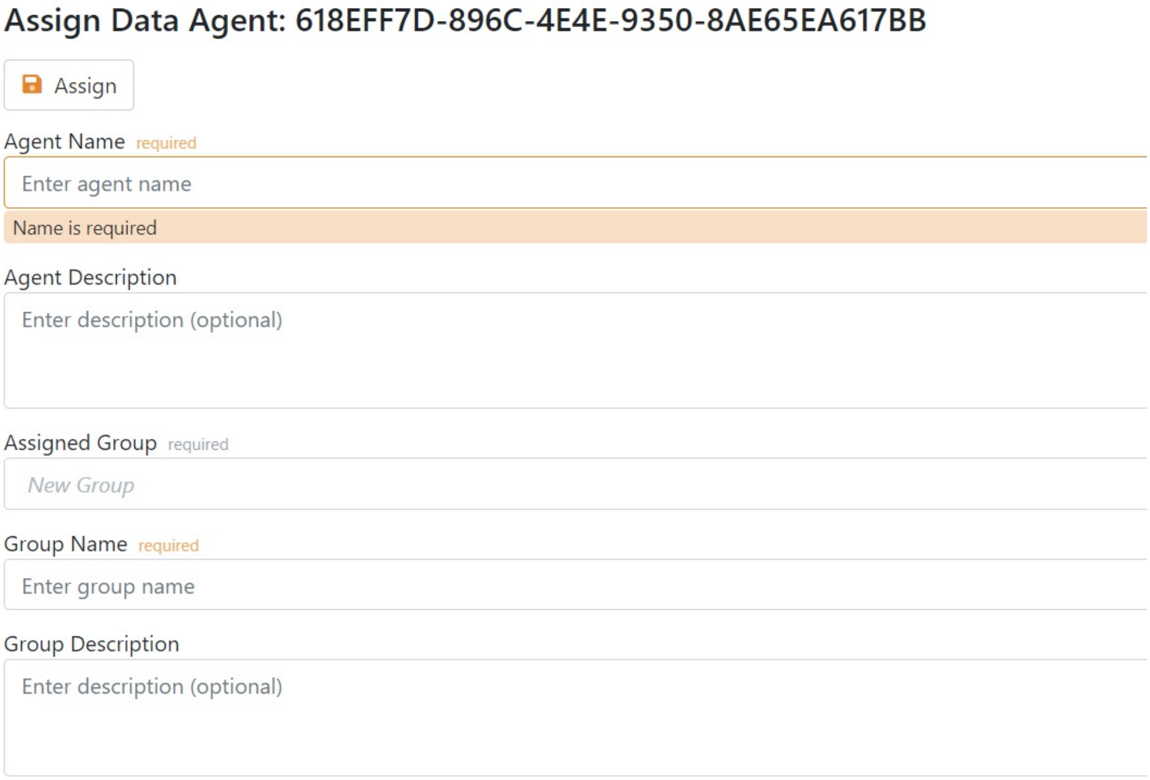
The format of the API endpoint will be similar to this: http://server/core/api

Validatar API URL required

https://mjlshwk.cloud.validatar.com/core/api

- Click Assign. The browser will send you to login to your cloud instance. If you already have a browser session open in your cloud instance it will take you directly to the following page.

- Fill out an Agent Name and a Group Name for the local computer you are currently assigning to the cloud instance. The Group Name is what you or others will select when setting up a new data source and telling it what group to send query requests to. This allows for multiple Agents to respond to one data soure's requests allowing for redundancy and scalability.



Assign Data Agent: 618EFF7D-896C-4E4E-9350-8AE65EA617BB

Agent Name required

Enter agent name

Name is required

Agent Description

Enter description (optional)

Assigned Group required

New Group

Group Name required

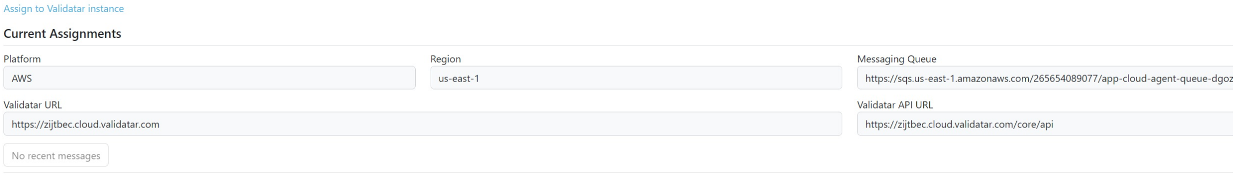
Enter group name

Group Description

Enter description (optional)

Copyright 2022 Validatar

- Click Assign and it should take a few seconds and then take you back to http://localhost:3282/ and show something like this



Assign to Validatar instance

Current Assignments

Platform: AWS

Region: us-east-1

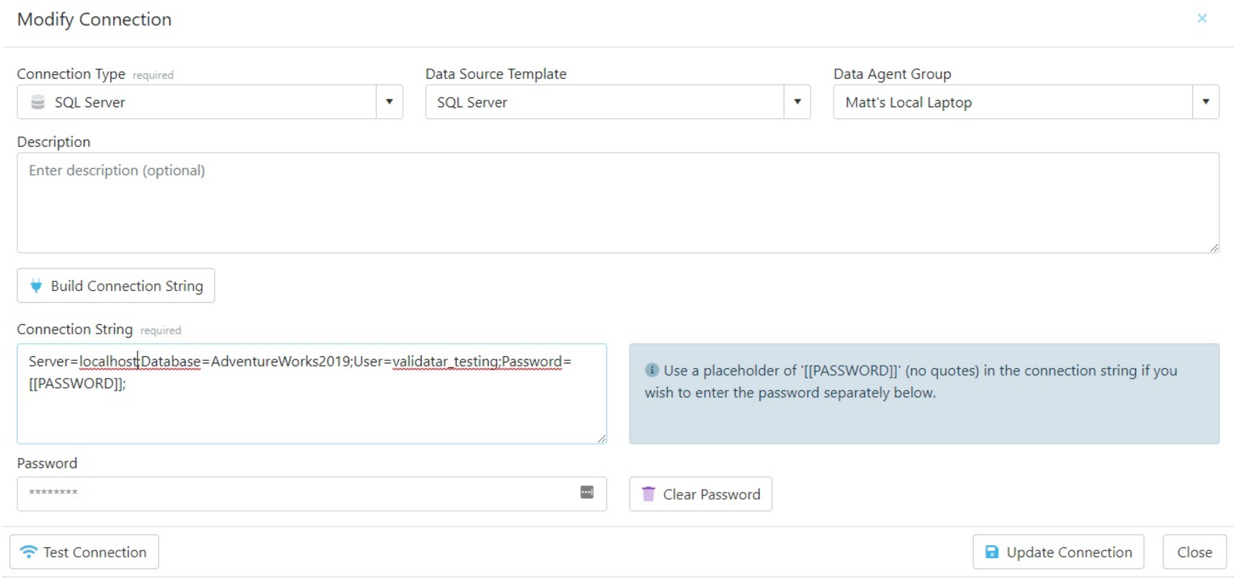
Messaging Queue: https://sqs.us-east-1.amazonaws.com/265654089077/app-cloud-agent-queue-dgoz

Validatar URL: https://zijtbec.cloud.validatar.com

Validatar API URL: https://zijtbec.cloud.validatar.com/core/api

No recent messages

- Now from your Validatar Cloud instance when you set up the connection for a new data source you can select the Group Name you just created and Validatar will send the queries to this Validatar Data Agent to be run.



Modify Connection

Connection Type required: SQL Server

Data Source Template: SQL Server

Data Agent Group: Matt's Local Laptop

Description: Enter description (optional)

Connection String required:

```
Server=localhost;Database=AdventureWorks2019;User=validatar_testing;Password=[[PASSWORD]];
```

Use a placeholder of '[[PASSWORD]]' (no quotes) in the connection string if you wish to enter the password separately below.

Password: ********

Pre-Installation Requirements Checklist - Windows

Requirements Checklist

Access needed during Installation Process

- Administrator user with access to the Application/Web server operating system to run the installer
- Administrator user with access to the Database Server

Service Accounts to be Created

If using Windows Server + SQL Server repository then:

Either

- Single Domain Account
  - Used for IIS configuration on Application/Web Server
  - Needs db_owner permission on SQL Server Validatar database
  - Needs db_datareader permission on any SQL Server databases that you would like to use Windows Authentication to connect to

or:

- Non-Domain Accounts
  - Local Account (automatically created during install) on Application/Web Server for IIS configuration
  - SQL Authentication Service Account with db_owner permission on Validatar database
  - Requires SQL Authentication accounts for all other SQL Server databases you would like to connect to

If using Windows Server + PostgreSQL repository then:

- Local or Domain Account (automatically created during install if Local) on Application/Web Server for IIS configuration
- PostgreSQL service account with full privileges on all tables in the public schema of the Validatar database

Windows Application/Web Server Recommended Specifications

Validatar Server Teams Edition

- HDD: 50 GB (the web application files take up roughly 150 MB)
- RAM: 8-16 GB
- CPU: 2-4 cores (2.0 GHz or faster)

Validatar Server Enterprise Edition

- HDD: 100 GB (the web application files take up roughly 150 MB)
- RAM: 32-64 GB
- CPU: 8-16 cores (2.0 GHz or faster)

Database Server Recommended Specifications

Validatar Server Teams Edition

- Storage: 200GB
- RAM: 8 GB
- CPU: 1 core equivalent

Validatar Server Enterprise Edition

- HDD: 400GB
- RAM: 16 GB
- CPU: 2 core equivalent

Windows Application/Web Server Dependencies

- The application/web server must have **Internet Information Services (IIS)** installed (see additional steps in the install guide)
- The application/web server must have the **IIS URL Rewrite** extension installed: download URL Rewrite here
- The application/web server must have the **Hosting Bundle** installed for ASP.NET Core Runtime 6.0.x: download .NET Core here
- ODBC Drivers
  - Drivers for all of the data source platforms you would like Validatar to connect to will need to be installed and available in the ODBC manager.
    - Common ODBC Drivers you may need to install:
      - Snowflake
      - Oracle
      - Teradata
      - DB2
  - Recommended to use 64 bit ODBC drivers if possible
  - If 32 bit drivers are required then IIS Application pool must be configured to run as 32 bit and all ODBC drivers will need to be 32 bit
- The application/web server must have Java and Solr installed to use the Global Search functionality
  - Download Java: https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.msi
  - Download Solr from here: https://solr.apache.org/downloads.html. Use the link for the latest binary release zip file (https://www.apache.org/dyn/closer.lua/lucene/solr/8.11.1/solr-8.11.1.zip?action=download)
  - Follow this guide for configuration once the main installation is complete: https://docs.validatar.com/docs/en/global-search

Notes:

- Python will be installed automatically with any installation of a Validatar Data Agent. You will not need to install Python separately
- The Validatar UI manages the installation of any additional Python libraries

Required Networking Conditions

- Port 80 (or 443 if SSL is enabled) will need to be open coming in to the Validatar Application/Web Server so users can log on via a web browser. You can limit this to only from connections inside your network if you do no want users to be able to log in from external networks
- Standard Database Ports will need to be open between the Validatar Application/Web Server and the Validatar Repository in SQL Server or PostgreSQL
- Standard Database Ports will need to be open between the Validatar Application/Web Server and any databases it is being used to test and monitor

Additional Recommended Networking Requirements

- Create a DNS record pointed to the Validatar application/web server so that Validatar can be accessed using something other than the server name or IP Address.
- Obtain an SSL certificate from a certificate authority to encrypt network traffic to the specified DNS for the Validatar Server. After installing Validatar, use IIS to assign the SSL certificate to the Validatar website (see detailed steps in the installation guide).
- SMTP Credentials for your organization's SMTP server will be needed in order for Validatar to be able to send password reset and other notification emails.
- SMTP Ports will need to be open between the Validatar Application/Web Server and your SMTP server

10 Steps for Validatar Server

What is Validatar?

Validatar is a Data Quality Management and Automation platform that enables organizations to improve data quality & increase trust in data by automating the discovery, profiling, testing, and monitoring of its data assets.

Who Can Use Validatar and Why?

- Business Leaders will gain confidence that the data they are using is trustworthy.
- Data Stewards are given the tools they need to monitor and govern data assets.
- Data Developers can deliver accurate and trusted data faster using automated data testing.
- Data Consumers are given the visibility to know and trust the data assets available to them.

Common Use Cases

- Data Warehouse ETL Dev QA
- Data Governance
- Cloud Migration
- Data Quality Monitoring

10 Steps to Getting Started

Getting started in Validatar is simple, but we've created a list of ten tasks to complete before running your first test to make it even easier.

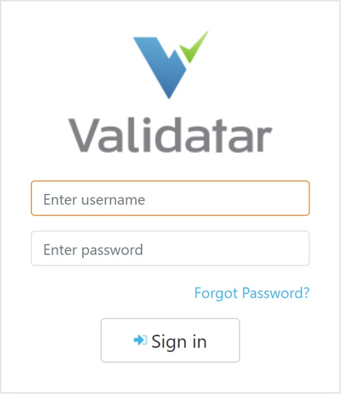
1.	Install the Validatar application on the appropriate server.
2.	Create a Global Admin user.
3.	Apply License information.
4.	Configure SMTP settings.
5.	Add additional Validatar users.
6.	Give your Validatar project a name.
7.	Create a data source with both primary and profiling connections configured.
8.	Ingest the data source's metadata.
9.	Run a full profile on your data. Try a small table first.
10.	Start running tests!

Accessing Validatar

Logging In

To login to Validatar:

1. Navigate to your instance of Validatar via the provided URL.
2. Enter your username and password.



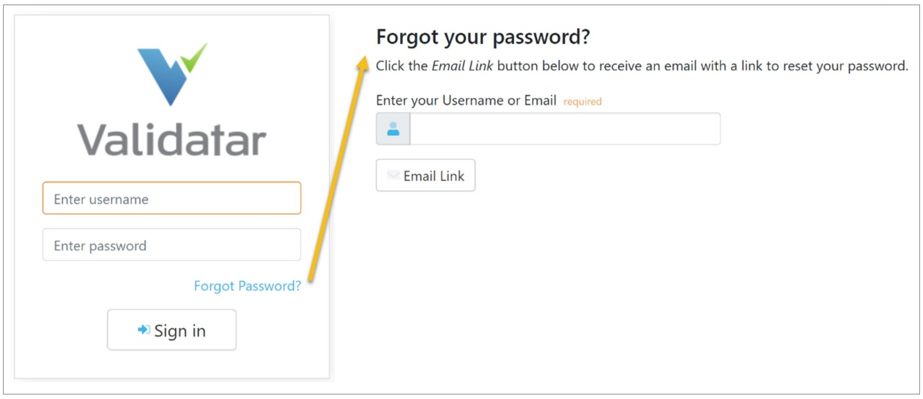
Forgot Password?

If you've forgotten your password:

- Navigate to the Login Screen and click on the **Forgot Password?** link. You will be prompted to provide your **username** or **email** and receive a link to reset your password.

If you've forgotten your username:

- Go through the reset password procedure described above using your email. This will allow you to log in and then view your username.


Forgot your password?

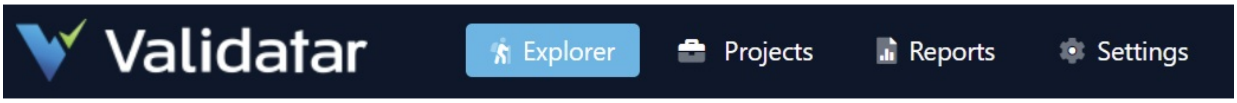
Click the *Email Link* button below to receive an email with a link to reset your password.

Enter your Username or Email required

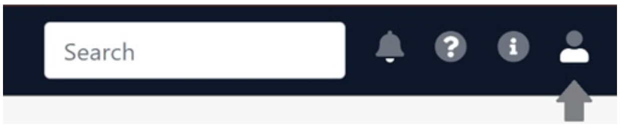
Navigating Validatar

The Toolbar

The main toolbar bar contains four main areas of Validatar: Explorer, Projects, Reports, and Settings.



1. **Explorer** - Brings you to a list of data sources that you have browse permissions for and changes the navigation tree to the explorer tree. If Validatar has ingested the data source's metadata, the latest ingestion date as well as statistical information about the data source will be shown.
2. **Projects** - Brings you to a list of every project that you have access to inside of Validatar and changes the navigation tree to the project tree. Browse the repository of existing tests, jobs, and results and create new test cases under Projects.
3. **Reports** - View standard reports like test and job history or create new custom reports by connecting Validatar to a business analytics tool like Tableau. Clicking Reports changes the navigation tree to the reports tree.
4. **Settings** - Global administrators can configure their Validatar server to support the needs of their organization. Add new data source connections, database engines, and custom profiles for enhanced monitoring and data quality testing. Clicking Settings changes the navigation tree to the settings tree.



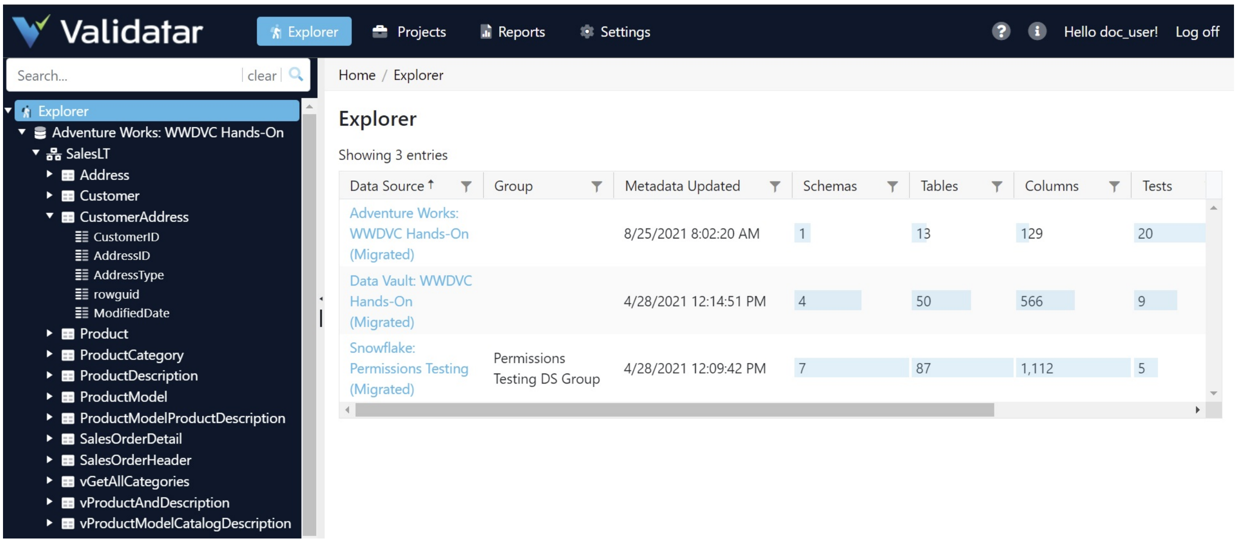
1. **Notifications** - Any recent Validatar notifications will appear here.
2. **Help** - Clicking the help icon displays Validatar's knowledge base. You can also find links to Validatar's website and support portal.
3. **About Validatar** - Here you can find information about Validatar such as the version you're currently on.
4. **User Account** - You can navigate to your account profile or log off by clicking on the user icon.

Understanding the Navigation Tree

Clicking a main Validatar area in the tool bar changes the navigation tree on the left side to display objects in that area of Validatar.

Explorer Tree

Browse a data source by expanding it's schemas, tables, and columns. You can also use the search bar to find an object. Clicking on an object in the data source tree brings you to that object's data profile.



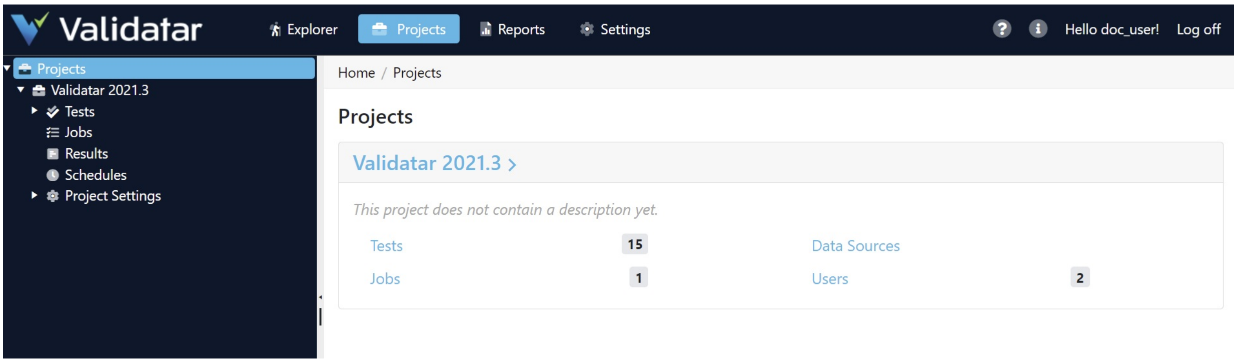
Explorer

Showing 3 entries

Data Source ↑	Group	Metadata Updated	Schemas	Tables	Columns	Tests
Adventure Works: WWDVC Hands-On (Migrated)		8/25/2021 8:02:20 AM	1	13	129	20
Data Vault: WWDVC Hands-On (Migrated)		4/28/2021 12:14:51 PM	4	50	566	9
Snowflake: Permissions Testing (Migrated)	Permissions Testing DS Group	4/28/2021 12:09:42 PM	7	87	1,112	5

Project Tree

Navigate between different Validatar projects using the project tree. Under each project you'll find the tests and job repositories, results, schedules, and project settings.



Home / Projects

Projects

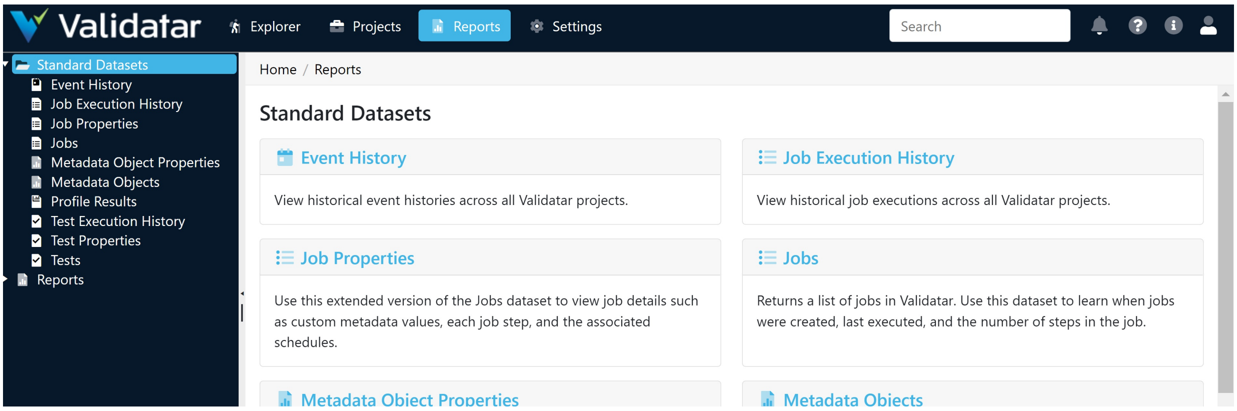
Validatar 2021.3 >

This project does not contain a description yet.

Tests	15	Data Sources	
Jobs	1	Users	2

Reports Tree

All datasets and reports you have access to will be listed on the reports tree.



Home / Reports

Standard Datasets

Event History

View historical event histories across all Validatar projects.

Job Execution History

View historical job executions across all Validatar projects.

Job Properties

Use this extended version of the Jobs dataset to view job details such as custom metadata values, each job step, and the associated schedules.

Jobs

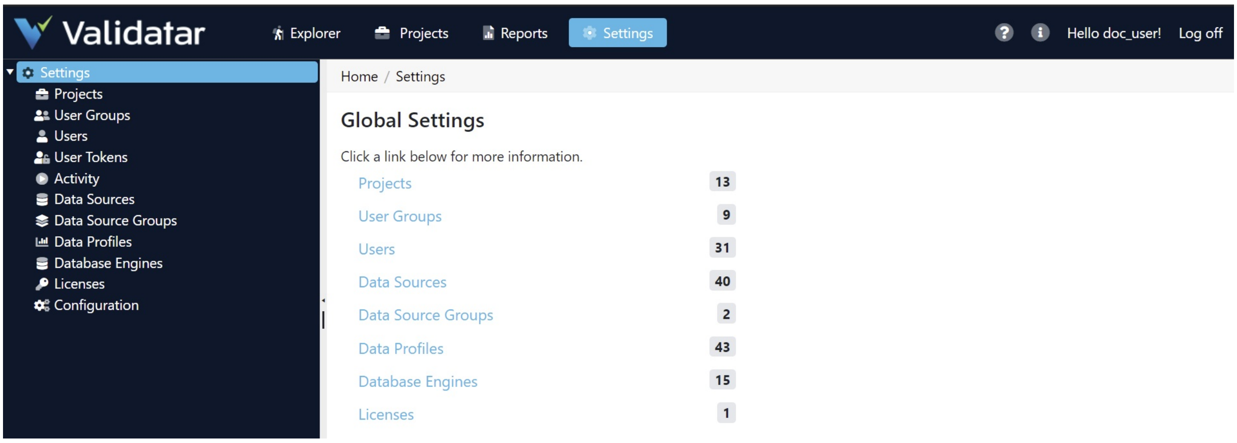
Returns a list of jobs in Validatar. Use this dataset to learn when jobs were created, last executed, and the number of steps in the job.

Metadata Object Properties

Metadata Objects

Settings Tree

Manage projects, users, data sources, and more using the Validatar settings tree.



Home / Settings

Global Settings

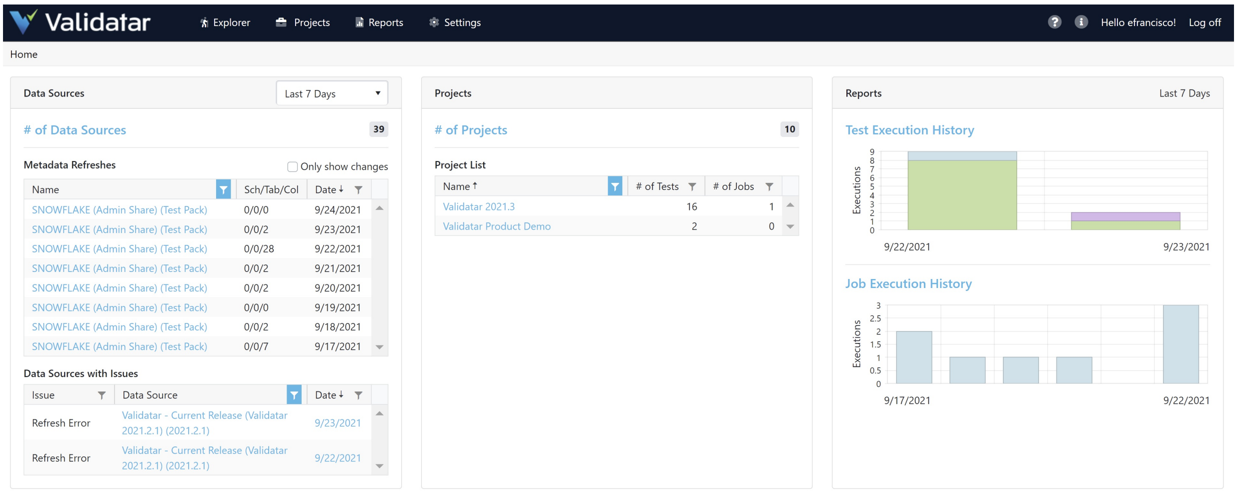
Click a link below for more information.

Projects	13
User Groups	9
Users	31
Data Sources	40
Data Source Groups	2
Data Profiles	43
Database Engines	15
Licenses	1

Home Page Dashboard

Overview

The Validatar Home page provides a quick look into your environment. Discover any changes or issues with your data sources, view your test and job execution history over the past seven days, and get quick access to your projects.



Data Sources — Last 7 Days

# of Data Sources 39

Metadata Refreshes ☐ Only show changes

Name	Sch/Tab/Col	Date
SNOWFLAKE (Admin Share) (Test Pack)	0/0/0	9/24/2021
SNOWFLAKE (Admin Share) (Test Pack)	0/0/2	9/23/2021
SNOWFLAKE (Admin Share) (Test Pack)	0/0/28	9/22/2021
SNOWFLAKE (Admin Share) (Test Pack)	0/0/2	9/21/2021
SNOWFLAKE (Admin Share) (Test Pack)	0/0/2	9/20/2021
SNOWFLAKE (Admin Share) (Test Pack)	0/0/0	9/19/2021
SNOWFLAKE (Admin Share) (Test Pack)	0/0/2	9/18/2021
SNOWFLAKE (Admin Share) (Test Pack)	0/0/7	9/17/2021

Data Sources with Issues

Issue	Data Source	Date
Refresh Error	Validatar - Current Release (Validatar 2021.2.1) (2021.2.1)	9/23/2021
Refresh Error	Validatar - Current Release (Validatar 2021.2.1) (2021.2.1)	9/22/2021

Projects

# of Projects 10

Project List

Name	# of Tests	# of Jobs
Validatar 2021.3	16	1
Validatar Product Demo	2	0

Reports — Last 7 Days

Test Execution History

Job Execution History

Click around to discover something new!

What's Your Validatar Role?

Overview

Validatar users can be broken down into three primary categories:

- An admin
- A creator
- A viewer

Your view and interaction with Validatar depend on which of the three categories is your primary responsibility. Take a look at your role below to get a nuanced view of Validatar.

I'm An Admin

As an admin, you'll spend adequate time in **Settings**, **Data Sources Settings**, and **Project Settings** to manage the use of Validatar. This includes

- Setup configuration
- Adding Users and User Groups
- Creating Data Sources
- Managing permissions for Projects and Data Sources
- and more!

Responsibilities

Depending on the type of admin you are, you have access to a combination of global, data source, and project settings. Let's look at what managing a Validatar instance may be like.

1. Admins have to **activate new license keys**. Navigate to Settings > Licenses to get started.
2. As an admin, **you must create any new users**. New users can access Validatar by
    1. Entering login credentials created by the admin
    2. Using the forgot password feature to create a new password
3. We highly suggest **creating user groups** to better control permissions assigned for Validatar users.
    1. Ideally, user groups are people who require the same level of access to projects and data sources.
    2. A Best Practice for grouping users is to group by team or by Validatar role (admins, testers, viewers)
4. As an admin, **you'll have to create new data sources and/or assign other users Data Source Admin privileges** so they can create data sources themselves. One of the main prerequisites to testing and observing your data is creating a connection to that data. So as an admin, you'll want to get data sources created immediately.
    1. Validatar ships with everything required to create SQL Server and Snowflake data sources. If you're using any other database engine, an admin will have to set up a new Database Engine under **Settings** before your data source can be used in its entirety.
5. Admins usually assist with **upgrading to a new Validatar version**.

Important Articles

Here are helpful documentation articles for admins to review.

- Security and Permissions Overview
- Upgrading to New Versions
- Database Engines
- Connection Strings
- Setting up SnowSQL on Windows

I'm A Creator

As a creator, you'll spend adequate time in **Projects** creating and editing tests, viewing results, and running jobs. As a creator, you'll have to work with admins to ensure you have the correct permissions and access to projects and data sources.

Responsibilities

Creators are mainly responsible for ensuring that testing is comprehensive enough for the organization's needs, tests are written accurately, and the right people have the right access to test results.

1. Learn how to **create standard and template tests**. Standard tests are used to test specific, narrow-ranging scenarios. Template tests are used to execute the same tests across multiple database objects. Building tests require knowledge of Validatar's test types, test data types, and results configuration.
2. You'll want to **continuously monitor test results** to assess the improvement of data quality. Continuous testing proves important for new deployments, code changes, regression testing, and ever-changing data. Validatar ensures that as new data arrives, it fits within the standards set and checked by tests.
3. We want you to evolve **from burdensome manual testing to simple and automated test execution.** Automated testing uses jobs, template tests, and schedules to execute tests and view results without having to log into Validatar.
4. Help bridge the gap between business users and technical users by **linking metadata to tests**. By creating this connection, the team will be able to monitor the quality of each metadata object based on the results of the linked tests. You'll also be able to use profile results to create tests by linking the metadata.

Important Articles

- Configuring a Test
- Configuring a Template Test
- Configuring a Job

Security and Permissions Overview

Overview

Prerequisite Required

You must be a **Global or Security Admin** to create user accounts and assign roles.

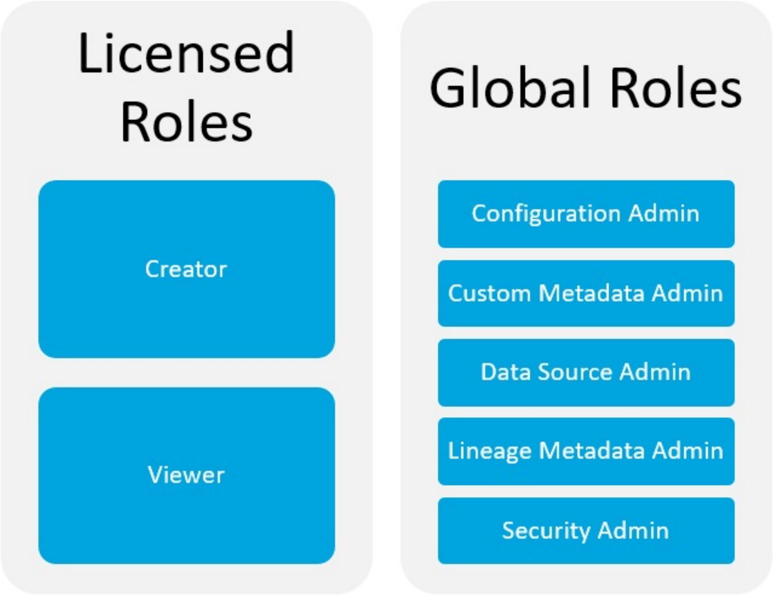
Validatar has various security permissions at the global, project, and data source levels. Users are added globally and then given the appropriate level of access to projects and data sources by either a global administrator or project administrator. Roles and Permissions determine how a user interacts with different objects and pages in Validatar. Learn more about both below.

You'll complete the following steps for each user:

1. Create a new user under **Settings**
2. Assign the appropriate licensed and global roles
3. Add the user to a user group (highly recommended)
4. Assign the user or user group permissions to data sources
5. Assign the user or user group permissions to projects
6. Invite each user to Validatar by
   - providing the server address and the user credentials you created
   - or sending them a password reset link using the *Forgot password* option on the login screen

1. Roles - What Can I Do?

When a user is created the user can have one or more roles assigned to their Validatar account. These roles define the type of Validatar admin a single user is, if required, and whether they can create content or only view content. There is a maximum limit of users who can be Creators at the same time based on the edition of Validatar that was purchased. Once that limit is reached, a user will have to lose the Creator role for another user to gain it. Find your volume utilization limits on the **Settings** page in Validatar.



- **Licensed Roles** are defined by the number of users allowed based on the purchased license tier and role type. Refer to Validatar's Pricing Guide for details about current volume limits for each licensing tier.
  - **Creator** - Allows users to create tests, jobs, and other Validatar objects.
  - **Viewer** - Allows users access to Validatar. **All user accounts are Viewers and this role cannot be removed.** To remove someone's access to Validatar, archive their user account.
- **Global Roles** provide administrative privileges to users and can be viewed in the **Settings** area. These admin roles

should be carefully assigned.

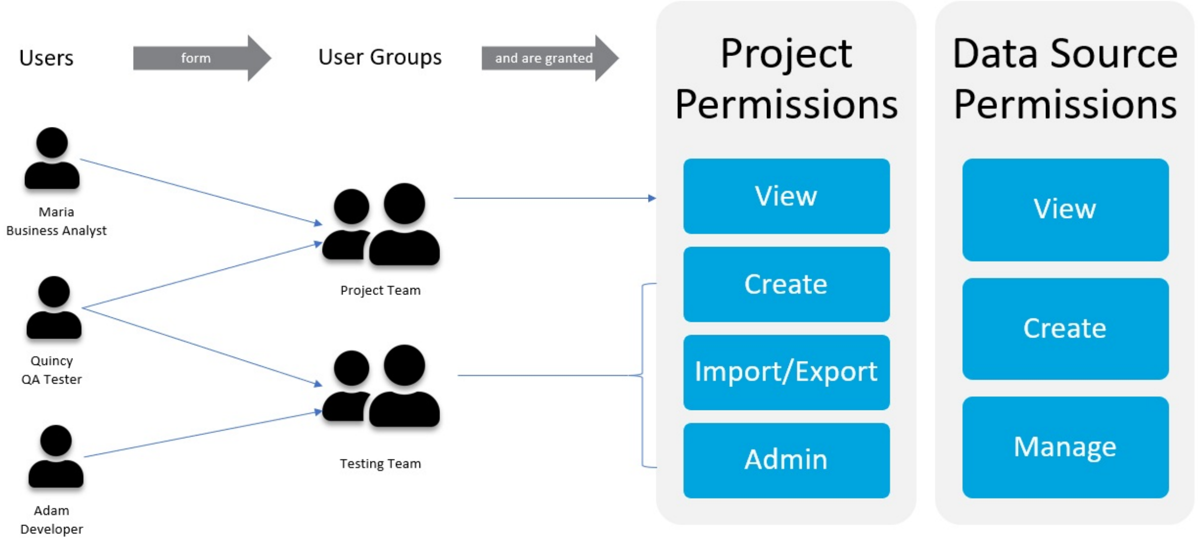
- **Configuration Admin** - Can manage Licenses and Configuration settings.
- **Custom Metadata Admin** - Can manage Catalog and Project Custom Fields.
- **Data Source Admin** - Can create new Data Sources. The permission to manage a single Data Source is set on the Data Source itself.
- **Lineage Metadata Admin** - Can create and manage Lineage Metadata sources.
- **Security Admin** - Can manage Projects, Users, Users Groups, and User Tokens.
- **NOTE:** A **Global Admin** has all of the above global roles.

> **Note**
>
> All users have access to the Activity page and read-only access to Data Sources they have any combination of permissions for which is found under **Settings**.

2. Permissions - What and Where Can I Do It?

After roles are defined for an individual user, permissions can be granted for Projects and Data Sources. This allows administrators to control what users or teams have access to each Validatar project and each data source. **We recommend using User Groups to assign permissions.**

- **Projects** house your testing and data validation work.
  - Most Validatar customers use **1 project** if it's acceptable for the entire team to have similar visibility into tests, results, and data sources. This setup is ideal for increasing transparency and consistency for comprehensive data testing.
  - You may want to create **two or more projects** if you want greater control over who can view specific content. If you have multiple teams whose testing needs don't overlap much, you may want to create a project for each team.
- A **Data Source** provides a connection to your database. Data Sources are used to create tests and execute data profiles.
  - Permissions are also assigned on a Data Source level which you'll learn about in the next section.
- **Users** are individual user accounts created in Validatar. Currently, users cannot create their own accounts. A Global or Security admin will create each account and then provide users access by either providing the user login credentials or sending them a password reset email. We suggest the latter. Users can then be granted access to projects and data sources.
- **User Groups** allow multiple users to have the same type of access to content. Whether users are on the same team or perform similar functions, user groups are a consistent way to assign and manage user access in Validatar. **We recommend using User Groups to assign permissions rather than granting permissions to individual users.**
- **Permission Sets** are default assignments of Project and Data Source permissions. We'll learn more about these a little later.



The diagram above demonstrates how three Validatar users can form User Groups and then be granted permissions to Projects and Data Sources.

- Maria and Quincy are a part of the business's project team and need to have visibility into tests, test results, and Data Source metadata and profiling information. Likely Maria will not be creating tests therefore she *will not* have the **Creator** role and will be a **Viewer** only.
- Quincy and Adam work together performing development and testing tasks so they'll be more hands-on creating Validatar content. We want them both to create tests, manage Validatar projects, and create the Data Sources they'll use for testing. In this case, Quincy and Adam *will* have the **Creator** role on their individual user accounts which will count towards the licensed creator volume limit.

Of course, you can create simple security rules like in the example above or a more complex usage of user groups and permissions that align better with your organizational structure. Validatar is fit to handle both.

Data Source Permissions

Validatar allows each data source to have its own independent set of permissions to prevent users from having unintentional access to data. You can add individual users or a user group to a data source by creating a **New Rule**.

The user who creates the data source automatically becomes the **data source owner** and has all permissions enabled. You can identify the owner by being the first username on the list and the image of a crown beside their username. The owner can be changed by double-clicking on the owner's username and selecting another user from the dropdown then selecting **Save**.

> **Data Source User Credentials**
>
> Keep in mind that a single user's credentials are used to connect to each database so be mindful of who is granted permission to create tests, run profiles, and see the data in that data source. For this reason, **we suggest using read-only, service account credentials to connect to each data source, unless otherwise needed**. If necessary, create multiple data sources using different credentials that provide the necessary level of access.

There are four groups of Data Source users:

- Global and data source admins
- Owner (the user who created the data source)

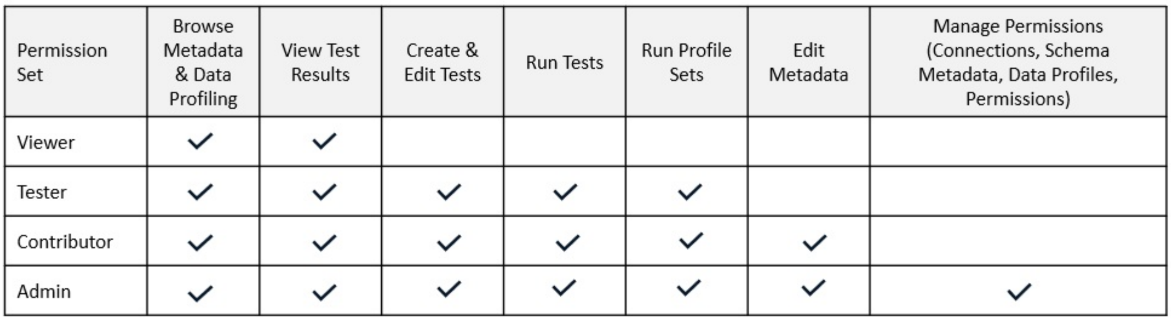
- User
- User Groups

The following is the list of available permissions available to configure for a data source.

Permission	Description
Browse Metadata & Data Profiling	The user can view the data source's data profile and metadata in the **Explorer** area. If the user does not have this permission, the data source will not appear on the **Explorer** page list.
View Test Results	The user can view test results where this particular data source is selected as the test or target data source.
Create & Edit Tests	The user can create new tests and modify existing tests.
Run Tests	The user can execute a test either from the test configuration page or the test home page.
Run Profile Sets	The user can execute profile sets that have been created under **Project Settings > Data Sources > Profile Sets**.
Edit Metadata	The user can modify custom field values for the data source in the **Explorer** area.
Manage Connections & Schema Metadata	The user can configure and modify a primary connection, metadata connection, and profile connection. The user can also configure metadata ingestion SQL, refresh a data source's metadata, and put the metadata refresh on a schedule.
Manage Data Profiling	The user can create new profile sets and modify existing profile sets. The user also has permission to rename and delete profile sets and the record of their subsequent executions.
Manage Permissions	The user can create new rules to add users and user groups to a data source. The user can also add and remove user permissions for the data source.

Data Source Permission Sets

Similar to Projects, you can easily assign permissions to a Data Source using Data Source permissions sets.



Permission Set	Browse Metadata & Data Profiling	View Test Results	Create & Edit Tests	Run Tests	Run Profile Sets	Edit Metadata	Manage Permissions (Connections, Schema Metadata, Data Profiles, Permissions)
Viewer	✓	✓					
Tester	✓	✓	✓	✓	✓		
Contributor	✓	✓	✓	✓	✓	✓	
Admin	✓	✓	✓	✓	✓	✓	✓

Project Permissions

The following is the list of available project roles and details of what they allow the user to do.

Role	Description
Create Jobs	The user can create jobs.
Create and Edit Labels	The user can create and modify labels to attach to tests, template tests, and jobs.
Import	The user can import objects into a project using an XML file.
Export	The user can export objects out of a project to an XML file.
Project Admin	The user has project admin access to the current project. You cannot remove your own project admin access from a project on this page.
Project Access	The user has read-only access to a project. In addition, the user can create and edit tests only if they are a global Test Author.

Granting Access

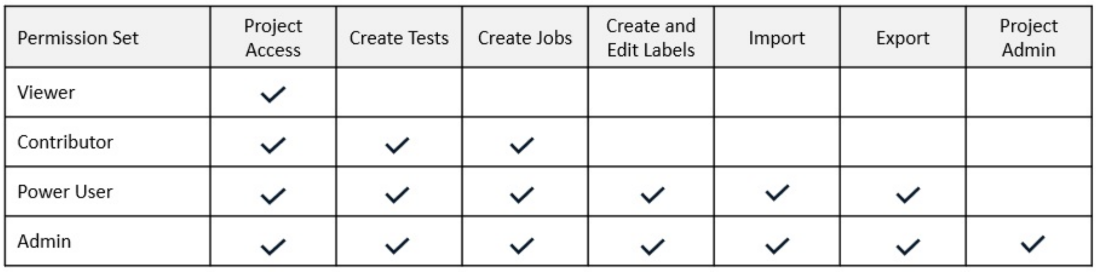
To add a user to a project:

1. Navigate to **Project Settings > Project Users**.
2. Select **Grant Access**.
3. Select the username from the **Choose Username** dropdown.

4. Select the appropriate roles for the user.
5. Select **Save**.

Project Permission Sets

Easily assign privileges to view, create, import/export, and perform admin tasks for each Validatar project. Project admin permissions can also be granted on a user's profile by following **Settings > Users > User Profile**. Custom permissions can also be granted if you choose not to use the permission sets.



Permission Set	Project Access	Create Tests	Create Jobs	Create and Edit Labels	Import	Export	Project Admin
Viewer	✓						
Contributor	✓	✓	✓				
Power User	✓	✓	✓	✓	✓	✓	
Admin	✓	✓	✓	✓	✓	✓	✓

Removing Access

To remove a user from a project:

1. Select the user from the list.
2. Select **Remove Access**.
3. Select **Remove** in the validation pop-up.

Note

You can immediately restore a user's access after you've removed them from a project as long as you do so before navigating away from the **Project User Permissions** page. Once you navigate away, you will have to go through the full process of granting a user access to a project to re-add them.

Using and Structuring Projects

Overview

Projects house testing and data validation work.

- Most Validatar customers use **one project** if it's acceptable for the entire team to have similar visibility into tests, results, and data sources. This setup is ideal for increasing transparency and consistency for comprehensive data testing.
- You may want to create **two or more projects** if you want greater control over who can view specific content. If you have multiple teams whose testing needs don't overlap much, you may want to create a project for each group.

Types of Projects

Projects can host any of the following testing scenarios plus more!

- Validating data warehouse development
- Data governance monitoring controls
- Validating cloud migrations

Organizing Your Project

Here are some ways Validatar users organize their projects:

- **By Environment** - Ideally, if you have multiple development environments (DEV, QA, PROD, etc.), you can keep relevant tests for each environment in separate projects. Validatar makes it easy to move tests from one project to another. So as development evolves, you can easily manage new and existing test cases.
- **By Project Team** - If you have multiple teams using Validatar, we recommend separating projects by the team. Usually, separate teams have separate functions, data sources, and processes they're responsible for. Having a project for each team ensures that the right teams have the proper access to your organization's data assets and increases collaborations amongst team members.
- **By Task** - Validatar can help businesses in a myriad of ways. Create a Validatar project for each internal project the organization has to ensure data quality is being monitored and is consistent throughout your business.
- **One all-inclusive project** - Using one project is advantageous when creating data transparency and consistency for the entire business. You're still able to assign permissions for specific users and data sources within a project, but all users are provided insight on what data quality metrics and scenarios are being tested.

Common Folder Structure

You can organize tests and jobs in Projects by using **Folders**. Here are a few suggestions folders can be used.

- **Source to Target Process** - Validatar is invaluable for testing entire data pipelines, but this can lead to a cluttered test repository if not managed well. Create a folder for each source to target hop in your data pipeline (Source to Load, Load to Stage, Stage to Fact/Dim, etc.). Also, you'll be able to associate business rules with the appropriate Validatar test.
- **Data Quality Metrics** - Here at Validatar, we call the six pillars of data quality CUVCAT (Completeness, Uniqueness, Validity, Consistency, Accuracy, Timeliness). Store tests related to each validated data quality metric in their own folder.
- **Business-Related Processes** - If there are organization-specific governing rules or practices, create a folder to hold all tests validating that process.

Turning on Notifications

Validatar can send notifications to users when significant events happen. You can configure notifications by opening your User Profile by:

1. Clicking the user icon in the main menu.
2. Selecting **User Profile**.
3. Selecting **Notifications** from the menu pane.

Validatar In-App Notifications

To receive notifications inside of Validatar, choose **Yes** under **Notify in Validatar**. You will see the number of notifications that haven't been viewed on the bell icon in the main menu bar. **Note:** Currently, notifications do not display screen banners or play a sound. The only visual indication is the notification bell icon.

Email Notifications

You can also set preferences to receive notifications via email. Set Email Preferences as

- **None** - You don't receive email notifications.
- **Immediate** - You receive a notification as soon as the event happens.
- **Daily Digest** - You receive an email containing all event notifications once daily. The Daily Digest time is configured at a Global Settings level.

Immediate Notifications

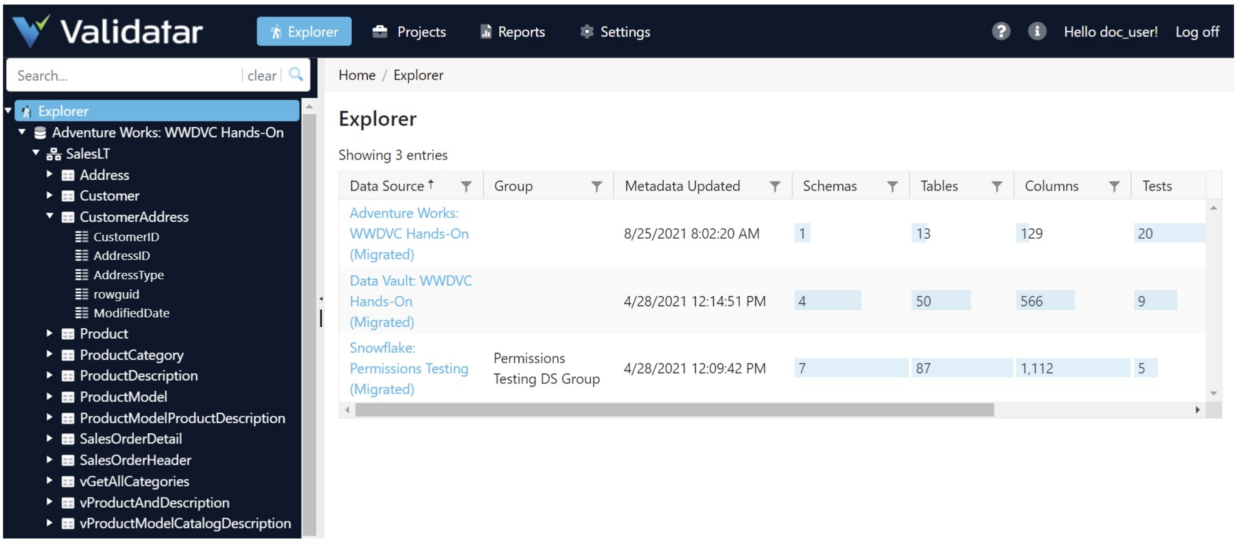
Events happening within minutes of each other may be grouped for efficiency, causing delays in immediate email notifications up to 5 minutes.

Explorer Overview

Overview

The Explorer is where data observability happens. Discover more about data profiling, data lineage, technical metadata, and more in Validatar's own data quality catalog.

- View metadata and data profiles.
- Use custom metadata for additional information about your metadata.
- Easily see test coverage and identify any gaps in testing.
- Search for objects of interest and see their data quality.



- Explorer
  - Adventure Works: WWDVC Hands-On
    - SalesLT
      - Address
      - Customer
      - CustomerAddress
        - CustomerID
        - AddressID
        - AddressType
        - rowguid
        - ModifiedDate
      - Product
      - ProductCategory
      - ProductDescription
      - ProductModel
      - ProductModelProductDescription
      - SalesOrderDetail
      - SalesOrderHeader
      - vGetAllCategories
      - vProductAndDescription
      - vProductModelCatalogDescription

Home / Explorer

Explorer

Showing 3 entries

Data Source	Group	Metadata Updated	Schemas	Tables	Columns	Tests
Adventure Works: WWDVC Hands-On (Migrated)		8/25/2021 8:02:20 AM	1	13	129	20
Data Vault: WWDVC Hands-On (Migrated)		4/28/2021 12:14:51 PM	4	50	566	9
Snowflake: Permissions Testing (Migrated)	Permissions Testing DS Group	4/28/2021 12:09:42 PM	7	87	1,112	5

Who Uses the Explorer?

Everyone! Whether you are a creator, admin, or viewer, you can find value in Validatar's data quality catalog because it shows you the current state of your data. This real-time snapshot of your data source allows you to discover potential problem areas, pinpoint critical tables and columns, and observe data movement using data lineage.

The data quality catalog isn't just about data, it's about collaboration. Is there a table whose record count has fallen out of acceptable range? Engage with your team using **Discussions** to further investigate the problem. Have you manually loaded a table? Let your team know so they can anticipate changes.

Benefits of a Data Quality Catalog

- Data Discovery
- Team Collaboration
- Improved Data Analysis
- Optimized Data Governance
- Consistent Data Quality
- Monitoring Data Assets

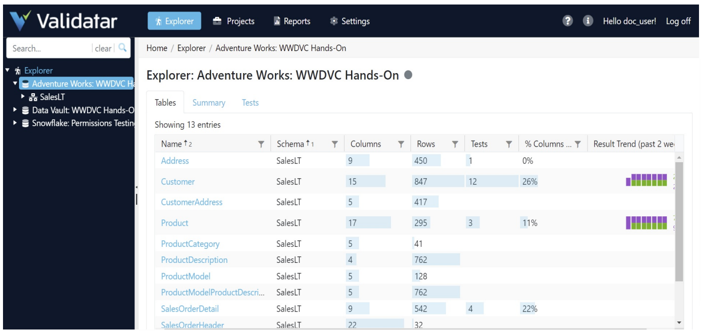
What Can You View

- Profile Results
- Lineage
- Technical Metadata
- Metadata Relationships
- Links to Tests
- Business Rules

Navigating the Explorer

Overview

The Explorer is designed to make viewing metadata simple.



Home / Explorer / Adventure Works: WWDVC Hands-On

Explorer: Adventure Works: WWDVC Hands-On

Showing 13 entries

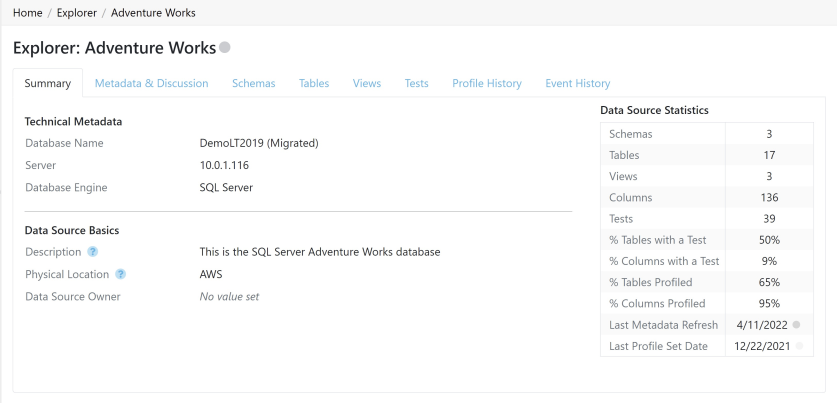
Name ↑2	Schema ↑1	Columns	Rows	Tests	% Columns ...	Result Trend (past 2 we
Address	SalesLT	9	450	1	0%	
Customer	SalesLT	15	847	12	26%	
CustomerAddress	SalesLT	5	417			
Product	SalesLT	17	295	3	11%	
ProductCategory	SalesLT	5	41			
ProductDescription	SalesLT	4	762			
ProductModel	SalesLT	5	128			
ProductModelProductDescri...	SalesLT	5	762			
SalesOrderDetail	SalesLT	9	542	4	22%	
SalesOrderHeader	SalesLT	22	32			

1. Search or choose your data source from the tree on the left.
2. Explore schema, table, and column-level metadata by expanding the tree or using the metadata tabs.
3. View table and column-level data profiles and lineage.
4. View and edit custom metadata fields that provide additional information about the metadata.
5. Engage in discussions with other Validatar users on each metadata object. (Hint: Use **@** to get the conversation started!)

What's On Each Tab

Summary

Every metadata object in the Explorer has a Summary tab. The summary tab lists the technical metadata and statistics related to that object as well as the priority custom fields assigned to that object. Keep your metadata refreshed to receive the latest data while viewing the Summary.



Home / Explorer / Adventure Works

Explorer: Adventure Works

Technical Metadata

Database Name	DemoLT2019 (Migrated)
Server	10.0.1.116
Database Engine	SQL Server

Data Source Basics

Description	This is the SQL Server Adventure Works database
Physical Location	AWS
Data Source Owner	*No value set*

Data Source Statistics

Schemas	3
Tables	17
Views	3
Columns	136
Tests	39
% Tables with a Test	50%
% Columns with a Test	9%
% Tables Profiled	65%
% Columns Profiled	95%
Last Metadata Refresh	4/11/2022
Last Profile Set Date	12/22/2021

Metadata & Discussion

Validatar enables teams to come together to test and discover more about their data. Metadata & Discussion was created for collaboration.

Data Source Metadata

The Metadata section shows all of the custom fields assigned to that object. Custom Field admins can modify custom field values using the **Edit** button though we recommend making bulk changes by importing custom fields on the Data Source.

Why Add Custom Metadata?

Custom Fields help to centralize information related to your data source. Business analysts tend to keep their work separate from data analysts and testers and then share their findings on an as-needed basis. Custom Fields allow business users to share additional context about a metadata object in the same area where data will be tested. This streamlines information to the tester and allows them to enhance their testing using automation. Automation can be driven by business requirements as well as technical requirements with the use of custom fields.

Comments

In addition, creating a discussion thread directly on the metadata object increases collaboration between the team without having to switch communication platforms. Share data issues, changed processes, and other information quickly with your team members. Simply, begin typing in the comment box to add to the discussion. Don't forget to mention (@) your team member so they can also be notified!

Discussion Notifications

Don't forget to turn on your notifications on the User Profile management page.

Metadata Object Tabs

Depending on the type of metadata object you're viewing, you will see different tabs appear after Metadata & Discussion. The viewable tabs will be the next associated child object type. For example, when viewing a schema, the shown tabs will be Tables and Views. When viewing a table, the shown tab will be Columns, and so forth.

You'll also see a Metadata & Discussion tab contextualized by that specific metadata object. That way, you can keep a separate discussion going for each object.

Schemas

The Schema tab shows all schema-related custom fields, technical metadata, and schema statistics. Learn about the number of tables and views inside of the schema and how much of the schema is covered by testing and profiles.

You'll be able to drill down into the Tables and Views tabs to view a list of schema-related objects along with their summary profiles.

Tables & Views

The Tables and Views tabs all show custom fields related to tables and table-level statistics. You'll find similar stats found on the schema level, but you'll also get record counts and table size (data size) if it's been stored.

Users on the Enterprise tier of Validatar will also be able to see a table lineage tab if lineage data has been collected in Validatar.

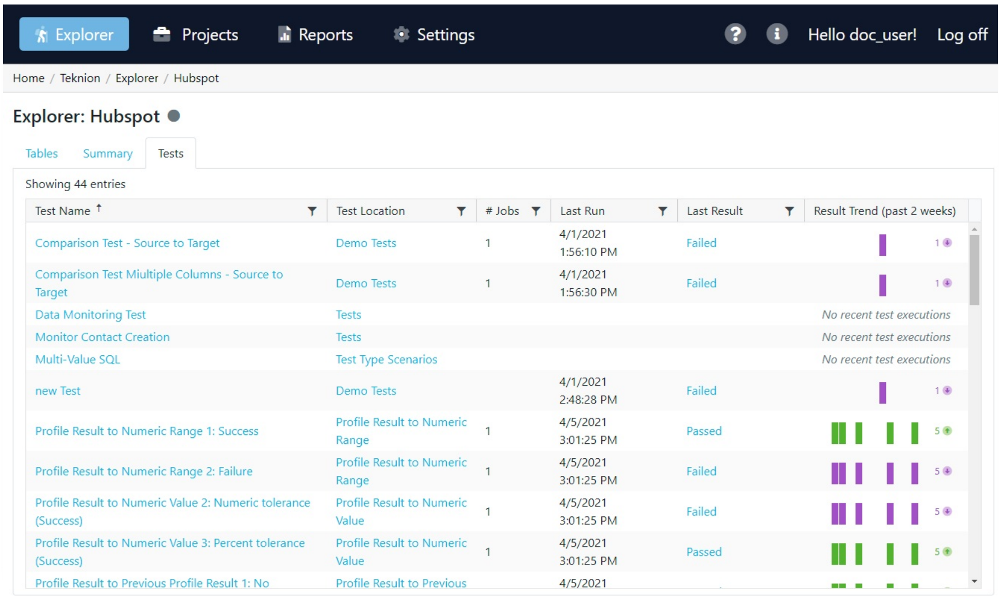
Columns

Once you drill down into a column, you can column-level profile statistics like row count, distinct value count, and null count. Your data must be profiled in order to return any of those stats. You'll also see column-level custom field data. In addition, you'll see a new tab called Column Profile which will show you a full visual representation of your column profiles. Try refreshing your metadata and re-running a profile set to see the changes in profile values.

If you've brought in lineage values for that column, you'll find the Lineage tab along the header as well.

Tests

The Tests tab shows a list of all tests that have either a table or column metadata link related to a data source. Clicking on the test name takes you directly to the test’s configuration screen. You can also see where the test is located within Validatar as well as the pass or fail result trend over the last two weeks.



Explorer: Hubspot ●

Showing 44 entries

Test Name ↑	Test Location	# Jobs	Last Run	Last Result	Result Trend (past 2 weeks)
Comparison Test - Source to Target	Demo Tests	1	4/1/2021 1:56:10 PM	Failed	1
Comparison Test Miultiple Columns - Source to Target	Demo Tests	1	4/1/2021 1:56:30 PM	Failed	1
Data Monitoring Test	Tests				*No recent test executions*
Monitor Contact Creation	Tests				*No recent test executions*
Multi-Value SQL	Test Type Scenarios				*No recent test executions*
new Test	Demo Tests		4/1/2021 2:48:28 PM	Failed	1
Profile Result to Numeric Range 1: Success	Profile Result to Numeric Range	1	4/5/2021 3:01:25 PM	Passed	5
Profile Result to Numeric Range 2: Failure	Profile Result to Numeric Range	1	4/5/2021 3:01:25 PM	Failed	5
Profile Result to Numeric Value 2: Numeric tolerance (Success)	Profile Result to Numeric Value	1	4/5/2021 3:01:25 PM	Failed	5
Profile Result to Numeric Value 3: Percent tolerance (Success)	Profile Result to Numeric Value	1	4/5/2021 3:01:25 PM	Passed	5
Profile Result to Previous Profile Result 1: No	Profile Result to Previous		4/5/2021		

Profile History

To see changes in profile results over time, you'll use the Profile History tab. If you view the profile history for the entire data source, you'll be shown a list of all profile sets that were executed. Viewing profile history on table and column objects allows you to pick which profile you want to view and its past record. Choose between a chart and table view to consume the information.

Event History

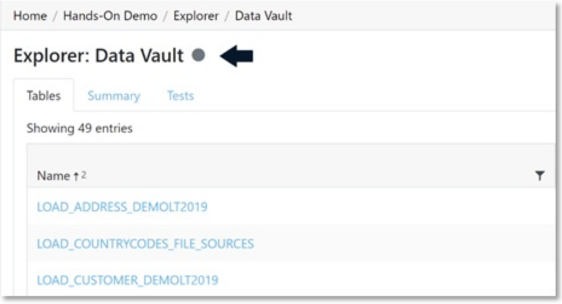
Lastly, Event History keeps a record of the changes made to a metadata object. You can view if the object itself has changed or if someone updated related Validatar information like comments and custom fields.

Understanding the Data Profile

Overview

Depending on the data type of the column that was profiled, the returned statistics may vary. At the top of the profile, you'll always see technical metadata. Technical metadata includes the column name, data type, number of rows, length (for strings), and more.

The data profile is populated by running profile sets on a data source. For this reason, each data profile can be updated at different times depending on the content of a profile set. To view the recency of a profile statistic, hover over the gray circle next to the name. The color of the gray circle represents how recent the metadata is. The darker the circle, the more recent.



Home / Hands-On Demo / Explorer / Data Vault

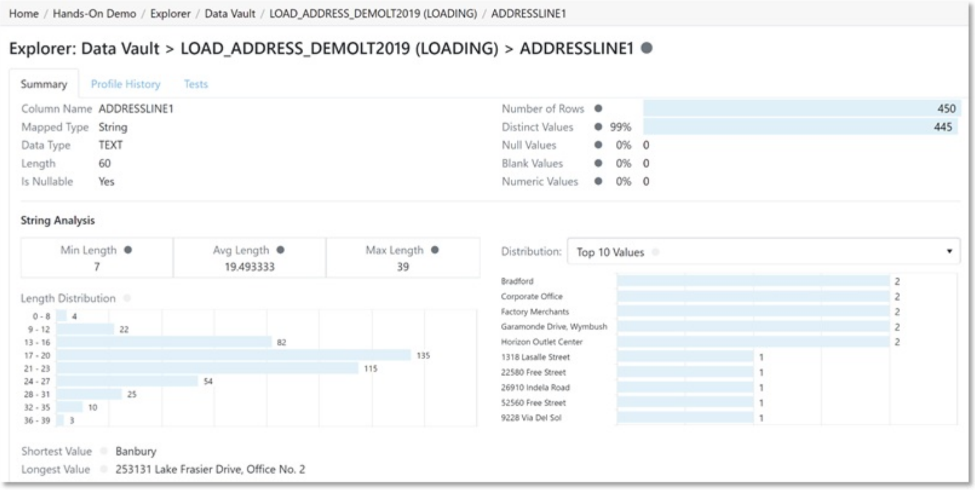
Explorer: Data Vault ●

Showing 49 entries

Name ↑2
LOAD_ADDRESS_DEMOLT2019
LOAD_COUNTRYCODES_FILE_SOURCES
LOAD_CUSTOMER_DEMOLT2019

The profile only shows the results of profile sets that were executed. If you're missing profile information, go back into the Data Source and create a profile set containing the chosen data profile(s) on the appropriate objects.

String Profiles



Home / Hands-On Demo / Explorer / Data Vault / LOAD_ADDRESS_DEMOLT2019 (LOADING) / ADDRESSLINE1

Explorer: Data Vault > LOAD_ADDRESS_DEMOLT2019 (LOADING) > ADDRESSLINE1 ●

Column Name	ADDRESSLINE1
Mapped Type	String
Data Type	TEXT
Length	60
Is Nullable	Yes

Number of Rows	●		450
Distinct Values	●	99%	445
Null Values	●	0%	0
Blank Values	●	0%	0
Numeric Values	●	0%	0

String Analysis

Min Length ●	Avg Length ●	Max Length ●
7	19.493333	39

Length Distribution

0 - 8	4
9 - 12	22
13 - 16	82
17 - 20	135
21 - 23	115
24 - 27	54
28 - 31	25
32 - 35	10
36 - 39	3

Distribution: Top 10 Values

Bradford	2
Corporate Office	2
Factory Merchants	2
Garamonde Drive, Wymbush	2
Horizon Outlet Center	2
1318 Lasalle Street	1
22580 Free Street	1
26910 Indela Road	1
52560 Free Street	1
9228 Via Del Sol	1

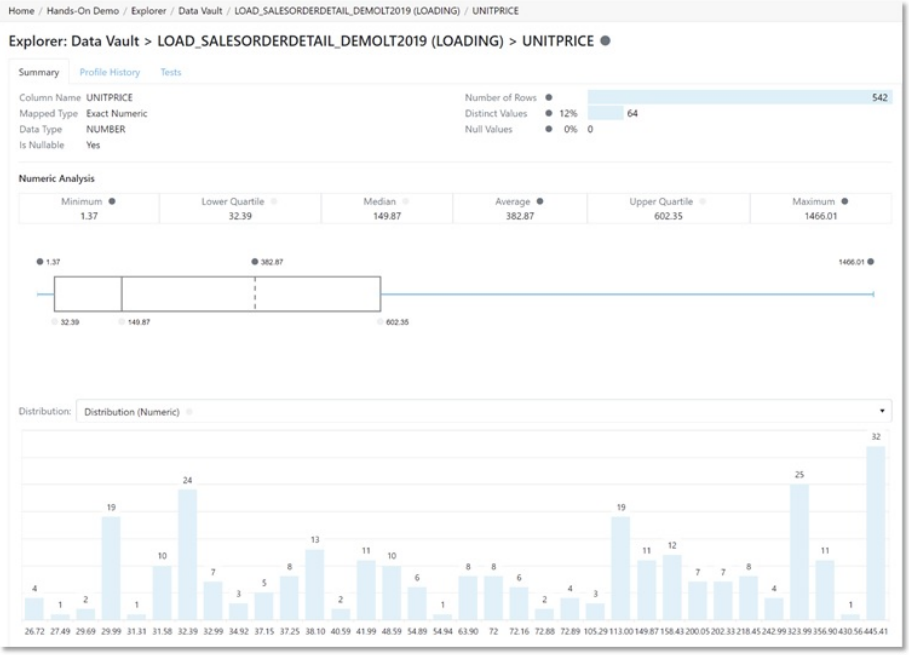
Shortest Value Banbury

Longest Value 253131 Lake Frasier Drive, Office No. 2

String profiles show the following information.

Profile Type	Data Profile
Summary	• Length • Is Nullable • Number of Rows • Number, Percent of Distinct Values • Number, Percent of Null Values • Number, Percent of Blank Values • Number, Percent of Numeric Values
Analysis	• Minimum Length • Average Length • Maximum Length
Distribution	• Length Distribution • Top 10 Values • Bottom 10 Values
Length	• Shortest Value • Longest Value

Numeric Profiles



Home / Hands-On Demo / Explorer / Data Vault / LOAD_SALESORDERDETAIL_DEMOLT2019 (LOADING) / UNITPRICE

Explorer: Data Vault > LOAD_SALESORDERDETAIL_DEMOLT2019 (LOADING) > UNITPRICE ●

Column Name UNITPRICE
Mapped Type Exact Numeric
Data Type NUMBER
Is Nullable Yes

Number of Rows ● 542
Distinct Values ● 12% 64
Null Values ● 0% 0

Numeric Analysis

Minimum ●	Lower Quartile	Median	Average ●	Upper Quartile	Maximum ●
1.37	32.39	149.87	382.87	602.35	1466.01

Distribution: Distribution (Numeric)

Numeric profiles show the following information.

Profile Type	Data Profile
Summary	• Is Nullable • Number of Rows • Number, Percent of Distinct Values • Number, Percent of Null Values
Analysis	• Minimum • Maximum • Average • Lower Quartile • Upper Quartile • Median
Distribution	• Numeric Distribution • Top 10 Values • Bottom 10 Values

Date/Time Profiles

Date/Time profiles show the following information.

Profile Type	Data Profile
Summary	• Is Nullable • Number of Rows • Number, Percent of Distinct Values • Number, Percent of Null Values
Analysis	• Minimum Date • Maximum Date
Distribution	• Top 10 Values • Year Distribution • Year Month Distribution

What is Data Lineage?

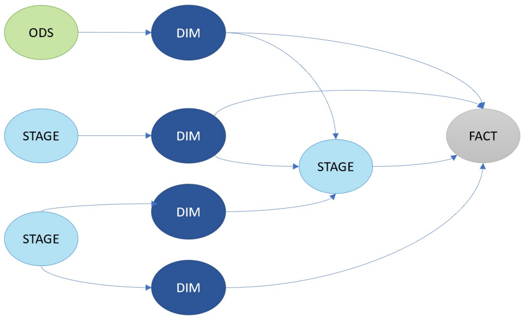
Overview

Data lineage is used to track the movement of data over time. As data is merged from multiple sources using multiple deployment processes, it's important to trace where your data has come from and where it's going. One of the biggest reasons to track lineage is to be able to identify the root cause of issues which reduces the time spent investigating the problem.

Data lineage supports data analysis, architecting, development, and testing by following data transformations' what, where, and how. Simply, lineage provides you a complete look at your data's journey.

The Value of Data Lineage

As data advances along the path to end-user consumption, it will interact with other pieces of information along the way. These points of interaction determine if the data continues to move forward, is transformed, or if the journey has ended. With the complexity of modern data architect solutions, tracking lineage has become essential to the data development and management process.



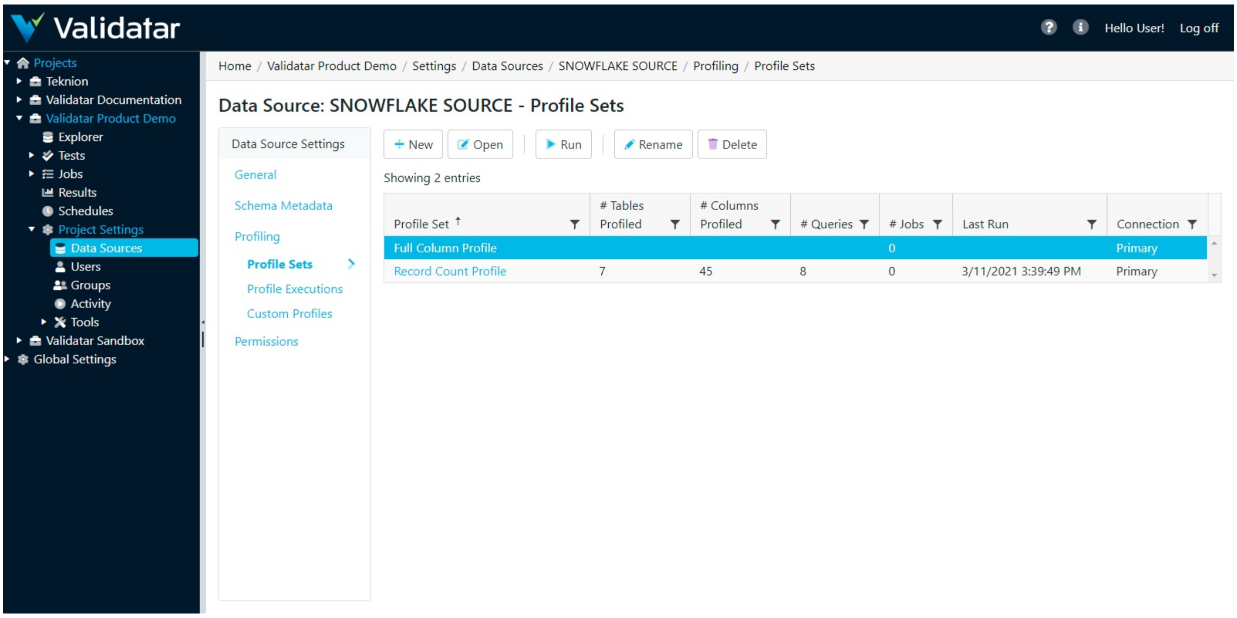
Lineage allows organizations to predict how changes to their ETL or business processes will impact their data environment and end-user reporting. Having a tool that automates lineage consumption enables your business to make quicker and better decisions which save time and resources and also increases trust in data operations.

Profile Sets

Overview

A profile set is a collection of data profiles that are run together to provide specific information about a data source.

Displayed Information



Data Source: SNOWFLAKE SOURCE - Profile Sets

Showing 2 entries

Profile Set	# Tables Profiled	# Columns Profiled	# Queries	# Jobs	Last Run	Connection
Full Column Profile				0		Primary
Record Count Profile	7	45	8	0	3/11/2021 3:39:49 PM	Primary

Action Buttons

New	Creates a new profile set.
Open	Opens an existing profile set.
Run	Allows you to run a profile set.
Rename	Renames a profile set.
Delete	Deletes a profile set.
Restore	Restores a deleted profile set. After you've deleted a profile set, you cannot restore it once you navigate away from the Profile Set screen.

Table Headers

Profile Set	The name of the profile set.
# Tables Profiled	The number of tables that the set profiles.
# Columns Profiled	The number of columns that the set profiles.
# Queries	The number of queries that will be executed when you run the profile set.
# Jobs	The number of jobs that include the profile set as a job step.
Last Run	The time and date of the last profile set execution.
Connection	Displays whether the profile set uses the primary connection or a specific profile connection.

Access

1. Click on a data source name on the Data Source page in Project Settings.
2. Click on Profile Sets.

Configuring a Profile Set

Overview

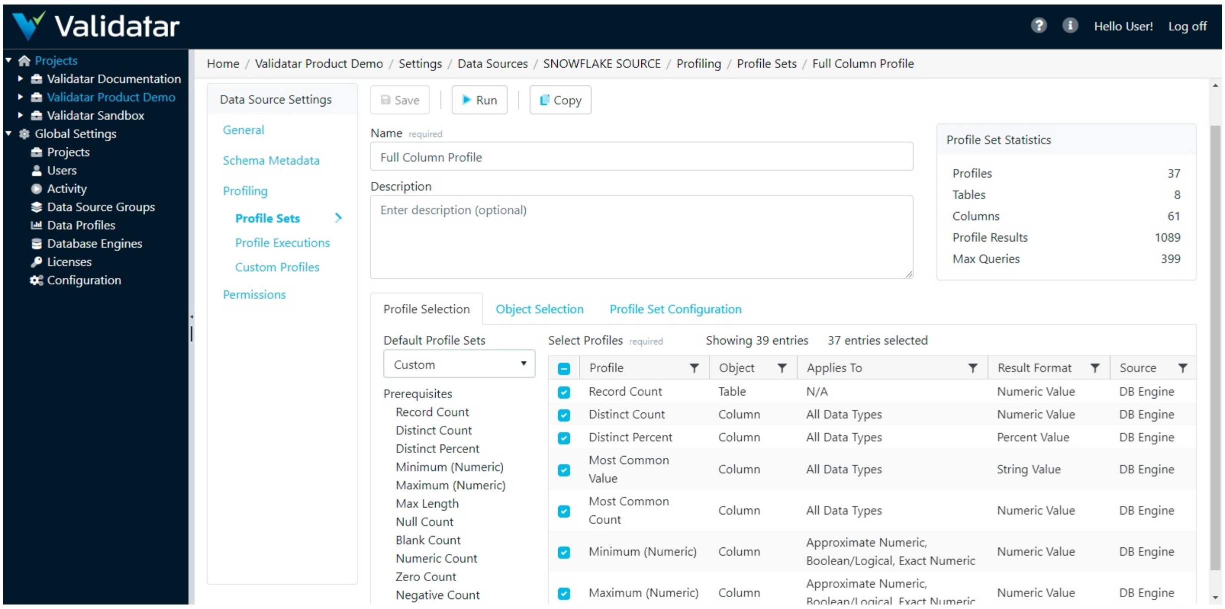
The Profile Set configuration screen is separated into three tabs:

1. Profile Selection
2. Object Selection
3. Profile Set Configuration

Profile Selection

The profile selection tab is where you select the data profile you want to include in a profile set out of a list of data profiles available for that respective data source to use. The list of available data profiles depends on which profiles are enabled for the respective database engine. See more in Global Settings – Database Engines.

Displayed Information



Name required: Full Column Profile

Description: Enter description (optional)

Profile Set Statistics

Profiles	37
Tables	8
Columns	61
Profile Results	1089
Max Queries	399

Default Profile Sets: Custom

Prerequisites
- Record Count
- Distinct Count
- Distinct Percent
- Minimum (Numeric)
- Maximum (Numeric)
- Max Length
- Null Count
- Blank Count
- Numeric Count
- Zero Count
- Negative Count

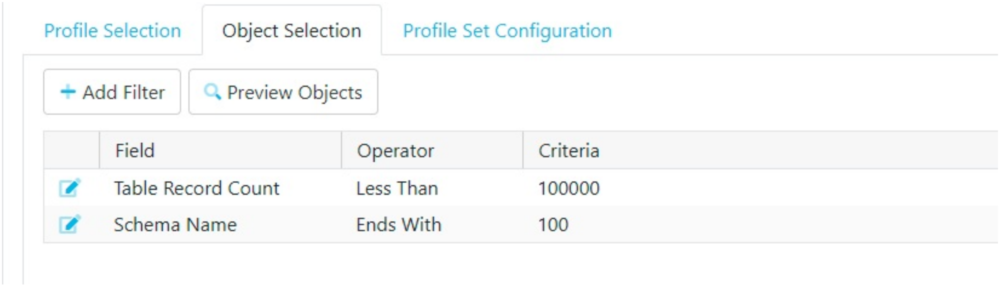
Select Profiles required   Showing 39 entries   37 entries selected

☐	Profile	Object	Applies To	Result Format	Source
☑	Record Count	Table	N/A	Numeric Value	DB Engine
☑	Distinct Count	Column	All Data Types	Numeric Value	DB Engine
☑	Distinct Percent	Column	All Data Types	Percent Value	DB Engine
☑	Most Common Value	Column	All Data Types	String Value	DB Engine
☑	Most Common Count	Column	All Data Types	Numeric Value	DB Engine
☑	Minimum (Numeric)	Column	Approximate Numeric, Boolean/Logical, Exact Numeric	Numeric Value	DB Engine
☑	Maximum (Numeric)	Column	Approximate Numeric, Boolean/Logical, Exact Numeric	Numeric Value	DB Engine

Default Profile Sets	Validatar ships with three pre-configured profile sets: • Table Profile: Selects all profiles that are run on a table level. E.g., table record count • Full Column Profile: Selects all profiles that are run on a column level. E.g., distinct count or null count • Basic Column Profile: Selects a smaller number of column-level profiles to run.
Custom Profile Set	To customize a profile set, check any box to the left of the profile name to add it to the set.
Prerequisites	Prerequisites appear when you have added a profile to the profile set and that profile is dependent on another profile to be run first. For example, if you want to run a Distinct Count profile you must include the Record Count profile in the profile set. To know what the distinct count is, you must have the total record count for that table. Any missing prerequisites are highlighted in orange and can be added by simply selecting the *add plus sign screenshot* button.

Object Selection

Displayed Information



	Field	Operator	Criteria
✎	Table Record Count	Less Than	100000
✎	Schema Name	Ends With	100

Note

Add filters on your profile sets to limit the number of objects that will be profiled. View the Profile Set Statistics box as you add filters so you can see how extensive running the profile set will be on your database.

Filter Options

Field	Operator	Criteria
Schema Name	String Filters	Any string input
Table Name	String Filters	Any string input
Table Record Count	Numeric Filters	Any numeric input
Table Type	Equals, Not Equals	Table or View
Column Name	String Filters	Any string input
Column Data Type	String Filters	Any string input
Column Mapped Type	Equals, Not Equals	Validatar data types
Column Nullable	Equals, Not Equals	Yes or No
Column Identity	Equals, Not Equals	Yes or No
Column Primary Key	Equals, Not Equals	Yes or No
Column Max Length	Numeric Filters	Any numeric input

Action Buttons

Add Filter	Add an object filter on the data profile set.
Preview Objects	Preview the objects that will be run on the data profile set.

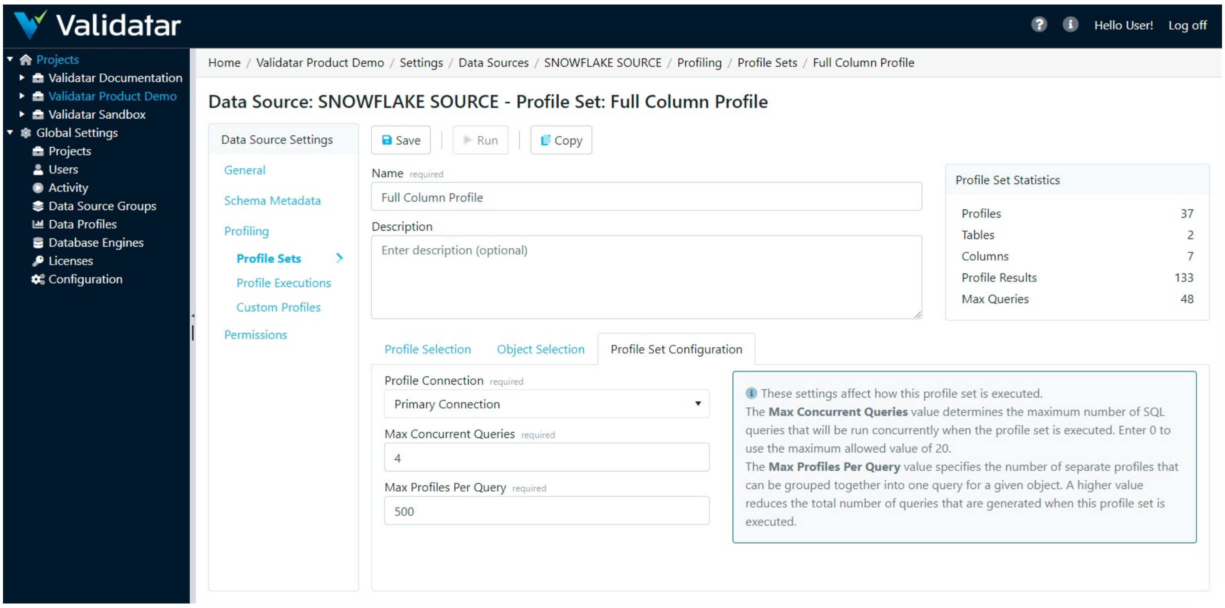
Table Headers

Field	The type of object filter.
Operator	The comparison operator used for the object filter.
Criteria	The criteria must be met by an object to be included in the filter.

Profile Set Configuration

The profile set configuration tab contains advanced configuration settings that affect how a profile set is executed.

Displayed Information



Data Source: SNOWFLAKE SOURCE - Profile Set: Full Column Profile

Name required: Full Column Profile

Description: Enter description (optional)

Profile Set Statistics

Profiles	37
Tables	2
Columns	7
Profile Results	133
Max Queries	48

Profile Connection required: Primary Connection

Max Concurrent Queries required: 4

Max Profiles Per Query required: 500

These settings affect how this profile set is executed.
The **Max Concurrent Queries** value determines the maximum number of SQL queries that will be run concurrently when the profile set is executed. Enter 0 to use the maximum allowed value of 20.
The **Max Profiles Per Query** value specifies the number of separate profiles that can be grouped together into one query for a given object. A higher value reduces the total number of queries that are generated when this profile set is executed.

Profile Connection	Select from the dropdown whether to use the primary connection or the specified profile connection.
Max Concurrent Queries	Determines the maximum number of SQL queries that will be run concurrently when the profile set is executed. The value can range from 0 to 20.
Max Profiles Per Query	Specifies the number of separate profiles that can be grouped together into one query for a given object. A higher value reduces the total number of queries that are generated when this profile set is executed.
Purge Results After Days	Choose to purge profile results after a certain number of days.

40 Standard Data Profiles

Overview

Validatar ships with 40 data profiles already configured. Each one can be modified at both the management level and definition level within a database engine.

1. **Record Count**
    - Table profile
    - Returns the record count of the table.
    - Result Format: Numeric Value
2. **Total Data MB**
    - Table profile
    - Returns the total size of the table in megabytes.
    - Result Format: Numeric Value
3. **Distinct Count**
    - Column profile
    - Returns the distinct number of records in the column.
    - Result Format: Numeric Value
4. **Distinct Percent**
    - Column profile
    - Prerequisites: Distinct Count, Record Count
    - Returns the distinct count divided by the total number of records.
    - Result Format: Percent Value
5. **Most Common Value**
    - Column profile
    - Returns the most frequently used value.
    - Result Format: String
6. **Most Common Count**
    - Column profile
    - Returns the most frequently used number.
    - Result Format: Numeric Value
7. **Minimum (Numeric)**
    - Column profile
    - Returns the minimum value of a numeric column.
    - Result Format: Numeric Value
8. **Maximum (Numeric)**
    - Column profile
    - Returns the maximum value of a numeric column.
    - Result Format: Numeric Value
9. **Mean (Numeric)**
    - Column profile
    - Returns the average value of a numeric column.
    - Result Format: Numeric Value
10. **Median (Numeric)**
    - Column profile
    - Returns the median value of a numeric column.
    - Result Format: Numeric Value
11. **Lower Quartile**
    - Column profile
    - Returns the middle value between the lowest value and the median of the column.

- Result Format: Numeric Value

12. **Upper Quartile**
    - Column profile
    - Returns the middle value between the median and the highest value of the column.
    - Result Format: Numeric Value

13. **Minimum (String)**
    - Column profile
    - Returns the minimum value of a string column.
    - Result Format: String Value

14. **Maximum (String)**
    - Column profile
    - Returns the maximum value of a string column.
    - Result Format: String Value

15. **Standard Deviation**
    - Column profile
    - Returns the measure of variance in a numeric column.
    - Result Format: Numeric Value

16. **Max Length**
    - Column profile
    - Returns the maximum length of a value in a string column.
    - Result Format: String Value

17. **Min Length**
    - Column profile
    - Returns the minimum length of a value in a string column.
    - Result Format: String Value

18. **Mean Length**
    - Column profile
    - Returns the average length of a value in a string column.
    - Result Format: String Value

19. **Length Distribution**
    - Column profile
    - Returns a list of lengths and their count of occurrences in a string column.
    - Result Format: Numeric Array

20. **Null Count**
    - Column profile
    - Returns the count of null values in a column.
    - Result Format: Numeric Value

21. **Null Percent**
    - Column profile
    - Returns the count of null values divided by the total record count in a column.
    - Result Format: Percent Value

22. **Blank Count**
    - Column profile
    - Returns the count of blank values in a column.
    - Result Format: Numeric Value

23. **Blank Percent**
    - Column profile
    - Returns the count of blank values divided by the total record count in a column.
    - Result Format: Percent Value

24. **Numeric Count**
    - Column profile

- Returns the count of numeric values in a string column.
- Result Format: Numeric Value

25. **Numeric Percent**
    - Column profile
    - Returns the count of numeric values divided by the total record count in a string column.
    - Result Format: Percent Value

26. **Zero Count**
    - Column profile
    - Returns the count of values that equal 0 in a numeric column.
    - Result Format: Numeric Value

27. **Zero Percent**
    - Column profile
    - Returns the count of values that equal 0 divided by the total record count in a numeric column.
    - Result Format: Percent Value

28. **Negative Count**
    - Column profile
    - Returns the count of values less than 0 in a numeric column.
    - Result Format: Numeric Value

29. **Negative Percent**
    - Column profile
    - Returns the count of values less than 0 divided by the total record count in a numeric column.
    - Result Format: Percent Value

30. **Minimum (Date)**
    - Column profile
    - Returns the earliest date in a column.
    - Result Format: Date/Time Value

31. **Maximum (Date)**
    - Column profile
    - Returns the most recent date in a column.
    - Result Format: Date/Time Value

32. **Longest Value**
    - Column profile
    - Returns the value with the longest length in a string column.
    - Result Format: String Value

33. **Shortest Value**
    - Column profile
    - Returns the value with the shortest length in a string column.
    - Result Format: String Value

34. **Distribution (String)**
    - Column profile
    - Returns a list of values and their count of occurrences in a string column.
    - Result Format: String Array

35. **Distribution (Numeric)**
    - Column profile
    - Returns a list of values and their count of occurrences in a numeric column.
    - Result Format: Numeric Array

36. **Top 10 Values**
    - Column profile
    - Returns the 10 most common values in a column.
    - Result Format: String Array

37. **Bottom 10 Values**

- Column profile
- Returns the 10 least common values in a column.
- Result Format: String Array

38. **Binned (Numeric)**
    - Column profile
    - Returns the grouping of values into bins for a numeric column.
    - Result Format: Numeric Array

39. **Year Distribution**
    - Column profile
    - Returns a list of years and their count of occurrences in a Date/Time column.
    - Result Format: String Array

40. **Year Month Distribution**
    - Column profile
    - Returns a list of years and months and their count of occurrences in a Date/Time column.
    - Result Format: String Array

Configuring Custom Fields

Permissions Required

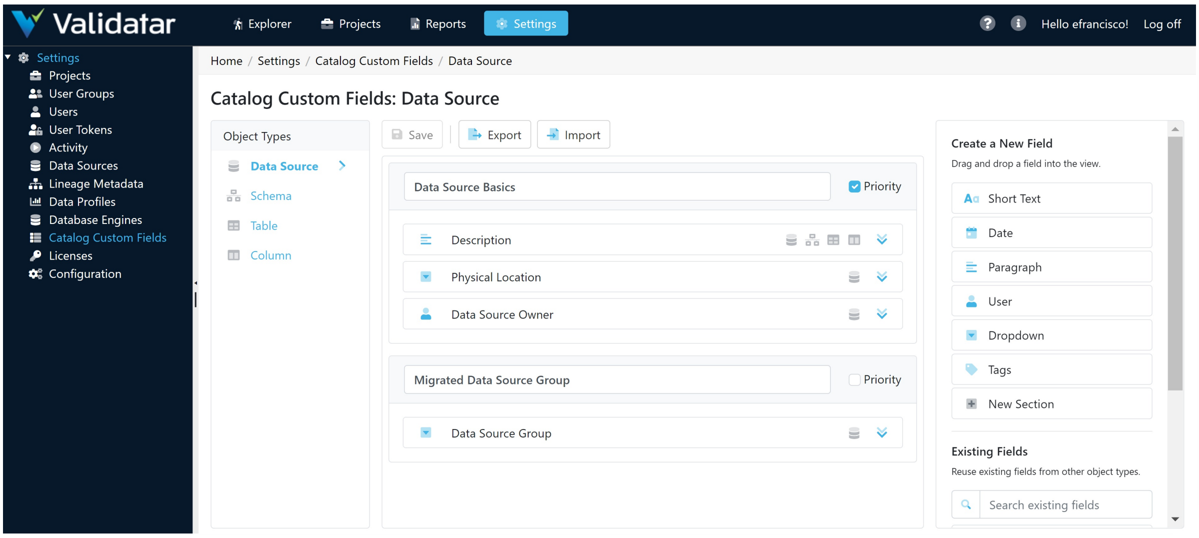
A user

- must be a **Custom Metadata Admin** to create and modify custom fields configuration.
- must have **Browse Metadata and Data Profiling** permissions for a data source to view catalog custom fields in the Explorer. View the full list of custom fields using the **Metadata & Discussion** tab on each metadata object level.
- must have **Edit Metadata** permission for a data source to populate and modify custom field values.

Custom Fields appear on the following Validatar Editions:

Edition	Custom Field Limits
Validatar Teams	Create up to 20
Validatar Enterprise	Unlimited

Using the Configuration Page

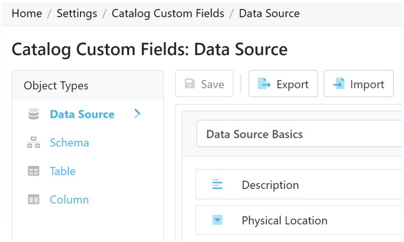


How it Works

1. Navigate to **Settings**. Then choose whether to configure **Catalog** or **Project Custom Fields**.
2. Use the Object Types pane to navigate to the object in which the custom field should be applied.
3. From the right pane, drag and drop a **New Section**into the view.
    1. Note: Fields **cannot** be created without a section.
4. Review the field types and drag and drop the appropriate **New Field** into a section.
5. Populate the field information. Only the name and reference key are required.
6. Click **Save** before navigating to a new object type.

Learn more about each component of a custom field below.

Object Types



Home / Settings / Catalog Custom Fields / Data Source

Catalog Custom Fields: Data Source

Object Types

- Data Source
- Schema
- Table
- Column

Save | Export | Import

Data Source Basics

- Description
- Physical Location

Object types are the metadata or Validatar objects where custom fields can be applied. These objects are:

- Data Sources, Schemas, Tables, Columns
- Projects, Tests, Jobs, Labels

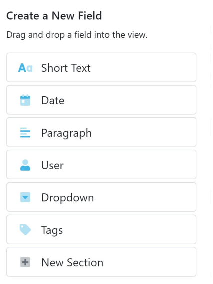
Custom Fields created in **Settings** will automatically be applied to every data source in Validatar and its respective metadata objects unless you add logic that determines when to show the field.

Creating Sections

A section is a logical grouping of custom fields. Each custom field must be placed in a section, and a section name cannot be duplicated. If there are any duplicates, you must rename the section before saving it.

To create a section, drag and drop the **New Section** field into the view from the Create a New field pane on the right. Some sections will hold more significant information and require more visibility than others. To identify those sections, select the **Priority** checkbox next to the section name. Priority Sections will appear on the Summary tab of each metadata object in the Explorer.

Field Types



Create a New Field

Drag and drop a field into the view.

- Short Text
- Date
- Paragraph
- User
- Dropdown
- Tags
- New Section

Building a custom field starts with the field type. There a six types of custom fields:

Field Type	Definition
Short Text	Up to 300 characters of text
Date	A date in M/D/YYYY format
Paragraph	Up to 10,000 characters of text
User	One or more Validatar users or user groups
Dropdown	A single selection of defined options
Tags	One or more key terms assigned to information

Creating a New Field

Drag and drop the chosen field type into a section from the **Create a New Field** pane on the right to create a new field. Enter the name and description of the field. The reference key, used to identify custom fields in tests, is automatically populated from the field name, removing special characters and replacing spaces with underscores.

If you create a field that you want to remove, click **Delete** and the bottom of the field.

Create all of the necessary custom fields on an object type, then **Save**. You must save before you can move onto another object type.

Dropdown Fields

Dropdowns are the only field type that requires further configuration than name and description. You must provide a list of valid options that can be selected in the dropdown.

On the field card, click **Add Options** under Dropdown Options and enter each valid value. Drag and drop each option to reorder them. Click the **x** to remove an option.

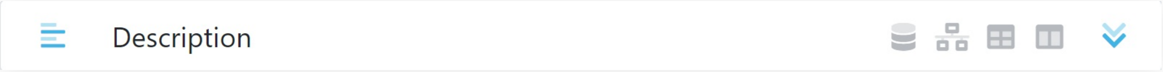
View how to create a section and dropdown custom field below.

Using Existing Fields

Once a field is created and saved, it becomes an existing field and can be used on other object types. The name, reference key, and description are brought into the new object type. This keeps the user from having to create duplicate fields.

For example, if you created a custom field called **Technical Data Steward** on the schema object type, you can reuse that field on the data source, table, and/or column object types.

Identify which object types the custom field is used by the icons next to the name. For example, the **Description** field is found on the data source, schema, table, and column object types since all four icons are next to the name.



Archived Fields

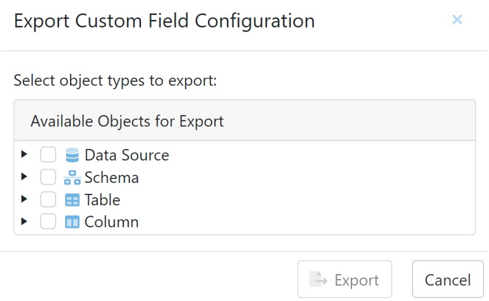
Newly created fields, meaning they haven't been saved, can be deleted. You cannot delete an object after saving it, but you can archive it. Archived fields can be restored by checking the **Show archived fields** checkbox in the right pane and dragging and dropping the field back into the view.

Custom Field Import & Export

Overview

Import and Export is used to easily apply custom fields to other object types.

Export



Export Custom Field Configuration

Select object types to export:

Available Objects for Export

- □ Data Source
- □ Schema
- □ Table
- □ Column

To export custom fields,

1. Navigate to **Settings** > **Catalog Custom Fields**.
2. Click **Export** in the toolbar.
3. Choose which objects you want to export.
4. Click **Export**.

Import

To import custom fields,

1. Navigate to **Settings** > **Catalog Custom Fields**.
2. Click **Import** in the toolbar.
3. Select the exported file. Click **Next**.
4. In the **Import Custom Field** pane on the right, choose the object type from the dropdown that you want to import **OR** choose to **Import All** custom fields. Only object types included in the file will be included in the dropdown.
5. Drag and drop the field(s) into the view.
6. Switch between each object type, dropping the respective fields into view until completion..
7. Click **Complete**

During import, fields can be reordered and edited like standard custom fields. There cannot be duplicate fields on the same object type so a warning will be displayed on the card if this happens.

Imported fields can be deleted from the view before completing the import by clicking **Delete** at the bottom of the field card.

Custom Metadata Import

Overview

Custom fields provide additional information about data sources. Custom fields are first created in Settings and are applied to every Validatar data source. You must navigate to Data Source settings to populate the values for these custom fields but a data steward, for example, can change these values directly in the Explorer.

To add this additional information to a Validatar data source, you must import the values via file or SQL ingestion. We recommend populating custom metadata using one of these methods initially over editing the metadata directly in the Explorer.

Accessing the Import

1. Navigate to **Settings > Data Sources**.
2. Open the respective data source.
3. On the Data Source Settings pane, click **Custom Metadata**.
4. You can then choose between a File Import or SQL Ingestion.

File Import

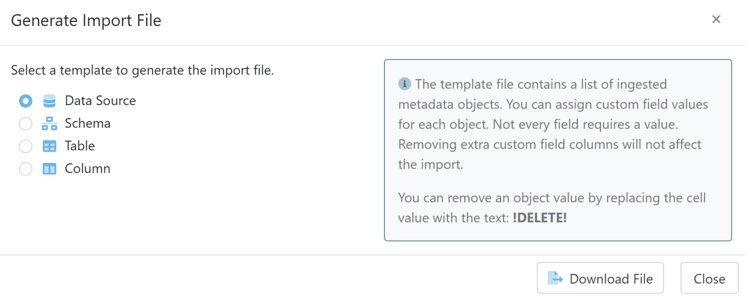
One way to import custom fields into a data source is via an Excel file.

Performing the Import

To import:

1. Click the **Generate Import File** button. Learn more about the file template in the Generating an Import File section below.
2. Select the object type you want to import metadata for.
3. Click **Download File**.
4. Open the downloaded file and begin to fill out the spreadsheet. Learn more about entering field values in the Configuring the Import File section below.
5. Save the Excel file.
6. In Validatar, go back to the File Import tab and click **Upload Import File**.
7. Select the file.
8. Click **Import**.
9. The file will be added to the Event History log.
10. Verify the custom metadata was imported by navigating to the Explorer. Open the respective data source and locate the data objects, or use the search bar to find the objects. The imported custom metadata can be found on the **All Metadata** tab.

Generating an Import File



Generate Import File ×

Select a template to generate the import file.

● Data Source
○ Schema
○ Table
○ Column

The template file contains a list of ingested metadata objects. You can assign custom field values for each object. Not every field requires a value. Removing extra custom field columns will not affect the import.

You can remove an object value by replacing the cell value with the text: **!DELETE!**

We recommend generating the file from Validatar because it has all of the required headers for a successful import. You can choose one of the following object types to import:

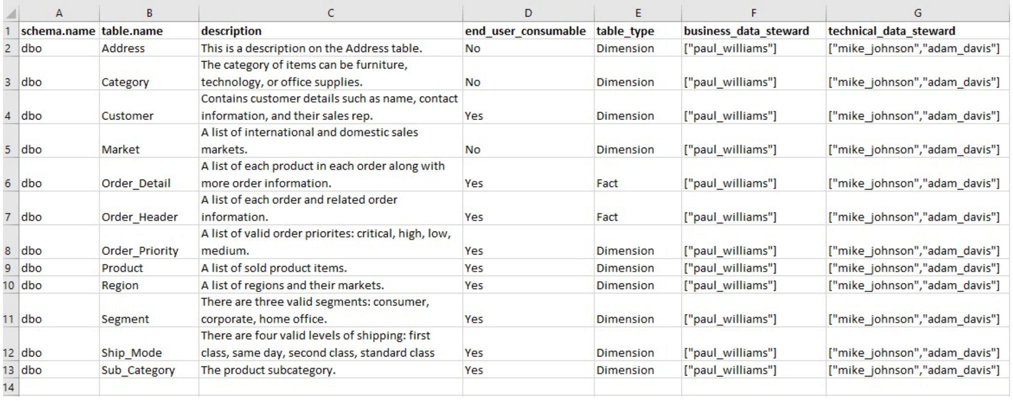
- Data Source
- Schema

- Table
- Column

The template file contains a list of all the ingested metadata objects in the data source based on the selected object type. There will also be one column for each global custom field in Validatar in the template file. You can then assign custom field values to each object.

Need a refresher on what custom fields are in Validatar? Navigate to **Settings** > **Custom Fields** to see the full list.

Configuring the Import File



	A	B	C	D	E	F	G
1	**schema.name**	**table.name**	**description**	**end_user_consumable**	**table_type**	**business_data_steward**	**technical_data_steward**
2	dbo	Address	This is a description on the Address table.	No	Dimension	["paul_williams"]	["mike_johnson","adam_davis"]
3	dbo	Category	The category of items can be furniture, technology, or office supplies.	No	Dimension	["paul_williams"]	["mike_johnson","adam_davis"]
4	dbo	Customer	Contains customer details such as name, contact information, and their sales rep.	Yes	Dimension	["paul_williams"]	["mike_johnson","adam_davis"]
5	dbo	Market	A list of international and domestic sales markets.	No	Dimension	["paul_williams"]	["mike_johnson","adam_davis"]
6	dbo	Order_Detail	A list of each product in each order along with more order information.	Yes	Fact	["paul_williams"]	["mike_johnson","adam_davis"]
7	dbo	Order_Header	A list of each order and related order information.	Yes	Fact	["paul_williams"]	["mike_johnson","adam_davis"]
8	dbo	Order_Priority	A list of valid order priorites: critical, high, low, medium.	Yes	Dimension	["paul_williams"]	["mike_johnson","adam_davis"]
9	dbo	Product	A list of sold product items.	Yes	Dimension	["paul_williams"]	["mike_johnson","adam_davis"]
10	dbo	Region	A list of regions and their markets.	Yes	Dimension	["paul_williams"]	["mike_johnson","adam_davis"]
11	dbo	Segment	There are three valid segments: consumer, corporate, home office.	Yes	Dimension	["paul_williams"]	["mike_johnson","adam_davis"]
12	dbo	Ship_Mode	There are four valid levels of shipping: first class, same day, second class, standard class	Yes	Dimension	["paul_williams"]	["mike_johnson","adam_davis"]
13	dbo	Sub_Category	The product subcategory.	Yes	Dimension	["paul_williams"]	["mike_johnson","adam_davis"]
14							

Custom field values must be entered in the proper format to be evaluated correctly. Use the following rules to fill in the custom field values:

Field Type	Configuration Rules
Short Text	Enter any text up to 300 characters.
Paragraph	Enter any text up to 10,000 characters.
Date	Enter a date in one of the following formats: • MM/DD/YYYY • M/YYYY • YYYY-MM-DD • DD Month YYYY
User	Enter the person's Validatar username in square brackets [ ] wrapped in double quotation marks " ". For multiple users, use a comma separator in between users. For example, **["jsmith"]** or **["jsmith","ajacobs","bwilliams"]**
Dropdown	The dropdown options were defined when the custom fields were configured. Enter one of the dropdown options in the cell.
Tags	Enter the tag in square brackets [ ] wrapped in double quotation marks " ". For multiple tags, use a comma as a separator. For example, **["Entity"]** or **["Entity","Dimension"]**

Deleting Field Values

If you do not have a value to assign to an object, leave the cell blank. To remove an object value, replace the cell value with "**!DELETE!**", no quotation marks.

SQL Ingestion

The second way to import custom metadata into a data source is by writing a SQL ingestion script.

Configuring the Connection

Unlike using an import file, SQL ingestion requires a connection to a database. You can use the current connection by checking **Same As Primary** or you can configure a different data source connection. To configure a connection click **Add**. Learn more about how to configure a connection here.

The database should contain a table with the required metadata object type and custom fields as column headers. Each row should represent one metadata object and the custom field values related to that object.

Here's an example of what Table-Level ingestion table would look like:

schema.name	table.name	description	table_type	business_data_steward	technical_data_steward
dbo	Category	The category of items can be furniture, technology, or office supplies.	Dimension	["paul_williams"]	["mike_johnson","adam_davis"]
dbo	Customer	Contains customer details such as name, contact information, and their sales rep.	Dimension	["paul_williams"]	["mike_johnson","adam_davis"]
dbo	Product	A list of products and their details.	Dimension	["paul_williams"]	["mike_johnson","adam_davis"]
dbo	SalesOrderDetail	A historical record of each product sold in a sales order.	Fact	["paul_williams"]	["mike_johnson","adam_davis"]

You must establish a database connection before you can configure the ingestion SQL.

Configuring the SQL Script

Each object type requires its own SQL ingestion statement. To configure the script, click **Configure Ingestion SQL** and navigate to the appropriate object type by using the tabs at the top of the pop-up window. Check the **Run Script** checkbox to begin writing your SQL statement. There will be a section that lists what fields are required in the query and another section that lists all of the available custom fields you can choose from. Drag and drop the fields into the view for easier script building.

Using the same Table-Level example from above, the structure of the SQL statement will be similar to:

```PL/SQL
select
meta_schema AS [schema.name],
meta_table AS [table.name],
[description],
[table_type],
[business_data_steward],
[technical_data_steward]
from dbo.CustomFieldValues;
```

Things to Remember

- Schema-level SQL ingestion requires [schema.name] to be used in the query.
- Table-level SQL ingestion requires [schema.name] and [table.name] to be used in the query.
- Column-level SQL ingestion requires [schema.name], [table.name], and [column.name] to be used in the query.
- If the column names in the table don't match the required or custom field names, alias the column using the AS command.

Setting a Refresh Schedule

Setting a refresh schedule allows for custom metadata to be automatically updated. Only the metadata object types that are enabled to run will execute when the schedule kicks off.

Event History

The Event History log records each attempt to import custom metadata into Validatar. Each record is for a single object type and shows whether the import was successful or not. To view the details of unsuccessful errors, click the error icon on the record.

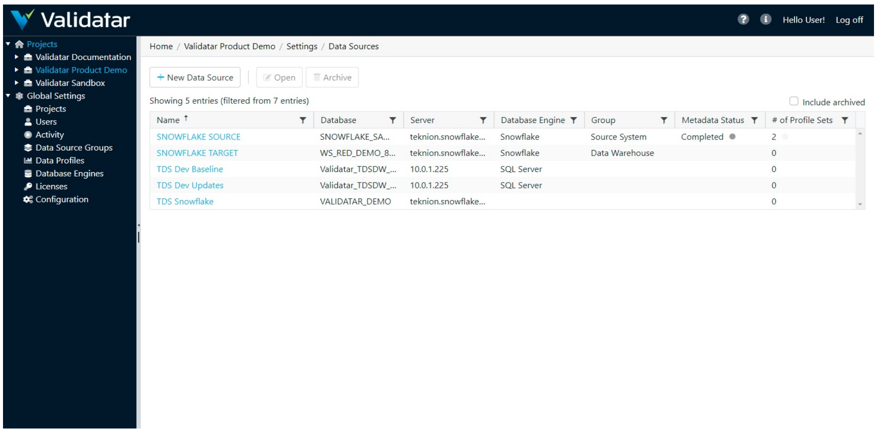
What Is a Data Source?

Overview

A data source is what you use to run Validatar tests and profiles against. A data source consists of:

- A primary database connection
- A metadata connection
- A data profiling connection
- Profile sets and executions
- Data Source permissions

Displayed Information



Showing 5 entries (filtered from 7 entries)

Name ↑	Database	Server	Database Engine	Group	Metadata Status	# of Profile Sets
SNOWFLAKE SOURCE	SNOWFLAKE_SA...	teknion.snowflake...	Snowflake	Source System	Completed ●	2
SNOWFLAKE TARGET	WS_RED_DEMO_8...	teknion.snowflake...	Snowflake	Data Warehouse		0
TDS Dev Baseline	Validatar_TDSDW_...	10.0.1.225	SQL Server			0
TDS Dev Updates	Validatar_TDSDW_...	10.0.1.225	SQL Server			0
TDS Snowflake	VALIDATAR_DEMO	teknion.snowflake...				0

Action Buttons

New Data Source	Opens a new data source configuration page
Open	Opens an existing data source configuration page
Archive	Archives a data source
Restore	Restores a deleted data source
Include archived	Shows any archived data sources when checked

Table Headers

Name	The data source name
Database	The primary connection's database name
Server	The primary connection's server name
Database Engine	The primary connection's related database engine
Group	The data source group the primary connection belongs to
Metadata Status	Shows if metadata has been ingested for this data source
# of Profile Sets	Shows the number of profile sets created under this data source

Access

1. A user must have Project Admin rights
2. In the navigation pane, expand Project Settings
3. Click Data Sources

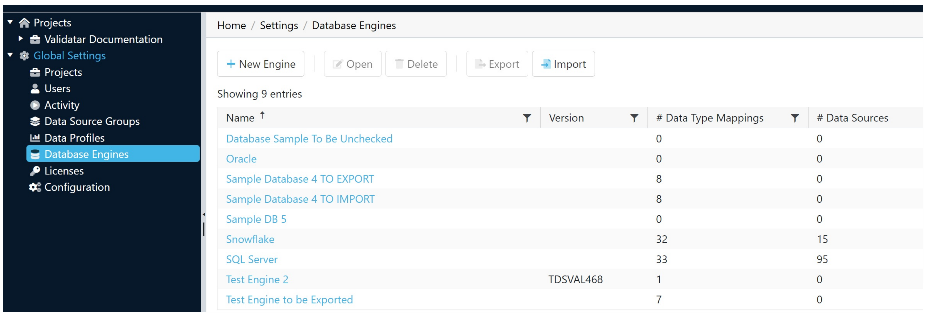
Data Source Templates

New Release Update

Database Engines has been renamed to **Data Source Templates** starting in Validatar 2022.3.

Overview

The **Data Source Templates** page is the place to create and manage all the various data engines that Validatar is connected to. Each data engine is listed on the main page, with the ability to add new engines through the **New Engine** button at the top of the page. The name of each engine is clickable and a detail page opens. The same detail page opens when you click the new engine button. Whether creating a new engine or editing an existing one, the detail page allows you to name or rename (required) and enter a version (optional).

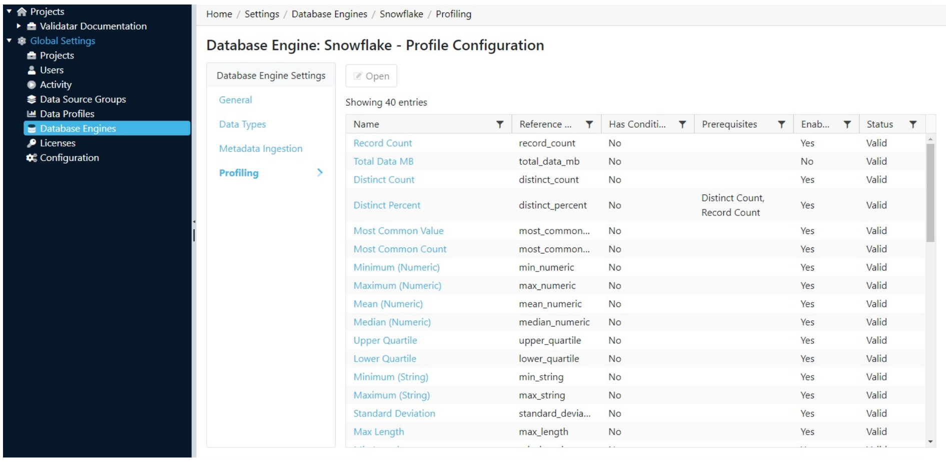


Showing 9 entries

Name	Version	# Data Type Mappings	# Data Sources
Database Sample To Be Unchecked		0	0
Oracle		0	0
Sample Database 4 TO EXPORT		8	0
Sample Database 4 TO IMPORT		8	0
Sample DB 5		0	0
Snowflake		32	15
SQL Server		33	95
Test Engine 2	TDSVAL468	1	0
Test Engine to be Exported		7	0

Data Source Template Settings

Unlike some of the other detail pages within the Global Settings sub-categories, this page has a left-pane list with three sub-pages of data engine settings to create or edit.



Database Engine: Snowflake - Profile Configuration

Showing 40 entries

Name	Reference ...	Has Conditi...	Prerequisites	Enab...	Status
Record Count	record_count	No		Yes	Valid
Total Data MB	total_data_mb	No		No	Valid
Distinct Count	distinct_count	No		Yes	Valid
Distinct Percent	distinct_percent	No	Distinct Count, Record Count	Yes	Valid
Most Common Value	most_common...	No		Yes	Valid
Most Common Count	most_common...	No		Yes	Valid
Minimum (Numeric)	min_numeric	No		Yes	Valid
Maximum (Numeric)	max_numeric	No		Yes	Valid
Mean (Numeric)	mean_numeric	No		Yes	Valid
Median (Numeric)	median_numeric	No		Yes	Valid
Upper Quartile	upper_quartile	No		Yes	Valid
Lower Quartile	lower_quartile	No		Yes	Valid
Minimum (String)	min_string	No		Yes	Valid
Maximum (String)	max_string	No		Yes	Valid
Standard Deviation	standard_devia...	No		Yes	Valid
Max Length	max_length	No		Yes	Valid

The **Data Types** sub-page allows the user to associate the Data Type with Mapped Type. The Mapped Type options are based on a dropdown menu of standard options, such as Boolean, Date/Time, and String. Mappings can be added and deleted on this page by using the appropriate buttons at the top of the page.

The **Metadata Ingestion** sub-page is the place for writing, storing, and managing SQL scripts for data engines. This page contains three separate tabs, Schema Script, Table Script, and Column Script. all of these scripts are required as part of the database engine information.

The third sub-page, **Profiling**, is to set the profile configuration. The Formula Type needs to be set to either Query or Calculation, from the dropdown menu in the Formula Type box. There is also an option to set the Grouping Key and Profile Condition. In addition, the Aggregate Formula and FROM statement (including optional WHERE clause) are defined on this page.

Creating a Data Source

Overview

Creating a data source is one of the first steps you'll take in Validatar as a data source provides the connection to the data you want to learn more about and execute tests against.

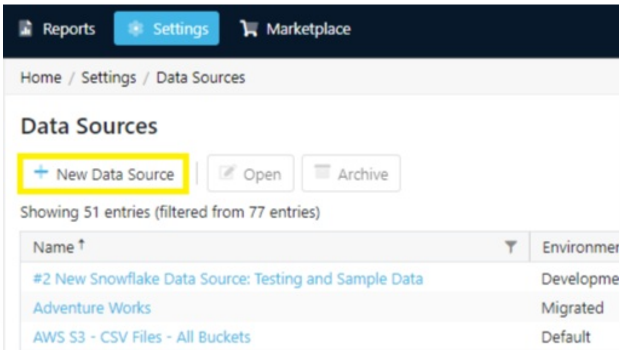
First Steps

Before creating a data source, you'll want to modify your data environments list. A data environment is a way to distinguish similar data sources. Most often, users have Development, QA or Test, and Production environments. To add environments:

1. Navigate to **Settings** > **Configuration** > **Data Environments**.
2. Click **+ Add Environment** and enter each name.
3. Click **Save**.

Creating a New Data Source

After you have your environments added, you can start creating data sources. A user must be a Global or Data Source Admin to create a data source.



Showing 51 entries (filtered from 77 entries)

Name	Environmer
#2 New Snowflake Data Source: Testing and Sample Data	Developme
Adventure Works	Migrated
AWS S3 - CSV Files - All Buckets	Default

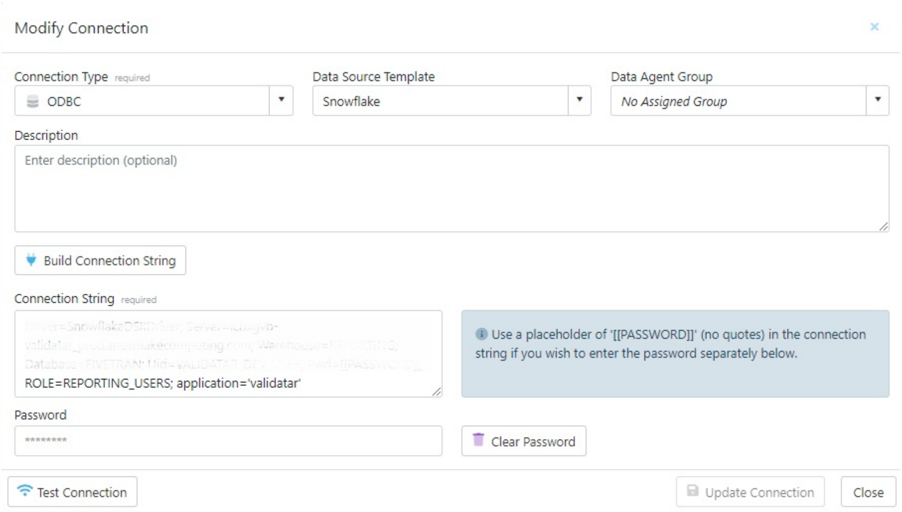
1. Navigate to **Settings** > **Data Sources**.
2. Click **+ New Data Source**.
3. Enter a name for your data source. Use something that's easily identifiable by your entire team.
4. Select the **Environment** from the dropdown.
5. Enter a data source description (optional).

Creating the Connection

You will have to enter a connection string or Python script and parameters to connect to data. This is referred to as your Primary Connection. To set up the connection, click **Edit**.

SQL Based Connections

1. Choose the appropriate connection type.
2. Choose the associated Data Source Template or choose No Assigned Template.
3. Choose the Data Agent Group or choose No Assigned Group.
4. Enter the connection string. Use the Connection String Builder for assistance.
5. Use the secure text box field to enter the Password.
6. Test the connection.



Modify Connection

Connection Type required: ODBC

Data Source Template: Snowflake

Data Agent Group: *No Assigned Group*

Description: Enter description (optional)

Connection String required

[redacted]
ROLE=REPORTING_USERS; application='validatar'

Use a placeholder of '[[PASSWORD]]' (no quotes) in the connection string if you wish to enter the password separately below.

Password: ********

Validatar has **four connection types**: SQL Server, ODBC, OLE DB, and a Python Script. **Note:** Make sure the appropriate drivers are installed on the web server before creating the connection. Also, if you'd rather use a DSN for the ODBC connection, be sure to create the DSN using the ODBC Data Source (64-bit).

Data Source Templates help Validatar pull metadata from the Data Source. You may have to create or import a Data Source Template before using the data source in Validatar.

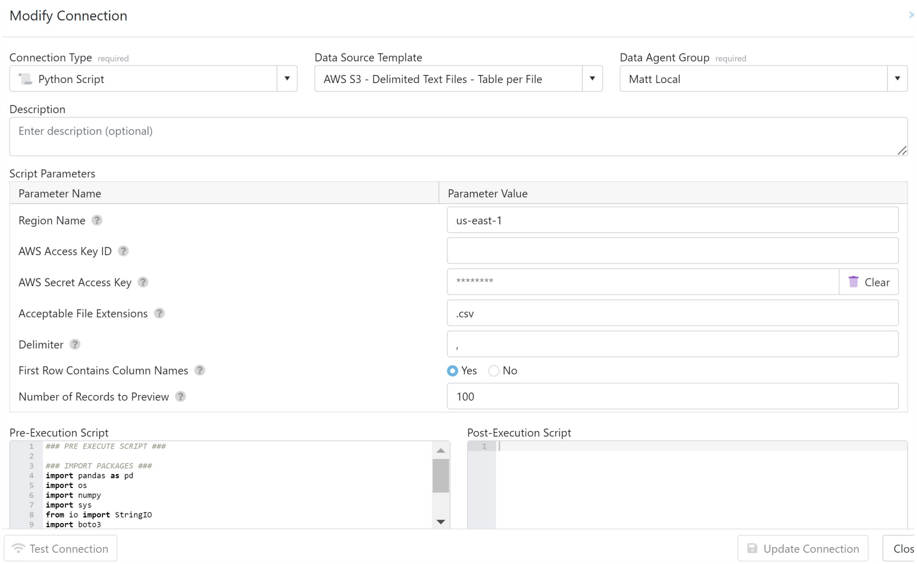
Data Agents allow Validatar to connect to data that is not stored locally and isn't directly accessible. View the articles on Connecting to Remote Data to learn more about Data Agent installation and the prerequisites required before creating a connection to a remote data source. Only select a Data Agent if you need to connect to data remotely.

The **Connection String Builder** uses the selected Connection Type to help you create the connection string. If your connection string requires any additional parameters, exit the Connection String Builder, and add them to the configuration box.

Snowflake Data Sources

Add the following parameter to your Snowflake connection string: **application='validatar'**.

Python Based Connections



Modify Connection

Connection Type required: Python Script

Data Source Template: AWS S3 - Delimited Text Files - Table per File

Data Agent Group required: Matt Local

Description: Enter description (optional)

Script Parameters

Parameter Name	Parameter Value
Region Name	us-east-1
AWS Access Key ID	
AWS Secret Access Key	********
Acceptable File Extensions	.csv
Delimiter	,
First Row Contains Column Names	● Yes ○ No
Number of Records to Preview	100

Pre-Execution Script

```
### PRE EXECUTE SCRIPT ###

### IMPORT PACKAGES ###
import pandas as pd
import os
import numpy
import sys
from io import StringIO
import boto3
```

Post-Execution Script

To connect to a flat file using Python, select **Python Script** as the Connection Type. You can find a list of Python-based Data Source Templates in the Marketplace or create your own in Settings. The example above uses the AWS S3 Data Source Template found in the Marketplace.

Script Parameter names are populated based on the default settings in the associated Data Source Template therefore they can change depending on which template is selected. Complete the respective parameter values based on the data agent fields.

Note: Python execution requires the use of a data agent because you can not directly execute scripts on Validatar Server or Cloud.

Pre- and Post-Execution Scripts are exactly as they sound, script elements that are run before and after a script that is executed against this data source. This applies to tests, metadata ingestion, and custom field ingestion.

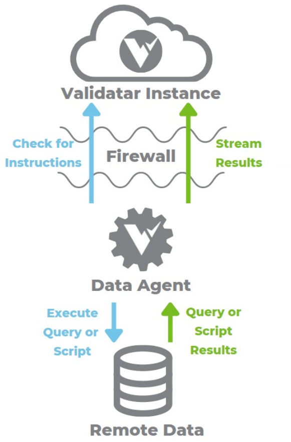
Connecting to Remote Data

Overview

A data agent is required to connect to remote data that is not directly accessible or behind a firewall. This article will walk through the steps of installing, configuring and managing the data agent.

How It Works

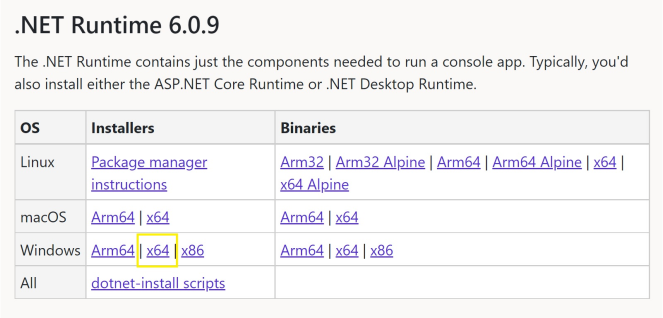
Data Agents can be installed on any computer that can access your data. The agent reaches out to a messaging queue and checks for instructions to stream results back to Validatar. This process helps eliminate networking headaches so you can test more data easily.



Configuring a Data Agent

Prerequisites

Validatar requires **.NET Runtime 6.0x** to be installed. Download .NET 6.0 at https://dotnet.microsoft.com/en-us/download/dotnet/6.0



.NET Runtime 6.0.9

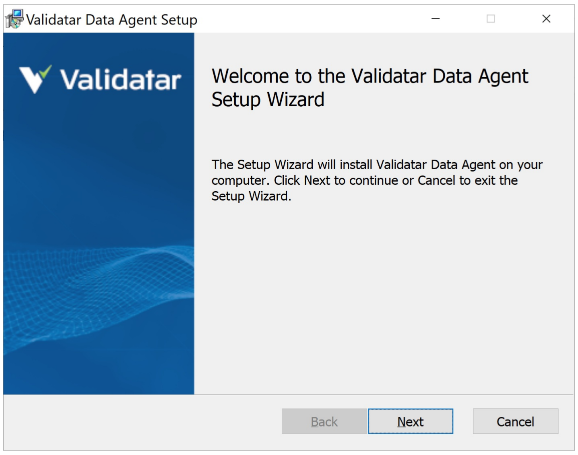
The .NET Runtime contains just the components needed to run a console app. Typically, you'd also install either the ASP.NET Core Runtime or .NET Desktop Runtime.

OS	Installers	Binaries
Linux	Package manager instructions	Arm32 | Arm32 Alpine | Arm64 | Arm64 Alpine | x64 | x64 Alpine
macOS	Arm64 | x64	Arm64 | x64
Windows	Arm64 | x64 | x86	Arm64 | x64 | x86
All	dotnet-install scripts	

Note: This is different than the ASP.NET Core Runtime 6.0x prerequisite needed when Validatar was installed. On the installer download page, scroll past the ASP.NET Core and .NET Desktop sections to find the appropriate .NET Runtime installer.

Data Agent Setup

The Data Agent installer can be found at https://admin.validatar.com/releases under the appropriate release.



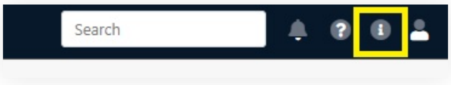
Welcome to the Validatar Data Agent Setup Wizard

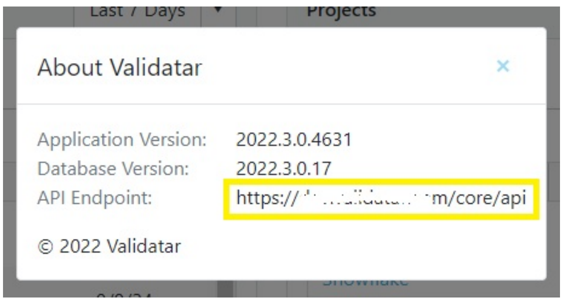
The Setup Wizard will install Validatar Data Agent on your computer. Click Next to continue or Cancel to exit the Setup Wizard.

1. Open the **validatar-data-agent-setup** file.
2. Select the Destination Folder. The C:\ drive is selected by default.
3. Enter a port number to use for the embedded web server. **Port 3282** is the default and should be used unless a different application is listening on port 3282.
4. Click **Install**.
5. Leave **Configure the Data Agent** checked.
6. Click **Finish**.

After the installation is complete, Validatar will open in a browser window with your Data Agent Key.

1. Click Assign to Validatar instance.
2. Enter the Validatar API URL. This can be found by clicking the information icon on the Validatar header bar.





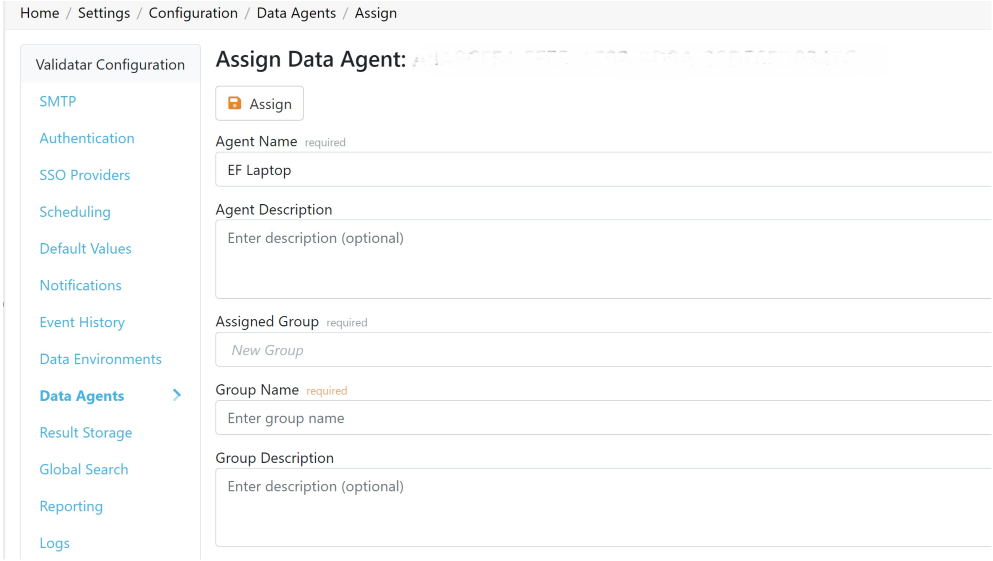
About Validatar

Application Version: 2022.3.0.4631
Database Version: 2022.3.0.17
API Endpoint: https://[redacted]m/core/api

© 2022 Validatar

Managing the Data Agent

After adding the API URL, log into Validatar to finish setting up the Data Agent.



Assign Data Agent: [redacted]

Agent Name required
EF Laptop

Agent Description
Enter description (optional)

Assigned Group required
New Group

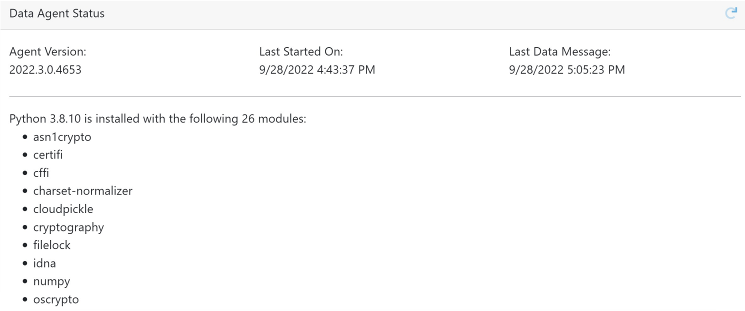
Group Name required
Enter group name

Group Description
Enter description (optional)

After you fill out the required fields click **Assign**.

Data Agent Settings

Navigate to **Settings** > **Configuration** > **Data Agents** to view each Data Agent.



Data Agent Status

Agent Version:	Last Started On:	Last Data Message:
2022.3.0.4653	9/28/2022 4:43:37 PM	9/28/2022 5:05:23 PM

Python 3.8.10 is installed with the following 26 modules:

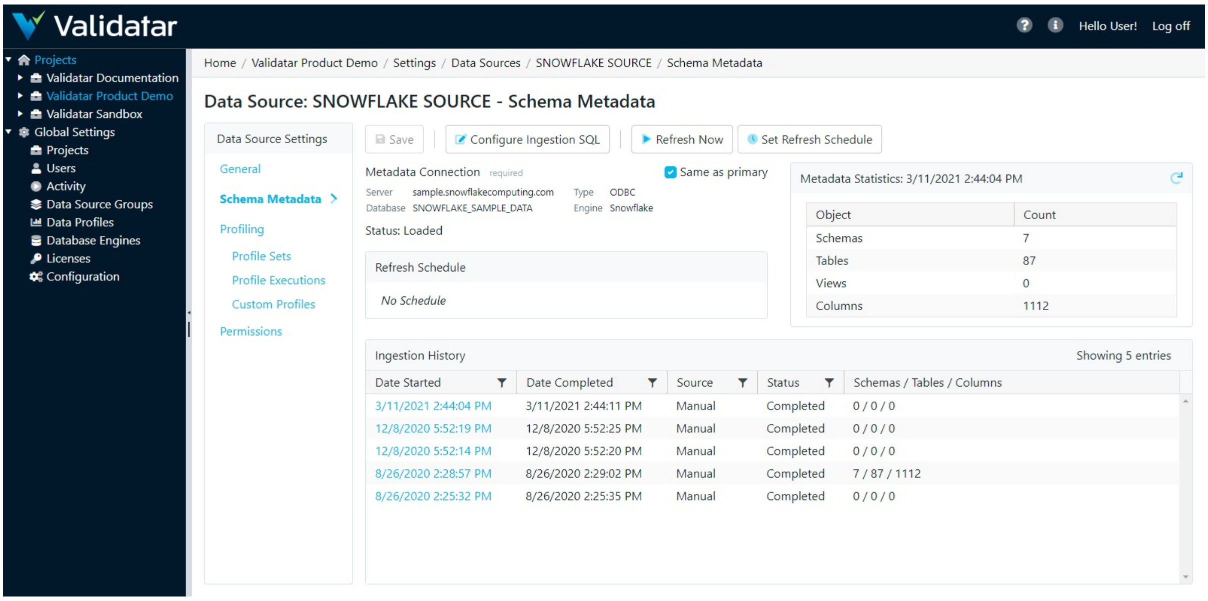
- asn1crypto
- certifi
- cffi
- charset-normalizer
- cloudpickle
- cryptography
- filelock
- idna
- numpy
- oscrypto

You'll be able to pull configuration and exception logs for each agent as well as modify the agent's name, description, and assigned group. You'll also be able to see data agent statistics such as version, last start date, and last data message date. Lastly, you can view which Python modules are installed on the agent as well as install your own.

Ingesting Metadata

Overview

To perform profile testing and ensure complete data testing coverage, Validatar has the ability to store a data source's metadata. We call this metadata ingestion.



Data Source: SNOWFLAKE SOURCE - Schema Metadata

Metadata Connection required ☑ Same as primary

Server sample.snowflakecomputing.com Type ODBC
Database SNOWFLAKE_SAMPLE_DATA Engine Snowflake

Status: Loaded

Refresh Schedule

No Schedule

Metadata Statistics: 3/11/2021 2:44:04 PM

Object	Count
Schemas	7
Tables	87
Views	0
Columns	1112

Ingestion History — Showing 5 entries

Date Started	Date Completed	Source	Status	Schemas / Tables / Columns
3/11/2021 2:44:04 PM	3/11/2021 2:44:11 PM	Manual	Completed	0 / 0 / 0
12/8/2020 5:52:19 PM	12/8/2020 5:52:25 PM	Manual	Completed	0 / 0 / 0
12/8/2020 5:52:14 PM	12/8/2020 5:52:20 PM	Manual	Completed	0 / 0 / 0
8/26/2020 2:28:57 PM	8/26/2020 2:29:02 PM	Manual	Completed	7 / 87 / 1112
8/26/2020 2:25:32 PM	8/26/2020 2:25:35 PM	Manual	Completed	0 / 0 / 0

Here are a few steps to complete in order to ensure your metadata is ingested and used as you would like:

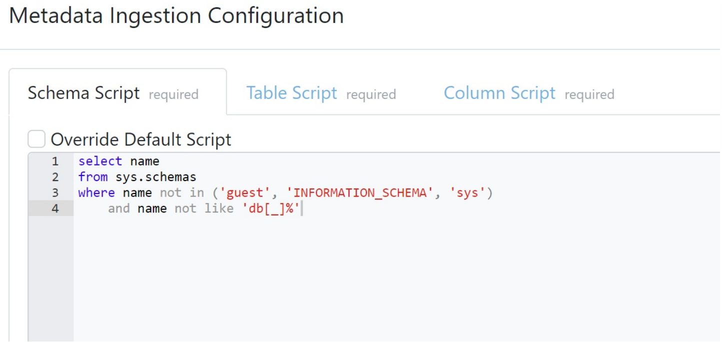
1. Configure the ingestion SQL
2. Set up the metadata connection
3. Refresh the metadata
4. Set a metadata refresh schedule
5. Profile your data source
6. Link each test written in Validatar to its metadata on the test configuration page

Configuring the Ingestion Script

In order to pull metadata from a data source, you must configure the ingestion script. First, choose the connection you want to pull the metadata from. This can be the same as your Primary connection, or you can set up a new connection by unchecking the **Same as primary** checkbox.

SQL Ingestion

Validatar needs at least three different metadata queries for SQL-based data sources: schema, table, and column. These scripts are populated by the associated data source template configuration by default but can be overridden on each data source.



Metadata Ingestion Configuration

Schema Script required | Table Script required | Column Script required

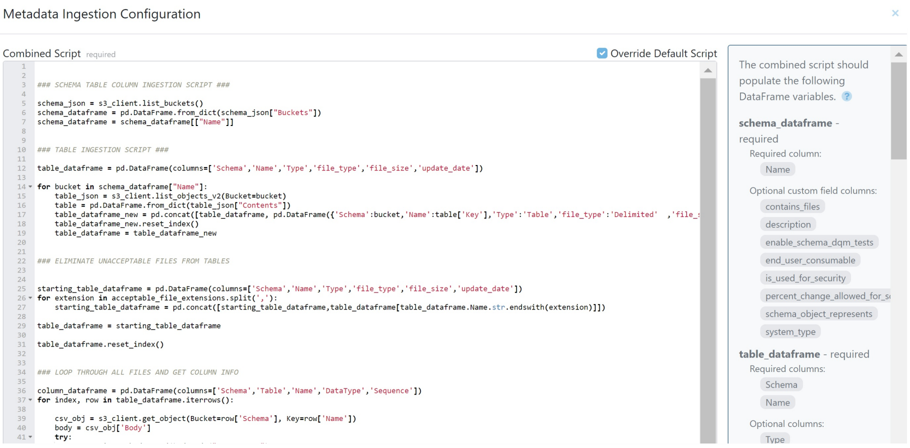
□ Override Default Script

```
select name
from sys.schemas
where name not in ('guest', 'INFORMATION_SCHEMA', 'sys')
    and name not like 'db[_]%'
```

By overriding the ingestion script, you can customize what metadata is returned. This is especially helpful in preventing a large volume of data from being unnecessarily returned. Every time metadata is refreshed, these queries will run separately.

Python Ingestion

On the other hand, Python-based data sources only require one script for metadata ingestion. The script must populate three data frame variables: schema_dataframe, table_dataframe, and column_dataframe. Again, the default combined script is pulled from the associated data source template.



Metadata Ingestion Configuration

Combined Script required

☑ Override Default Script

```


### SCHEMA TABLE COLUMN INGESTION SCRIPT ###

schema_json = s3_client.list_buckets()
schema_dataframe = pd.DataFrame.from_dict(schema_json["Buckets"])
schema_dataframe = schema_dataframe[["Name"]]


### TABLE INGESTION SCRIPT ###

table_dataframe = pd.DataFrame(columns=['Schema','Name','Type','file_type','file_size','update_date'])

for bucket in schema_dataframe["Name"]:
    table_json = s3_client.list_objects_v2(Bucket=bucket)
    table = pd.DataFrame.from_dict(table_json["Contents"])
    table_dataframe_new = pd.concat([table_dataframe, pd.DataFrame({'Schema':bucket,'Name':table['Key'],'Type':'Table','file_type':'Delimited'  ,'file_s
    table_dataframe_new.reset_index()
    table_dataframe = table_dataframe_new


### ELIMINATE UNACCEPTABLE FILES FROM TABLES


starting_table_dataframe = pd.DataFrame(columns=['Schema','Name','Type','file_type','file_size','update_date'])
for extension in acceptable_file_extensions.split(','):
    starting_table_dataframe = pd.concat([starting_table_dataframe,table_dataframe[table_dataframe.Name.str.endswith(extension)]])

table_dataframe = starting_table_dataframe

table_dataframe.reset_index()


### LOOP THROUGH ALL FILES AND GET COLUMN INFO

column_dataframe = pd.DataFrame(columns=['Schema','Table','Name','DataType','Sequence'])
for index, row in table_dataframe.iterrows():

    csv_obj = s3_client.get_object(Bucket=row['Schema'], Key=row['Name'])
    body = csv_obj['Body']
    try:
```

The combined script should populate the following DataFrame variables.

schema_dataframe - required

Required column:

- Name

Optional custom field columns:

- contains_files
- description
- enable_schema_dqm_tests
- end_user_consumable
- is_used_for_security
- percent_change_allowed_for_s
- schema_object_represents
- system_type

table_dataframe - required

Required columns:

- Schema
- Name

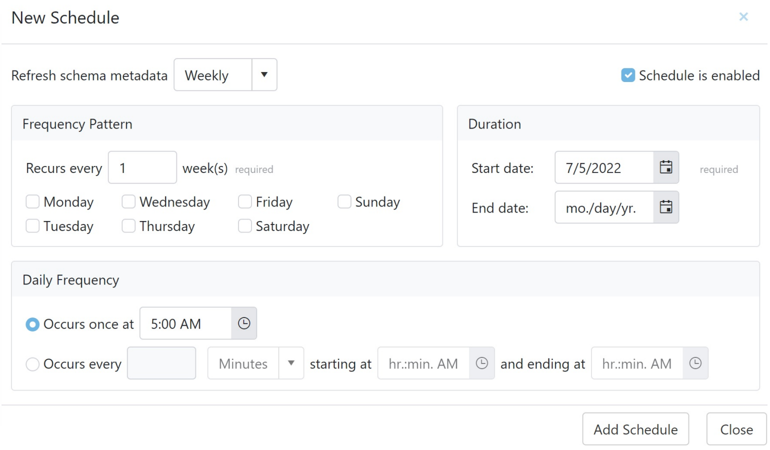
Optional columns:

- Type

Use the guide panel on the right-hand side for all required columns for each data frame.

Refreshing Metadata

A metadata refresh pulls the most current metadata from your data source. You can refresh metadata manually or set up a refresh schedule.



New Schedule

Refresh schema metadata: Weekly

☑ Schedule is enabled

Frequency Pattern

Recurs every 1 week(s) required

□ Monday □ Wednesday □ Friday □ Sunday

□ Tuesday □ Thursday □ Saturday

Duration

Start date: 7/5/2022 required

End date: mo./day/yr.

Daily Frequency

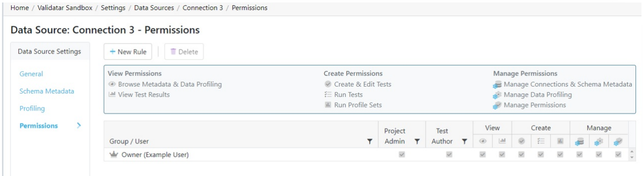
● Occurs once at 5:00 AM

○ Occurs every [ ] Minutes starting at hr.:min. AM and ending at hr.:min. AM

Data Source Permissions

Overview

The Permissions section of the Data Source Configuration page consists of a grid listing each user with access to the Data Source, along with the Data Source permissions that he or she has been granted.



Home / Validatar Sandbox / Settings / Data Sources / Connection 3 / Permissions

Data Source: Connection 3 - Permissions

View Permissions
- Browse Metadata & Data Profiling
- View Test Results

Create Permissions
- Create & Edit Tests
- Run Tests
- Run Profile Sets

Manage Permissions
- Manage Connections & Schema Metadata
- Manage Data Profiling
- Manage Permissions

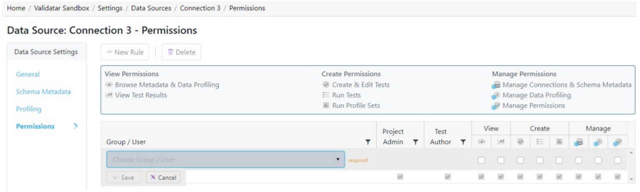
Group / User	Project Admin	Test Author	View		Create			Manage		
Owner (Example User)	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑

Types of Access

When a Data Source is first created, only the user who created it has access to it. This user is automatically given the role of being the Data Source Owner, ensuring that all permissions are always active. A Data Source can only have one owner at a time, though ownership can be changed by double-clicking on the owner's row in the grid and selecting a new name from the dropdown menu.

Permission rules for other users can be set by the owner via the New Rule button on the toolbar, which will add a row to the grid and allow a name and individual permissions to be set and saved.

Aside from the Data Source Owner, there is a row for global and data source admins. Global and data source admins automatically inherit manage permissions for created data sources. Permissions for global and data source admins cannot be changed.



Home / Validatar Sandbox / Settings / Data Sources / Connection 3 / Permissions

Data Source: Connection 3 - Permissions

View Permissions
- Browse Metadata & Data Profiling
- View Test Results

Create Permissions
- Create & Edit Tests
- Run Tests
- Run Profile Sets

Manage Permissions
- Manage Connections & Schema Metadata
- Manage Data Profiling
- Manage Permissions

Group / User	Project Admin	Test Author	View		Create			Manage		
Choose Group / User (required)			☐	☐	☐	☐	☐	☐	☐	☐
	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑

To set individual permissions for a certain user, simply check or uncheck the corresponding boxes on the appropriate row in the grid. Note that this can only be done by users who themselves have the Manage Permissions enabled.

Permission Sets

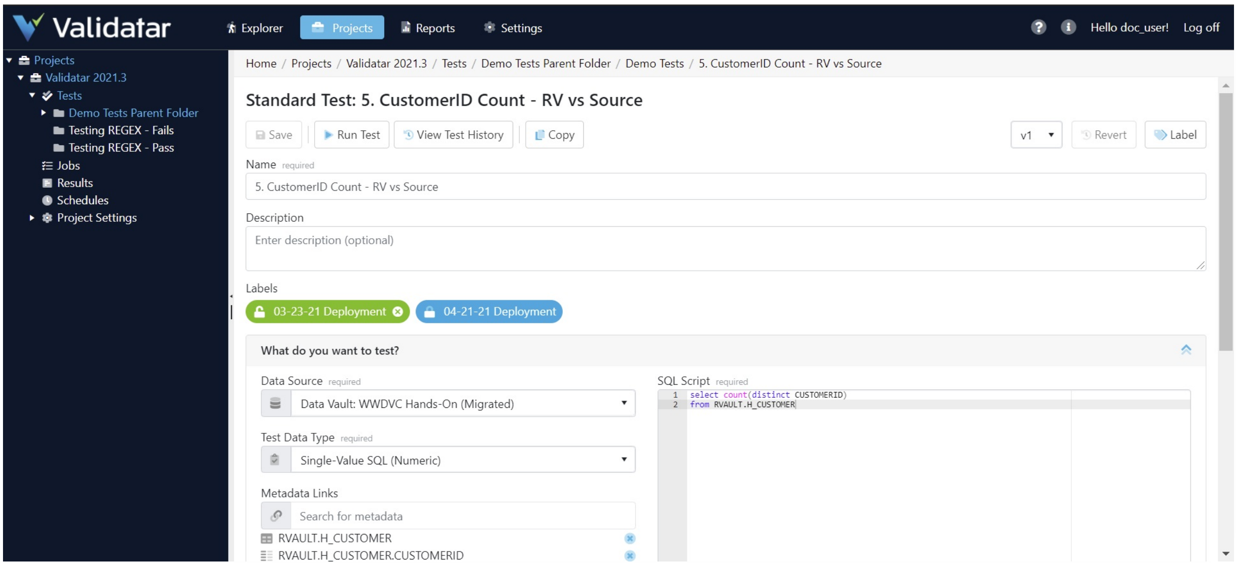
When adding user permissions to data sources, use the pre-defined permission sets to easily assign each permission. The chart below shows which permissions are included in each set.

Permission Set	Browse Metadata & Data Profiling	View Test Results	Create & Edit Tests	Run Tests	Run Profile Sets	Edit Metadata	Manage Permissions (Connections, Schema Metadata, Data Profiles, Permissions)
Viewer	✓	✓					
Tester	✓	✓	✓	✓	✓		
Contributor	✓	✓	✓	✓	✓	✓	
Admin	✓	✓	✓	✓	✓	✓	✓

Configuring a Test

Overview

Configuring a Validatar test is simple. Here's what you need to know:



Standard Test: 5. CustomerID Count - RV vs Source

Name required: 5. CustomerID Count - RV vs Source

Description: Enter description (optional)

Labels: 03-23-21 Deployment, 04-21-21 Deployment

What do you want to test?

Data Source required: Data Vault: WWDVC Hands-On (Migrated)

Test Data Type required: Single-Value SQL (Numeric)

Metadata Links: Search for metadata

- RVAULT.H_CUSTOMER
- RVAULT.H_CUSTOMER.CUSTOMERID

SQL Script required

```
select count(distinct CUSTOMERID)
from RVAULT.H_CUSTOMER
```

There are two types of test creation methods in Validatar:

- Standard Tests
- Template Tests

In this article, you'll learn about creating a standard test that is ideal for ad-hoc, one-time use cases. View the related articles to learn more about Template Tests.

When to Use a Standard Test

Here are a few examples of when you would choose to create a standard test.

- For ad hoc test scenarios
- Business rules only apply to one or a few tables
- All object names are known and/or won't change

Data Sources

A user must have Create & Edit Tests permissions enabled in Data Source Permissions to select a data source. You will only see the list of Data Sources you have access to.

Components of a Test

A test data type is exactly as it sounds: you are choosing the type of data you want to test. The options are:

Single Value SQL (String or Numeric)	Write a SQL statement that returns a single string or numeric value. For example, PL/SQL Copy `select count(*) from dbo.Customers;`

Key-Value List SQL (String or Numeric)	Write a SQL statement that returns two columns. The first column is a key and the second column can either be a string or numeric value. A Key-Value List SQL script usually includes a Group By clause when the second column returned is an aggregate.
	PL/SQL Copy
	`select CustomerID, count(*) from dbo.Customers`
	`group by CustomerID;`

Profile Result	If you've executed a profile set against a table or column, you can use that profile result in a Validatar test. You have to link the appropriate table or column to the test using the Metadata Links section before you can select a profile result. If *Create a link to the metadata that contains profile data* remains after you've linked table or column, go to Profile Set Executions and verify that a profile was run against the appropriate table or column.

Target Data Type

Target Data Type is where you choose the type of data you want to compare the previously selected data to. The options include Single Value SQL, Key-Value List SQL, and Profile Result as well as a few new options:

Numeric Value	Compare your data to a single numeric value without having to write a SQL statement. You would choose this option when you expect a specific number in your data and don't plan to change it.
Numeric Range	See if your data lies between a range of numbers. The range is inclusive.
String Value	Compare your data to a single string value. To search for patterns, check the Regular Expressions checkbox. Learn more about Regular Expressions syntax here.
Previous Profile Result	Compare your data to a previous profile result. You have to link the appropriate table or column to the test using the Metadata Links section before you can select a previous profile result. You can compare to either the average, minimum, or maximum of previous results for a selective amount of days, weeks, or executions. If Create a link to the metadata that contains profile data remains after you've linked table or column, go to Profile Set Executions and verify that a profile was run against the appropriate table or column.

The available options in the Target Data Type field depends on what is selected in the Test Data Type above.

Test Data Type	Available Target Data Types
Single Value SQL (string)	Single Value SQL (string), Profile Result, Previous Profile Result, String Value
Single Value SQL (numeric)	Single Value SQL (numeric), Profile Result, Previous Profile Result, Numeric Value, Numeric Range
Key-Value List SQL (string)	Key-Value List SQL (string), Single Value SQL (string), String Value
Key-Value List SQL (numeric)	Key-Value List SQL (numeric), Single Value SQL (numeric), Numeric Value, Numeric Range
Profile Result	Profile Result, Previous Profile Result, Numeric Value, Numeric Range, Single Value SQL (numeric)

Test Data Type Notes

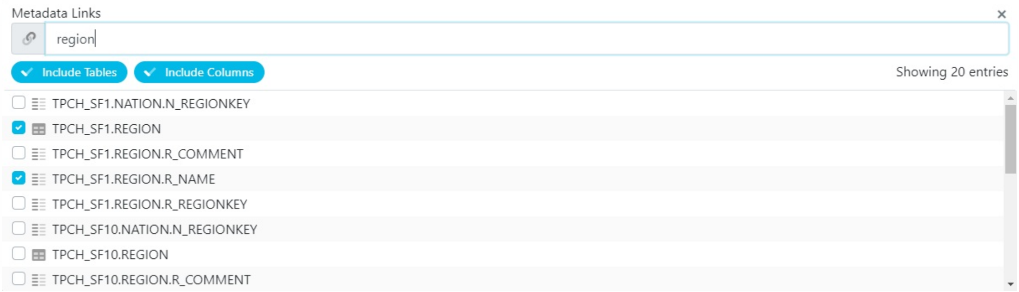
Starting in Validatar 2021.3, the **Multi-Value SQL** test data type is now referred to as **Key-Value List SQL**.

Metadata Links

Metadata Links allow profile results that have been run against a table or column to be used in a Validatar test. Linking metadata is not required but is *highly recommended* to

- Use data profile results in test cases
- Create a relationship between metadata and their related test cases
- Discover how comprehensive testing is

Start typing the name of a table or column to search for the object and select it. You can add multiple metadata links to a test.



Metadata Links

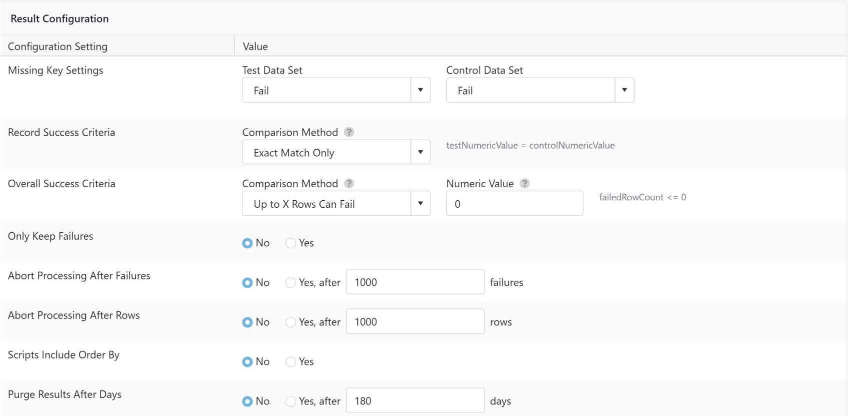
region

✓ Include Tables ✓ Include Columns — Showing 20 entries

- □ TPCH_SF1.NATION.N_REGIONKEY
- ☑ TPCH_SF1.REGION
- □ TPCH_SF1.REGION.R_COMMENT
- ☑ TPCH_SF1.REGION.R_NAME
- □ TPCH_SF1.REGION.R_REGIONKEY
- □ TPCH_SF10.NATION.N_REGIONKEY
- □ TPCH_SF10.REGION
- □ TPCH_SF10.REGION.R_COMMENT

Result Configuration

The Result Configuration section is used to tell Validatar what it should consider as a failed test. You can also select options to increase execution efficiency when running the test. The contents of the result configuration section depend on the combination of test and target data types selected in the previous sections.



Result Configuration

Configuration Setting	Value
Missing Key Settings	Test Data Set: Fail Control Data Set: Fail
Record Success Criteria	Comparison Method: Exact Match Only testNumericValue = controlNumericValue
Overall Success Criteria	Comparison Method: Up to X Rows Can Fail Numeric Value: 0 failedRowCount <= 0
Only Keep Failures	● No ○ Yes
Abort Processing After Failures	● No ○ Yes, after 1000 failures
Abort Processing After Rows	● No ○ Yes, after 1000 rows
Scripts Include Order By	● No ○ Yes
Purge Results After Days	● No ○ Yes, after 180 days

Configuration Setting	Description
Missing Key Settings	One benefit of using Validatar is to identify missing values in a data set. When that scenario happens, you're able to tell Validatar how that missing value should be evaluated. The comparison can either - Pass: You're okay with the missing value - Fail: You want to identify a missing value - Be overridden: You want to override the missing value with another value, then make the comparison.
Value Success Criteria	**For string tests**: Choose if the test should pass or fail based on how the source value compares to the target value. When a regular expression is used, the only comparison option is for the source value to match the target value.
Result/Record Success Criteria	**For numeric tests**: Compares the two pieces of data from the test together and calculate the difference. There are various comparison methods - Exact Match Only: The two numeric values must be exactly the same. - Absolute Numeric Difference: Takes the absolute value of the difference. ABS(Test Value - Control Value) - Percent of Test Value: Takes the absolute value of the percentage of the test value. ABS(Test Value - Control Value *100)/Test Value - Percent of Control Value: Takes the absolute value of the percentage of the control value. ABS(Test Value - Control Value *100)/Control Value - Custom Calculation: This allows you to use test variables (test value, control value, and key-value) to write arithmetic formulas. Use logic commands like TRUE, FALSE, AND, and IF and operators like Greater Than and NOT Equal To. For more complex comparisons, use some of the embedded functions like LOG(), ROUND(), and SQUARE(). You can also set a numeric or percent tolerance value depending on the selected comparison method.
Overall Success Criteria	**For Key-Value List SQL tests:** Since Key-Value List SQL results return multiple records that have been compared, the overall success criteria option allows the user to decide how many failed records account for a failed test. There are multiple comparison methods - All Rows Must Pass: All rows must pass for the entire test to pass. - Up to X Rows Can Fail: There's a maximum number of tests that can fail before the entire test fails. - Up to X Percent of Rows Can Fail: There's a maximum percentage of total records that can fail before the entire test fails. - Custom Calculation: This allows you to use test variables (passed row count, failed row count, total row count) to write arithmetic formulas. Use logic commands like TRUE, FALSE, AND, and IF and operators like Greater Than and NOT Equal To. For more complex comparisons, use some of the embedded functions like LOG(), ROUND(), and SQUARE().
Only Keep Failures	Choose Yes if you only want to return failed records. This keeps your results repository from being overloaded.
Abort Processing After Failures	Allows Validatar to abort the execution of a test after a certain number of record failures.
Abort Processing After Rows	Allows Validatar to abort the execution of a test after a certain number of rows have been returned.
Scripts Include Order By	Choose Yes if your SQL script includes an order by clause. This lets Validatar know that it does not have to sort your data, optimizing execution time.
Purge Results After Days	Allows Validatar to clear the results of this test from the Validatar database after a certain number of days. Configuring this option overrides the default purge settings set at the global level.

Access

1. Click on Tests in the navigation pane or from the project home page.
2. Select the New button.

Configuring a Template Test

Overview

Configuring a template test is made up of six easy steps:

1. Select the target object type and metadata object selection and definition.
2. Build out the test.
3. Define how each child test is to be named and in which folder they should be saved.
4. Configure the metadata object selection.
5. Materialize the child tests.
6. Run the test.

To configure a template test, you must first select a target object type. The target object type defines how the Template Test will generate each child test. A child test is a standard test that was created, or materialized, from a template instead of by the typical new test creation process.

Target Object Type	Definition
Table	For table-level comparisons. This test generates one child test per table. For example, each child test compares the record counts of two tables to each other.
Column	For column-level comparisons. This test generates one child test per column. For example, each child test compares the max value of two columns to each other.

The second option you'll have to configure is how you want Validatar to select the metadata objects that will populate the template when it's time to materialize the child tests.

Object Selection Options	Definition
Metadata Driven	This option uses the metadata that Validatar has ingested from the data source and allows the user to exclude any unwanted objects.
Metadata Query	This option allows a user to write a SQL query that will pull the intended metadata from the data source.
Import File	This option allows a flat file that has each object and its intended test definition settings to be imported into the template. Used to map source and target metadata objects.

Learn more about each of the object selection options in later sections.

Test Definition

The Test Definition tab is similar to the standard test configuration page. Test Variables and References are sections that have been added and they help to make template tests dynamic.

Test Variables

Test Variables are custom references you can make to your data source. Each variable requires a reference key, variable type, and default value.

Reference Key	The name Validatar will use to refer to the object.
Variable Type	The type of object you're referencing. Options are: • Schema • Table • Column • Numeric • Boolean • Other
Default Value	The default value Validatar will substitute for the variable. This is optional.

References

Use reference tags for metadata objects and profile objects to dynamically plug desired objects in throughout your template without having to hard code anything!

How to Use References

Anywhere you want to place a metadata object or data profile in your template, you can use a reference tag. The reference tag acts as a placeholder that detects and swaps out the reference for the intended object name. For example, if you want Validatar to grab a fully qualified table name, you would type:

PL/SQL Copy

```
{{meta.database_name}}.{{meta.schema_name}}.{{meta.table_name}}
```

After materialization, the standard tests may replace the reference tags with something like:

PL/SQL Copy

```
CRM.dbo.Customer
```

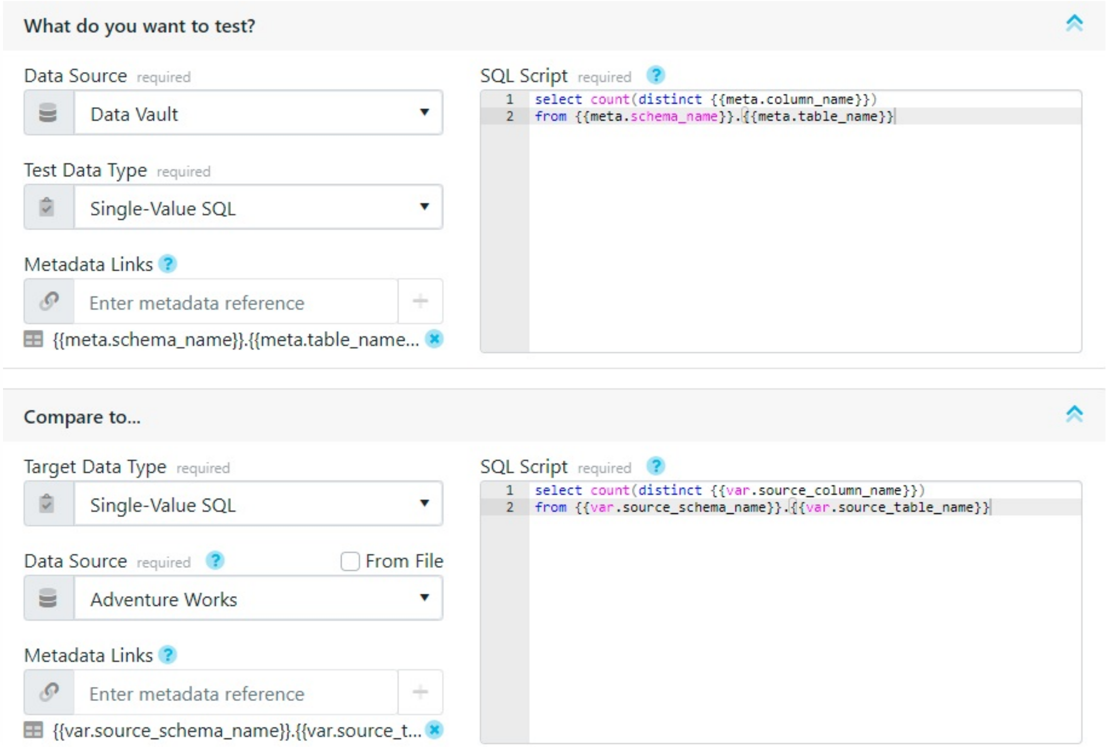
To use references, enclose the metadata object or data profile in double curly braces {{ }}. We recommend copying and pasting the reference tags into the view to ensure that the proper format is used.

Reference Note

Reference tags are case-sensitive. Verify that each reference tag uses all lowercase characters before saving.

View how metadata references are used in the SQL queries, Metadata Links, and Child Test Name field in the picture below.

Test Definitions Tab



What do you want to test?

Data Source required: Data Vault

Test Data Type required: Single-Value SQL

Metadata Links: Enter metadata reference

{{meta.schema_name}}.{{meta.table_name...

SQL Script required

```
select count(distinct {{meta.column_name}})
from {{meta.schema_name}}.{{meta.table_name}}
```

Compare to...

Target Data Type required: Single-Value SQL

Data Source required: Adventure Works ☐ From File

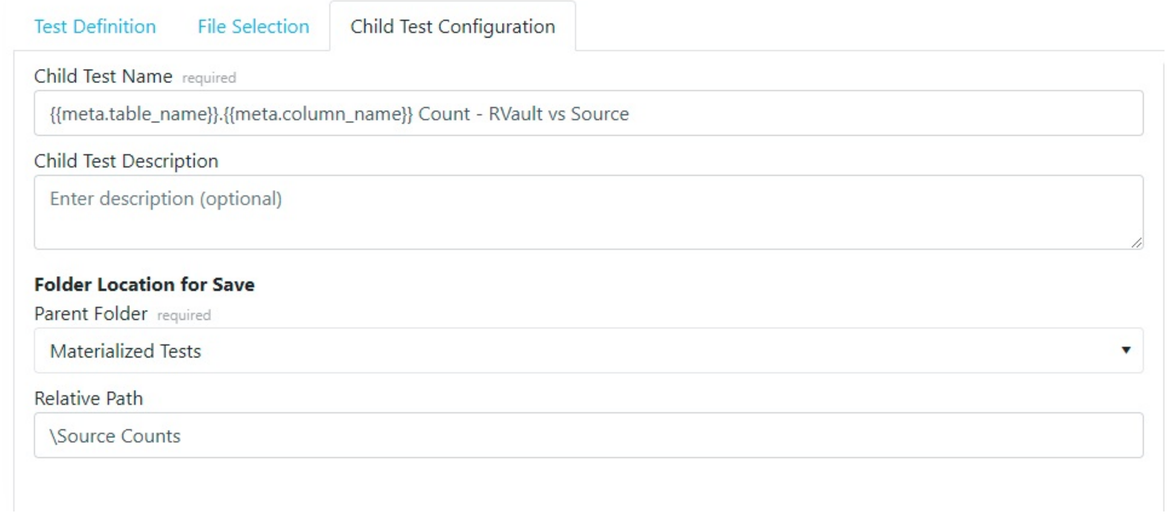
Metadata Links: Enter metadata reference

{{var.source_schema_name}}.{{var.source_t...

SQL Script required

```
select count(distinct {{var.source_column_name}})
from {{var.source_schema_name}}.{{var.source_table_name}}
```

Child Test Configuration Tab



Child Test Name required
{{meta.table_name}}.{{meta.column_name}} Count - RVault vs Source

Child Test Description
Enter description (optional)

Folder Location for Save

Parent Folder required
Materialized Tests

Relative Path
\Source Counts

Results Configuration

The Results configuration section is quite similar to the result evaluation section in a standard test. The evaluation options configured in the template will apply to all child tests when they are materialized but can be changed later.

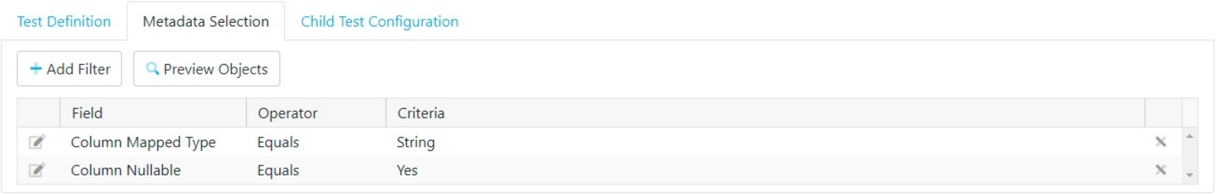
The available result configuration options depend on the Test Data Types that were selected in the sections above.

Metadata Selection

Using filters

By default, if the Metadata Driven object selection is selected, all objects in the data source will be used to populate the child tests. For a more focused set of objects, you can add filters to the metadata selections.

You can also use Custom Fields to filter the metadata objects. The custom fields are filtered in the list by object type.



Field	Operator	Criteria
Column Mapped Type	Equals	String
Column Nullable	Equals	Yes

Filter Options

Field	Operator	Criteria
Schema Name	String Filters	Any string input
Table Name	String Filters	Any string input
Table Record Count	Numeric Filters	Any numeric input
Table Type	Equals, Not Equals	Table or View
Column Name	String Filters	Any string input
Column Data Type	String Filters	Any string input
Column Mapped Type	Equals, Not Equals	Validatar data types
Column Nullable	Equals, Not Equals	Yes or No
Column Identity	Equals, Not Equals	Yes or No
Column Primary Key	Equals, Not Equals	Yes or No
Column Max Length	Numeric Filters	Any numeric input

Metadata Query

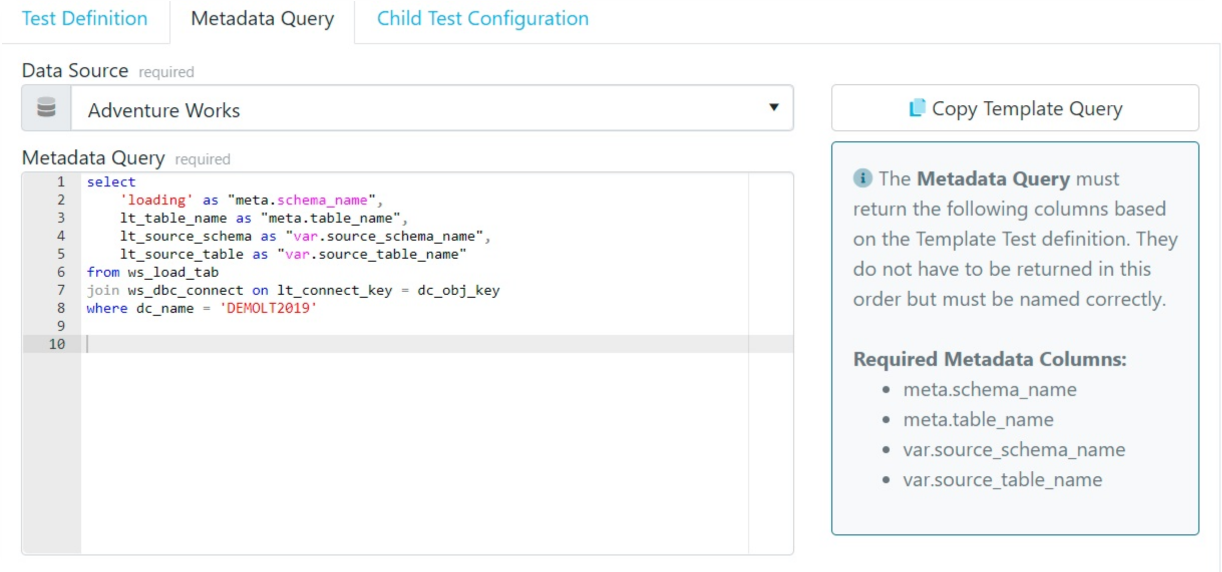
Using a metadata query is a second way to decide which objects will populate the template. This option allows you to pull metadata directly from the data source using a SQL script with the proper table joins or filter, if necessary. After you select your data sources on the Test Definition tab, you can select which data source you want to pull metadata from on the Metadata Query tab.

The Metadata Query must return certain columns based on what references were used on the Test Definition tab. You will see which columns are required under **Required Metadata Columns**. They do not have to be returned in the order they are listed but they must be named correctly.

To ensure the proper column aliases are used, copy a template of the accepted SQL query by clicking the **Copy Template Query** button on the right-hand side of the screen.

To ensure that the correct metadata is included click the **Preview and Validate Metadata Query**.

View what a metadata query object selection may look like in the image below.



Data Source required

Adventure Works

Metadata Query required

```
select
    'loading' as "meta.schema_name",
    lt_table_name as "meta.table_name",
    lt_source_schema as "var.source_schema_name",
    lt_source_table as "var.source_table_name"
from ws_load_tab
join ws_dbc_connect on lt_connect_key = dc_obj_key
where dc_name = 'DEMOLT2019'

```

The **Metadata Query** must return the following columns based on the Template Test definition. They do not have to be returned in this order but must be named correctly.

Required Metadata Columns:

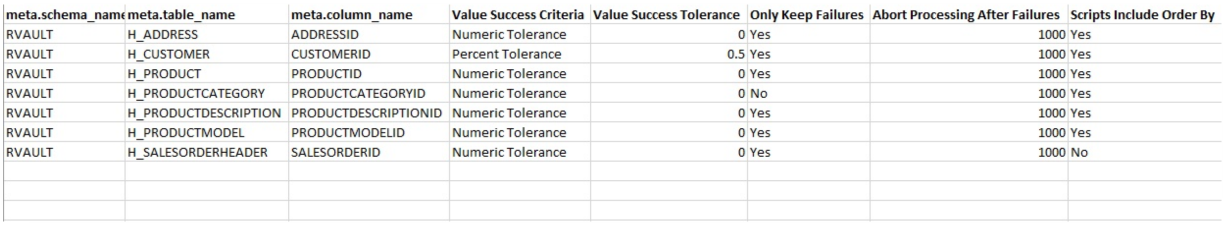
- meta.schema_name
- meta.table_name
- var.source_schema_name
- var.source_table_name

File Selection

The third way to choose which objects will populate the template is via importing a flat file.

The table contains a record for each object in the database that is to be used to replace the reference tags in the template. Each record represents one child test that will be created during materialization, therefore each record also includes results evaluation criteria that should be used for each test.

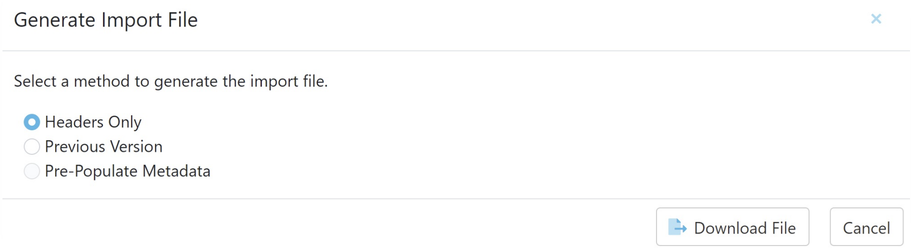
Below is a sample import file for a column-level template test.



meta.schema_name	meta.table_name	meta.column_name	Value Success Criteria	Value Success Tolerance	Only Keep Failures	Abort Processing After Failures	Scripts Include Order By
RVAULT	H_ADDRESS	ADDRESSID	Numeric Tolerance	0	Yes	1000	Yes
RVAULT	H_CUSTOMER	CUSTOMERID	Percent Tolerance	0.5	Yes	1000	Yes
RVAULT	H_PRODUCT	PRODUCTID	Numeric Tolerance	0	Yes	1000	Yes
RVAULT	H_PRODUCTCATEGORY	PRODUCTCATEGORYID	Numeric Tolerance	0	No	1000	Yes
RVAULT	H_PRODUCTDESCRIPTION	PRODUCTDESCRIPTIONID	Numeric Tolerance	0	Yes	1000	Yes
RVAULT	H_PRODUCTMODEL	PRODUCTMODELID	Numeric Tolerance	0	Yes	1000	Yes
RVAULT	H_SALESORDERHEADER	SALESORDERID	Numeric Tolerance	0	Yes	1000	No

Generating an Import File

To get the proper column headers in the import file the file should be generated from Validatar. The column headers depend on which reference tags are used in either the SQL script text box or Metadata Links box, or any field that has *Import From File* checked on the Test Definition tab. The file also includes any references used on the Child Test Configuration tab.



Generate Import File ×

Select a method to generate the import file.

● Headers Only
○ Previous Version
○ Pre-Populate Metadata

Download File | Cancel

There a 3 options when choosing to generate the import file:

Headers Only	This option downloads a blank Excel file with column headers only.
Previous Version	This option lists the previous files that were imported into the Template test. This file includes column headers as well as metadata and results evaluation content from the previous file.
Pre-Populate Metadata	Coming soon.

Uploading an Import File

Once the import file has been configured to include the appropriate metadata objects and test settings, it's time to upload the file. Just select the **Upload Import File** button on the File Selection tab to do so.

If column headers are references that aren't being used anywhere on the Test Definition or Child Test Configuration tabs

View Version History

Another option for templates that have been saved with different versions of the import file is viewing the version history. Learn who has previously imported a file and when it was uploaded.

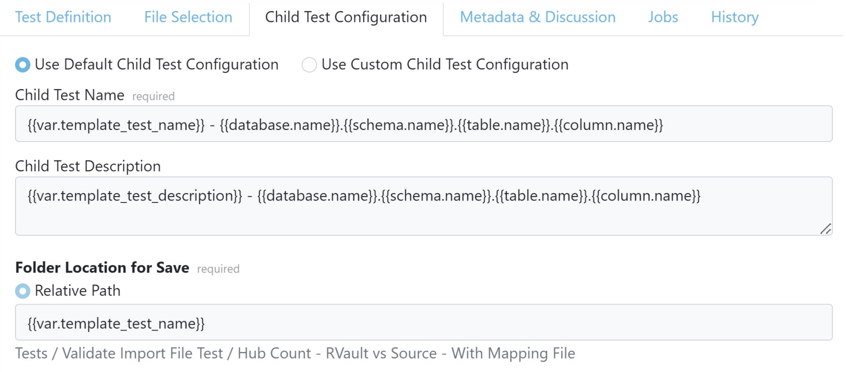
Child Test Configuration

Child tests are simply standard tests that have been created from a template. The Child Test Configuration tab lets Validatar know how you want to name each child test and where should they be saved in the test repository.

For naming and saving child tests, you can:

1. Use a default child test configuration
2. Or use a custom child test configuration.

Default Child Test Configuration



Test Definition | File Selection | Child Test Configuration | Metadata & Discussion | Jobs | History

● Use Default Child Test Configuration ○ Use Custom Child Test Configuration

Child Test Name required
{{var.template_test_name}} - {{database.name}}.{{schema.name}}.{{table.name}}.{{column.name}}

Child Test Description
{{var.template_test_description}} - {{database.name}}.{{schema.name}}.{{table.name}}.{{column.name}}

Folder Location for Save required
● Relative Path
{{var.template_test_name}}
Tests / Validate Import File Test / Hub Count - RVault vs Source - With Mapping File

By default, child tests are named using the template test name and then tagging the related metadata object. For example, Record Count Comparision - EDW.dbo.Customer. The description will use the template test's description and add the related metadata object.

The **Relative Path** used to save the child tests will be located in a folder using the same name as the Template Test.

Custom Child Test Configuration

With a custom configuration, you can choose any format and location to save the child tests. Similar to the Test Definition tab, you can use metadata reference tags in the Name and Description fields. Just type out the reference you want to use and when the child tests are materialized, the appropriate object will be plugged in its place.

Folder Location for Save

> **Parent Folder 2022.2 Update**
>
> Starting in 2022.2, we've removed **Parent Folder** from the test configuration.

Relative Path

The relative path starts at the location where the Template Test is currently located and can expand from there. Create new folders in the path by listing out the folder names or using reference tags delimited by a forward slash (/).

If the relative path is left blank, the child tests will be put in the same folder the Template Test is saved in.

Absolute Path

The absolute path expects the **Tests** folder to be the starting point of the path. You can use the names of existing folders or create new folders using the absolute path option.

Executing Child Tests

Overview

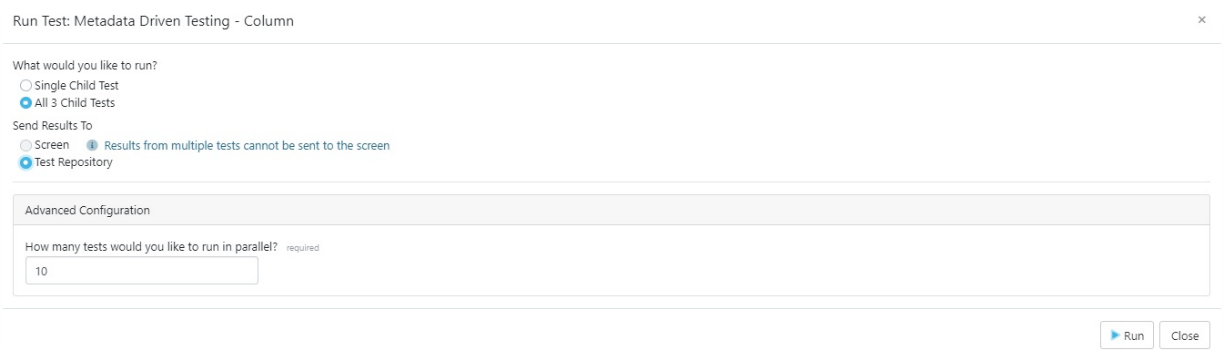
After materialization, you can think of child tests as a standard Validatar test that was created from a template. Because of this, child tests can be run from the template test configuration screen, the test repository after materialization, or as part of a job just like a standard test.

From the Template Test Screen

1. When running child tests from the template test screen there are a few configuration options to choose from. You can either choose to run a single child test, which is recommended to validate that your test was created and runs as expected, or you can run all child tests at once. If you choose to run a single child test you have the option to choose which test runs.
2. Next, choose where to send the test results. Similar to a standard test, results can be sent to the screen for one-time viewership or be saved in the test repository. Note that if you choose to run all child tests instead of a single child test, you can only send results to the test repository.

Advanced Configuration

There is also an option to configure how many tests will run in parallel. This option allows multiple tests to run simultaneously which helps with optimization. The default setting is 10.



Run Test: Metadata Driven Testing - Column

What would you like to run?
○ Single Child Test
● All 3 Child Tests

Send Results To
○ Screen Results from multiple tests cannot be sent to the screen
● Test Repository

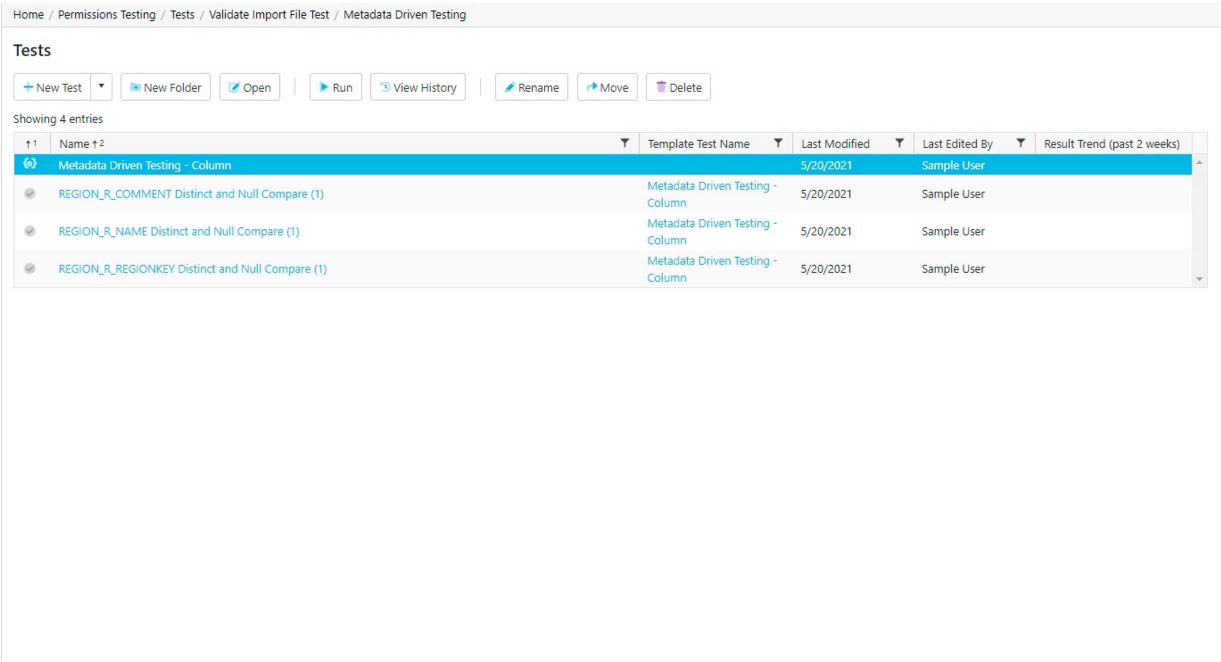
Advanced Configuration

How many tests would you like to run in parallel? required
10

If the template has been modified since the last time child tests were materialized, you will be prompted to materialize child tests before you run the tests.

From the Test Repository

Template tests can also be run from the test repository screen by selecting the template test record and clicking the Run button on the toolbar. This process will display the same Run Test pop-up configuration window that appears on the template test screen.


Home / Permissions Testing / Tests / Validate Import File Test / Metadata Driven Testing
Tests
New Test
New Folder
Open
Run
View History
Rename
Move
Delete
Showing 4 entries

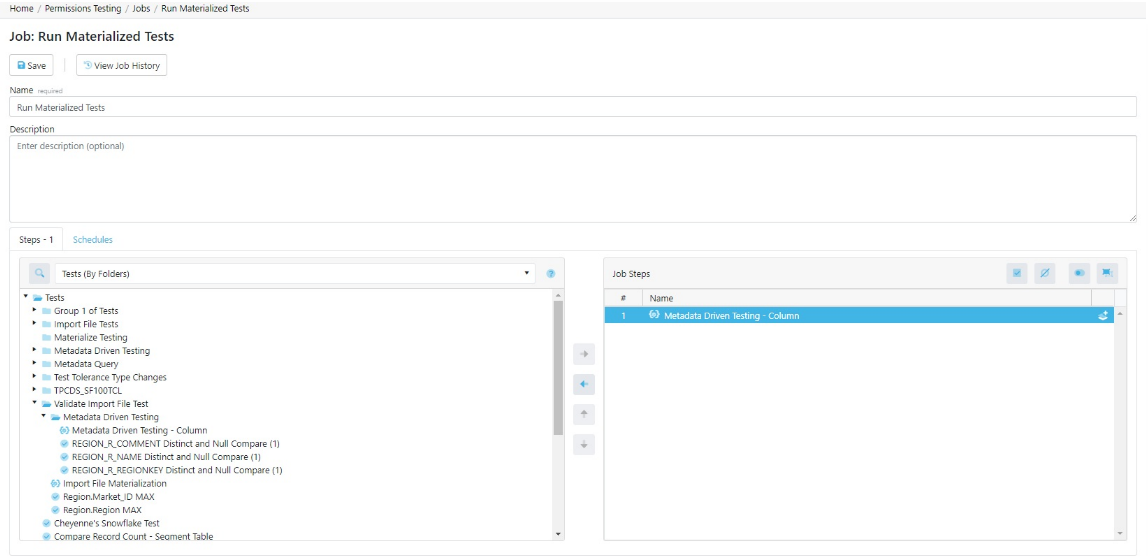
↑1	Name ↑2	Template Test Name	Last Modified	Last Edited By	Result Trend (past 2 weeks)
{x}	Metadata Driven Testing - Column		5/20/2021	Sample User	
✓	REGION_R_COMMENT Distinct and Null Compare (1)	Metadata Driven Testing - Column	5/20/2021	Sample User	
✓	REGION_R_NAME Distinct and Null Compare (1)	Metadata Driven Testing - Column	5/20/2021	Sample User	
✓	REGION_R_REGIONKEY Distinct and Null Compare (1)	Metadata Driven Testing - Column	5/20/2021	Sample User	

Not only can you run all child tests from the template itself, but you can also run each test as a standard test. Select the appropriate test standard test, which is identified by the checkmark icon next to it, and either select the Run button on the toolbar or drill into the test interface and run the test from there.

Within a Job

Another option to execute child tests is to add the template test or multiple child tests to a job. Adding the template test itself ensures that no child test is accidentally excluded from the job resulting in total test coverage for your metadata objects.

1. Just select the template test on the left-hand side and move it over to the Job Steps on the right.


Home / Permissions Testing / Jobs / Run Materialized Tests
Job: Run Materialized Tests
Save
View Job History
Steps - 1
Schedules
Tests (By Folders)
Job Steps
#
Name

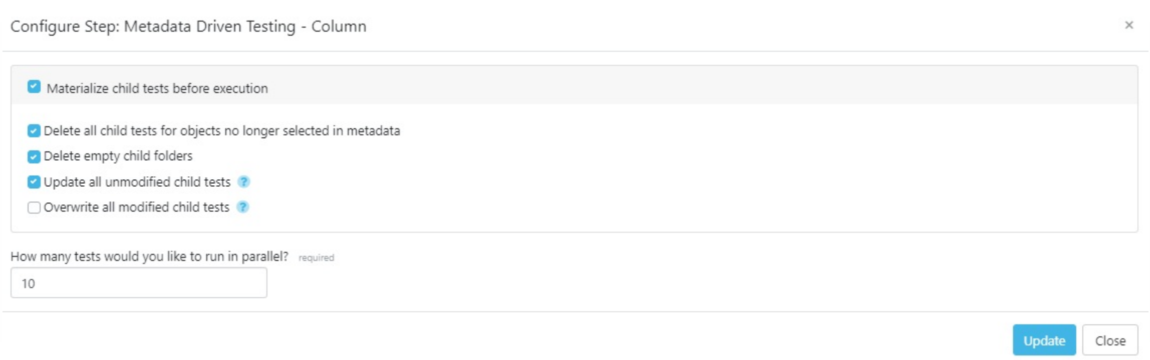
Name required: Run Materialized Tests

Description: Enter description (optional)

- Tests
  - Group 1 of Tests
  - Import File Tests
  - Materialize Testing
  - Metadata Driven Testing
  - Metadata Query
  - Test Tolerance Type Changes
  - TPCDS_SF100TCL
  - Validate Import File Test
    - Metadata Driven Testing
      - Metadata Driven Testing - Column
      - REGION_R_COMMENT Distinct and Null Compare (1)
      - REGION_R_NAME Distinct and Null Compare (1)
      - REGION_R_REGIONKEY Distinct and Null Compare (1)
    - Import File Materialization
    - Region.Market_ID MAX
    - Region.Region MAX
  - Cheyenne's Snowflake Test
  - Compare Record Count - Segment Table

#	Name
1	Metadata Driven Testing - Column

2. Once the template is added to the job step list, select the materialize icon to the right of the template name to choose if you want to materialize the child tests before execution. If you do, you will be given a list of materialization settings to choose from.



Configure Step: Metadata Driven Testing - Column

☑ Materialize child tests before execution

☑ Delete all child tests for objects no longer selected in metadata
☑ Delete empty child folders
☑ Update all unmodified child tests
□ Overwrite all modified child tests

How many tests would you like to run in parallel? required
10

3. Update the configuration, save the job, then navigate back to the Job Repository screen to run the job or add a schedule to the job for it to run automatically.

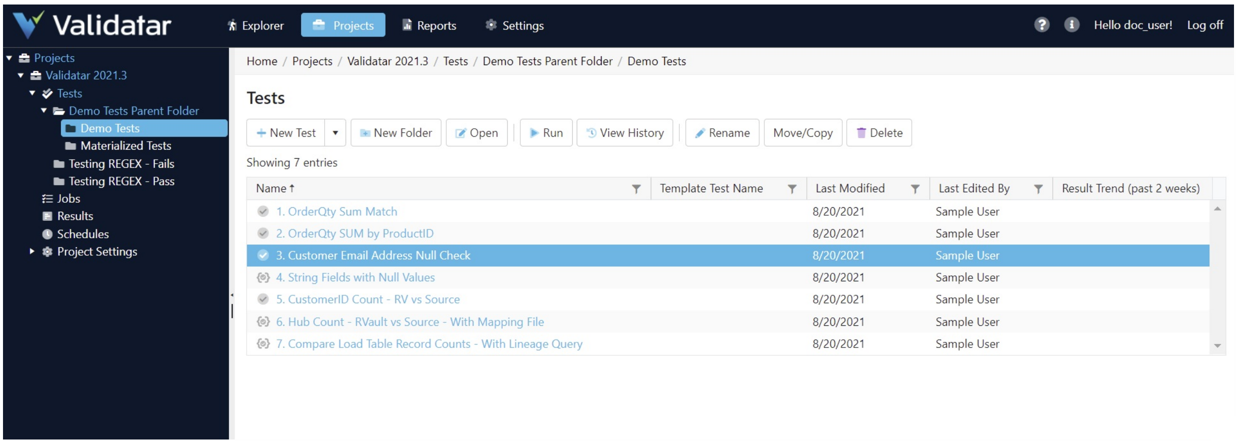
Test Repository

Overview

The **Test Repository** page provides access to all tests and access to various actions described below.

Each **Project** has its own Test repository. Tests in the test repository are organized in an easy-to-use folder structure that allows teams to organize their tests in whatever structure best fits their needs.

Displayed Information



Name ↑	Template Test Name	Last Modified	Last Edited By	Result Trend (past 2 weeks)
1. OrderQty Sum Match		8/20/2021	Sample User	
2. OrderQty SUM by ProductID		8/20/2021	Sample User	
3. Customer Email Address Null Check		8/20/2021	Sample User	
4. String Fields with Null Values		8/20/2021	Sample User	
5. CustomerID Count - RV vs Source		8/20/2021	Sample User	
6. Hub Count - RVault vs Source - With Mapping File		8/20/2021	Sample User	
7. Compare Load Table Record Counts - With Lineage Query		8/20/2021	Sample User	

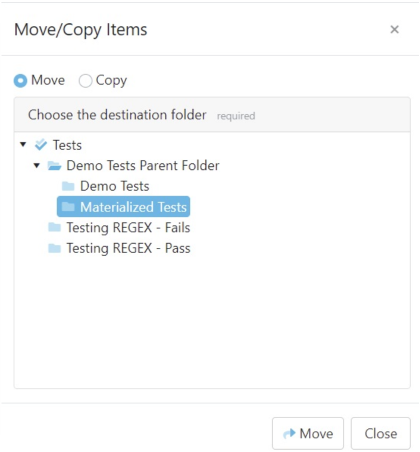
Action Buttons

New	Click on the dropdown to create a new standard test or new template test.
New Folder	Creates a new folder in the current view. There is no hierarchical limit. As many folders and sub-folders can be created as desired.
Open	Open the Test Configuration page for an existing test.
Run	Run a test. See information about the Run Test page.
View History	Navigate to a page displaying the historical pass/fail results for a selected test. See information about the View Test History page.
Rename	Rename a selected test or folder.
Move/Copy	Move or copy one or more selected tests or folders to another folder.
Delete	Delete one or more selected tests or folders.

What's in the Table?

Column Name	Description
Name	The name of the test or test folder.
Template Test Name	The parent template the child test was created from. Click the template test name to edit the template.
Last Modified	Last modified date.
Last Edited By	The user who made the last edit.
Result Trend (past 2 weeks)	The result trend column shows a graphical representation of how many tests related to a table or a column has passed, failed, or executed with an error in the past two weeks. No bar chart means that no test has been executed and saved to the repository in the past two weeks.

Using Move/Copy



Move/Copy Items

● Move ○ Copy

Choose the destination folder required

- Tests
  - Demo Tests Parent Folder
    - Demo Tests
    - Materialized Tests
  - Testing REGEX - Fails
  - Testing REGEX - Pass

To move or copy tests or folders in a project,

1. Highlight the intended test or folder
2. Select the Move/Copy button in the toolbar
3. Choose the Move or Copy radio button according to which function you want to use
4. Find and select the new destination in the folder tree
5. Select either Move or Copy at the bottom of the window

> **Move/Copy Note**
>
> The destination folder must be different than the current folder when moving items.

Access

1. On the navigation pane in the appropriate project, click **Tests.**

Template Tests

Overview

The common challenge of data quality testing is efficiently testing a large volume of data. QA testers tend to thoroughly test critical objects and processes while leaving less critical objects untouched. To help increase test coverage, Validatar can create multiple tests at one time by creating a test from a template. Just choose your target object type and how you want to select the database objects to populate the template.

When to Use a Template Test

When you run into some of these scenarios, try generating tests from a template to cut down on test creation time:

- You have to validate that a common business rule transformation has occurred across multiple tables.
- You have an Excel file with a list of known objects you want to test against.
- Your test cases requires the use of dynamic SQL script configuration because objects may be deleted or updated.
- You've spent over 20 hours creating test scripts and want to cut down on test creation time.

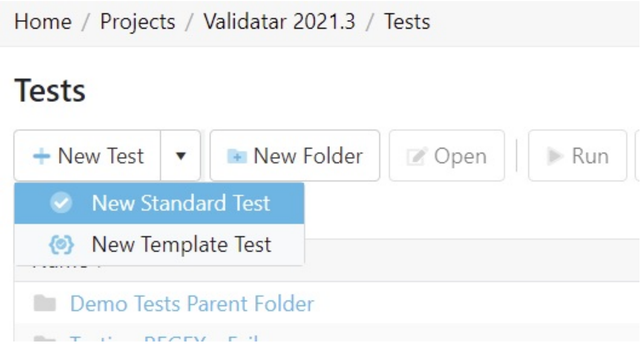
How to Get Started

There are a few concepts to familiarize yourself with before you can start creating tests from a template. These concepts include:

- Understanding metadata references
- Learning about child tests and how to materialize them
- Learning how to import metadata and test requirements via a flat file

View some of the related articles to get started.

Access



To access Template Tests,

1. On the **Test Repository** page, select the New Test dropdown.
2. Select **New Template Test**.

Permissions Note

A user must have Create & Edit Test permissions to create a Template Test.

Materializing Child Tests

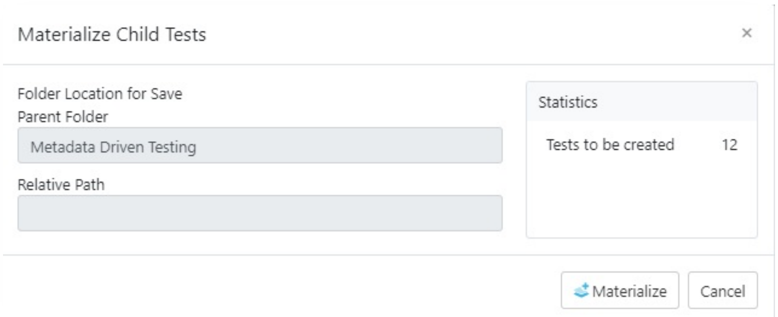
Overview

Materializing child tests create one test per table or column, depending on the selected target object type, for each metadata object that's been selected. Think of materialization as the creation of each test that will run for your designed test scenario. Each time your template is modified, it is recommended to re-materialize child tests to reflect the updated changes. For example, if you modify the metadata object filters to reflect tables that were deleted from your database, re-materialize the child tests so that the old tables won't be included in the scope of the test anymore. You may also want to re-materialize tests before any jobs using those child tests are run.

When tests are materialized each child test will be saved by the name and location that was established on the Child Tests Configuration Tab.

First-Time Materialization

Before you can materialize a test, the template must be saved. The first time a test is materialized, you will see a statistics box with the number of tests to be created as well as the folder location information.



Materialize Child Tests

Folder Location for Save

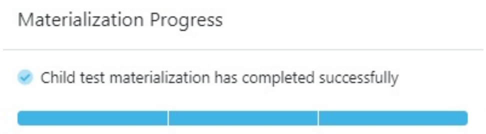
Parent Folder: Metadata Driven Testing

Relative Path:

Statistics

Tests to be created 12

Once the materialization is complete, a message will read that the materialization has been completed successfully.



Materialization Progress

Child test materialization has completed successfully

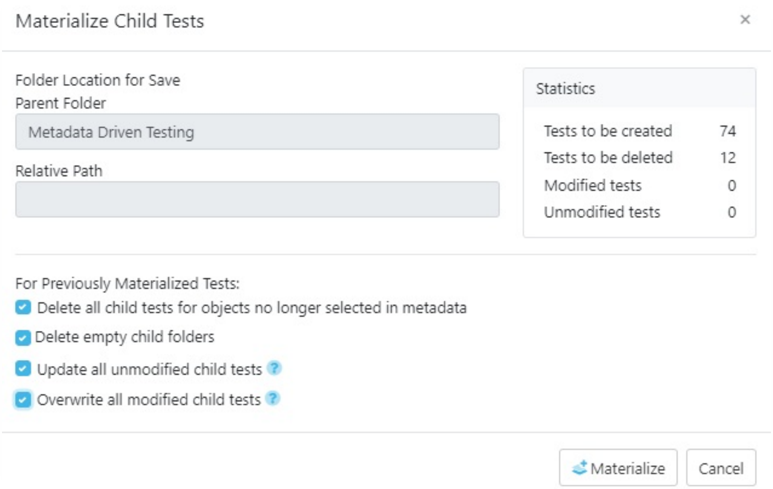
Subsequent Materializations

After initial materialization, as the template test or child tests are edited the Test Summary box on the template screen will update with statistics that reflect any changes.

Once you choose to materialize the child tests again, the pop-up will include configuration options for previously materialized tests.

- Delete all child tests for objects no longer selected in metadata
- Delete empty child folders
- Update all unmodified child tests - updates all previously materialized child tests to the current template version if they haven't been modified since creation.
- Overwrite all modified child tests - overwrites all previously materialized child tests to the current template version if they have been modified since creation.

Any changes to the Child Test Configuration tab are also reflected on the pop-up screen.


Materialize Child Tests

Folder Location for Save

Parent Folder

Metadata Driven Testing

Relative Path

Statistics	
Tests to be created	74
Tests to be deleted	12
Modified tests	0
Unmodified tests	0

For Previously Materialized Tests:

☑ Delete all child tests for objects no longer selected in metadata

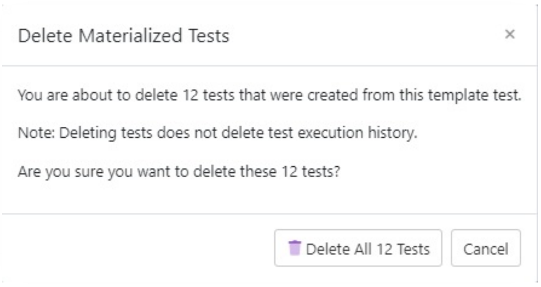
☑ Delete empty child folders

☑ Update all unmodified child tests

☑ Overwrite all modified child tests

Deleting Child Tests

To delete all child tests that were created from the template, use the Delete Child Tests button. To delete only the objects that are no longer selected in the metadata, use the Materialize Child Tests button and then select that configuration option.


Delete Materialized Tests

You are about to delete 12 tests that were created from this template test.

Note: Deleting tests does not delete test execution history.

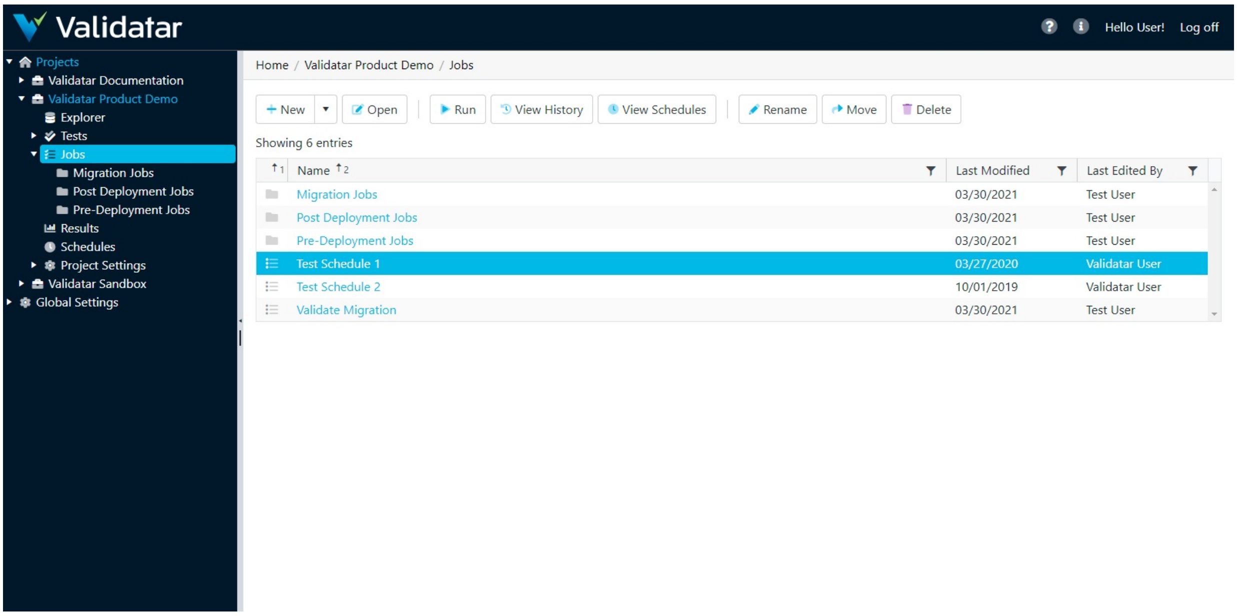
Are you sure you want to delete these 12 tests?

Job Repository

Overview

The Job Repository contains all of the jobs that have been created in a Project.

Displayed Information



Showing 6 entries

↑1	Name ↑2	Last Modified	Last Edited By
	Migration Jobs	03/30/2021	Test User
	Post Deployment Jobs	03/30/2021	Test User
	Pre-Deployment Jobs	03/30/2021	Test User
	Test Schedule 1	03/27/2020	Validatar User
	Test Schedule 2	10/01/2019	Validatar User
	Validate Migration	03/30/2021	Test User

Displayed information can be sorted by clicking on column headers or filtered by clicking on the filter icons also located on the column headers.

Action Buttons

New	Click on the dropdown to create a new job or job folder. There is no hierarchical limit. As many folders and sub-folders can be created as desired.
Open	Opens the Job Configuration page for an existing job.
Run	Runs the selected job.
View History	Navigates to a page displaying the historical pass/fail results for a selected job. More on Job History page.
View Schedules	Navigates to a page displaying schedule(s) assigned to a job. More on Schedules page.
Rename	Renames a selected job.
Move	Move one or more selected jobs.
Delete	Delete one ore more selected jobs.

Table Headers

Job/Job Folder	Job and Job Folders are represented by icons in the first column as shown in the screenshot below.
Name	The name of the job or job folder.
Last Modified Date	The date the job or job folder was last modified.
Last Edited By	The user who last modified the job or job folder.

Access

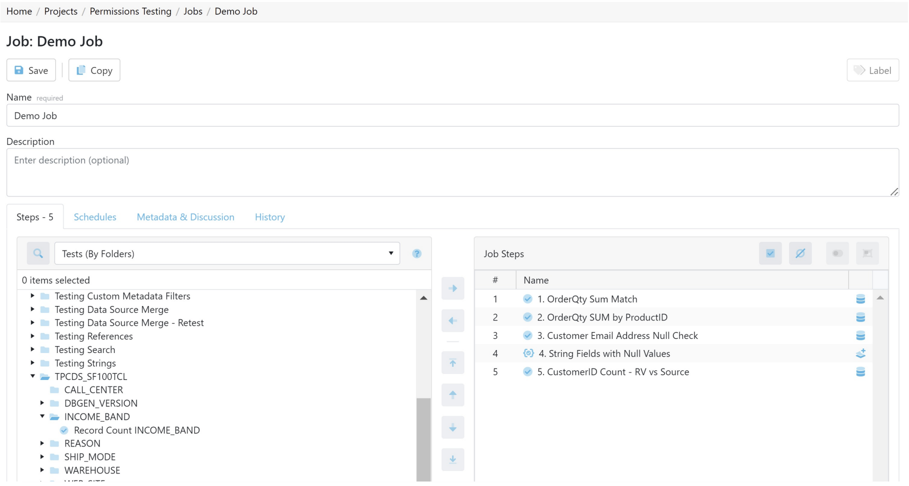
1. Click the dropdown associated with a given project.
2. Click Jobs.
3. Select the desired job.

Configuring a Job

Overview

Jobs provide your data testing and profiling efforts the opportunity to tell a more complete story of what's happening in your environment. By executing tests and profiles together, your test coverage increases which improve the measure of your data quality.

The Job Configuration page provides an interface for choosing the objects to be included in a job, when to run the job, and defining email notifications to send upon job completion.



Home / Projects / Permissions Testing / Jobs / Demo Job

Job: Demo Job

Name required: Demo Job

Description: Enter description (optional)

0 items selected

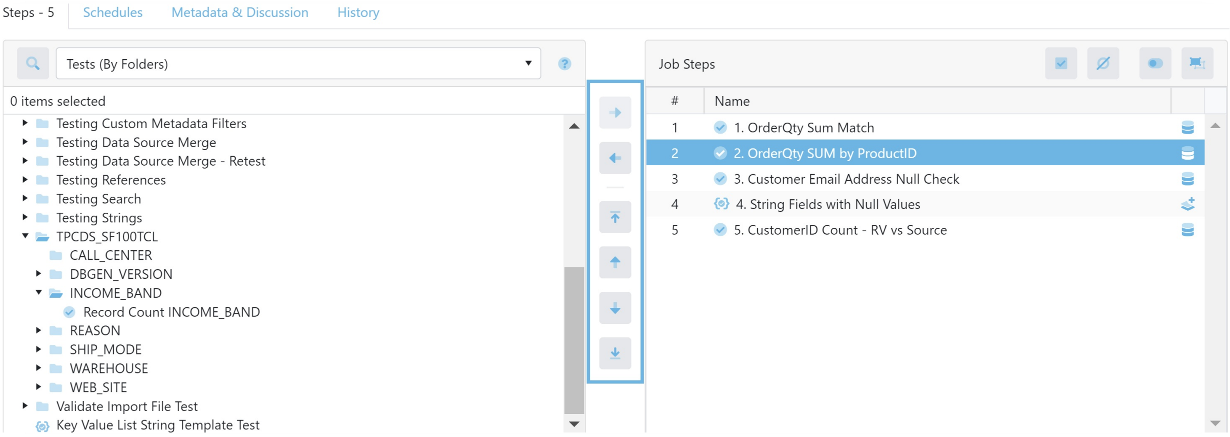
- Testing Custom Metadata Filters
- Testing Data Source Merge
- Testing Data Source Merge - Retest
- Testing References
- Testing Search
- Testing Strings
- TPCDS_SF100TCL
  - CALL_CENTER
  - DBGEN_VERSION
  - INCOME_BAND
    - Record Count INCOME_BAND
  - REASON
  - SHIP_MODE
  - WAREHOUSE

Job Steps

#	Name
1	1. OrderQty Sum Match
2	2. OrderQty SUM by ProductID
3	3. Customer Email Address Null Check
4	4. String Fields with Null Values
5	5. CustomerID Count - RV vs Source

Job Steps

The Steps section of the Job Configuration page consists of a list of available tests and jobs on the left-hand portion of the screen and a list of selected job steps on the right-hand portion of the screen tagged as Job Steps.



0 items selected

- Testing Custom Metadata Filters
- Testing Data Source Merge
- Testing Data Source Merge - Retest
- Testing References
- Testing Search
- Testing Strings
- TPCDS_SF100TCL
  - CALL_CENTER
  - DBGEN_VERSION
  - INCOME_BAND
    - Record Count INCOME_BAND
  - REASON
  - SHIP_MODE
  - WAREHOUSE
  - WEB_SITE
- Validate Import File Test
- Key Value List String Template Test

Job Steps

#	Name
1	1. OrderQty Sum Match
2	2. OrderQty SUM by ProductID
3	3. Customer Email Address Null Check
4	4. String Fields with Null Values
5	5. CustomerID Count - RV vs Source

There are four types of objects that can be added to a job:

1. Standard Tests
2. Template Tests
3. Jobs
4. Profile Sets

Available objects can be found using the Tests (By Folders) dropdown or by toggling to the search view using the search icon to the left of the dropdown.

Job Step Buttons

Arrow buttons (*highlighted in the above image*)	Allow a user to : ● Add an object to a job ● Remove an object from a job ● Move an object to the top of the list ● Move an object up on the list ● Move an object down the list ● Move an object to the bottom of the list **Note:** Adding the same object to the Job Step list produces a warning icon about the duplicate job step.
Select All	Selects all items on the job steps list.
Deselect All	Deselects all items on the job steps list.
Disable Job Step	Disables the selected job step from executing when running the job.
Group / Ungroup Job Steps	Group job steps together to run them in parallel. Ungroup job steps to run them sequentially. Grouped steps share the same step number. Ctrl + click on multiple tests within Job Steps and then click on the Group button to create a group of tests.

Job Configuration Note

Clicking the database icon in Job Steps allows the user to choose source and target connections that differ from the default for standard tests. If a different connection is chosen, the database icon will turn blue. Similarly for runtime configuration of Template Tests, clicking the materialize icon allows the user to enable child test materialization before running the job. If this option is enabled, the materialize icon will turn blue.

Schedules

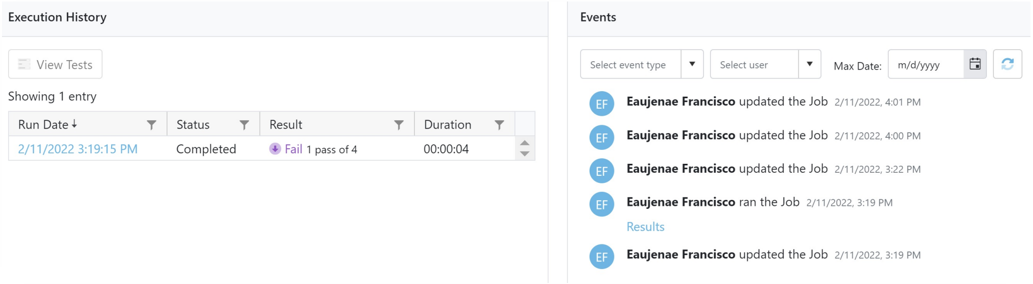
Each job can be assigned one more schedules designed to run the job automatically. Check out Scheduling a Job to learn more about creating a schedule.

Metadata & Discussion

Similar to tests and metadata objects, jobs are associated with custom fields and have a section for users to leave comments. Have a user with Edit Metadata permission update the custom fields for Jobs.

History

View the execution history of each job to see the results over time. You can also see when users create, modify, or copy the job using Event History.



Showing 1 entry

Run Date ↓	Status	Result	Duration
2/11/2022 3:19:15 PM	Completed	Fail 1 pass of 4	00:00:04

EF **Eaujenae Francisco** updated the Job 2/11/2022, 4:01 PM

EF **Eaujenae Francisco** updated the Job 2/11/2022, 4:00 PM

EF **Eaujenae Francisco** updated the Job 2/11/2022, 3:22 PM

EF **Eaujenae Francisco** ran the Job 2/11/2022, 3:19 PM
Results

EF **Eaujenae Francisco** updated the Job 2/11/2022, 3:19 PM

Scheduling a Job

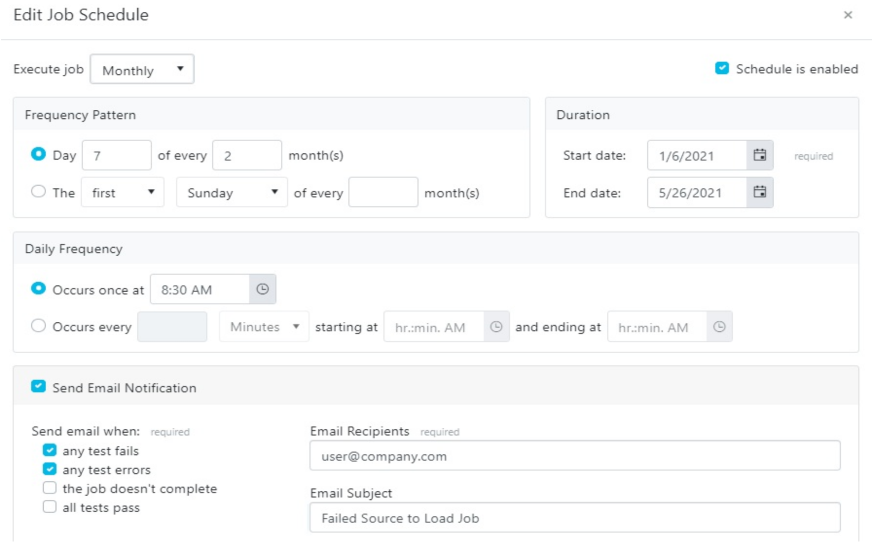
Overview

Scheduling a Job allows users to receive the results of their test and profile executions without having to be inside of Validatar. This feature is ideal for sending reports to higher level managers, informing data warehouse developers of issues as they arise, and providing peace of mind that your data is constantly being monitored.

The Job Schedule Configuration screen allows the user to define job scheduling and email notification.
A job schedule can be configured in many ways. Some examples include:

- A daily configuration that recurs every day at 5:00 AM after source system data has been updated.
- A weekly configuration that recurs every Monday to monitor changes in the data warehouse.
- A monthly configuration that recurs on the 2nd Friday of each month in preparation for sending status updates to higher level managers.

Displayed Information



Edit Job Schedule

Execute job: Monthly

☑ Schedule is enabled

Frequency Pattern

● Day 7 of every 2 month(s)

○ The first Sunday of every month(s)

Duration

Start date: 1/6/2021 required

End date: 5/26/2021

Daily Frequency

● Occurs once at 8:30 AM

○ Occurs every Minutes starting at hr.:min. AM and ending at hr.:min. AM

☑ Send Email Notification

Send email when: required

☑ any test fails

☑ any test errors

□ the job doesn't complete

□ all tests pass

Email Recipients required: user@company.com

Email Subject: Failed Source to Load Job

When scheduling a job, email notifications can be configured with the following options:

- Choose to send an email when a job is completed with a failure or error, when the job doesn't complete at all, or when a job is completed successfully with no failures or errors.
- Enter recepients emails using a comma or semi-colon delimited list.
- Provide an email subject line.

Note

It's easy to overlook whether a schedule is enabled or disabled. Verify that the schedule is enabled if you intend for it to run.

Access

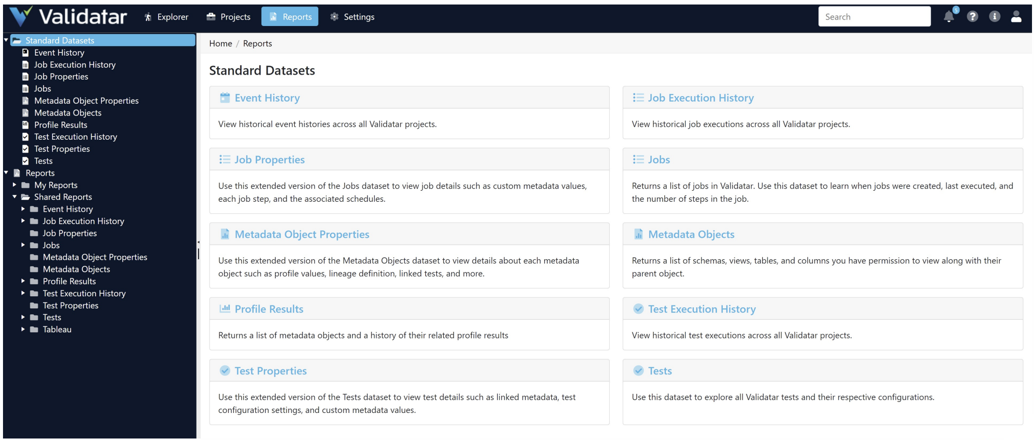
Existing Job Schedules can be accessed by clicking Schedules within the navigation pane. New Job Schedules can be created via the following steps:

1. Click the dropdown associated with a given project.
2. In the navigation pane click Jobs.
3. Click on the desired job.
4. Click the Schedules tab.
5. Click New.

Reports Home Page

Overview

A report displays detailed information about the tests, jobs, metadata, and profiles in Validatar. You can also embed Tableau Server dashboards in Validatar by creating a custom report.



Standard Datasets

Event History

View historical event histories across all Validatar projects.

Job Execution History

View historical job executions across all Validatar projects.

Job Properties

Use this extended version of the Jobs dataset to view job details such as custom metadata values, each job step, and the associated schedules.

Jobs

Returns a list of jobs in Validatar. Use this dataset to learn when jobs were created, last executed, and the number of steps in the job.

Metadata Object Properties

Use this extended version of the Metadata Objects dataset to view details about each metadata object such as profile values, lineage definition, linked tests, and more.

Metadata Objects

Returns a list of schemas, views, tables, and columns you have permission to view along with their parent object.

Profile Results

Returns a list of metadata objects and a history of their related profile results

Test Execution History

View historical test executions across all Validatar projects.

Test Properties

Use this extended version of the Tests dataset to view test details such as linked metadata, test configuration settings, and custom metadata values.

Tests

Use this dataset to explore all Validatar tests and their respective configurations.

Why Use Reports?

Reports help to answer questions about the status of your data quality. Need to send a status update? Send a report instead. Want to get consistent notifications about tests executed during a period of time? Create and subscribe to a report that details that information.

How Reports Work

Reports are driven by built-in datasets. These datasets are, in turn, driven by secured views against the Validatar database. Want to create connect the Validatar database to your own reporting tool? Use the secured views to create visualizations and reports.

Each dataset can be viewed on its own or can be used to create a report which is shareable with the entire team. Create a report by selecting the dataset from the navigation pane and clicking **Create New Report**. Customize the report by editing the columns the report displays and adding filters to return specific information.

Standard Datasets

- **Tests**: Explore All Validatar tests and their respective configurations.
- **Jobs**: Returns a list of jobs in Validatar. Use this dataset to learn when jobs were created, last executed, and the number of steps in the job.
- **Metadata Objects**: Returns a list of schemas, tables, and columns you have permission to view along with their parent object.
- **Test Execution History**: View historical test executions across all Validatar projects.
- **Job Execution History**: View historical job executions across all Validatar projects.
- **Event History**: View historical event histories across all Validatar projects.
- **Test Properties**: Use this extended version of the Tests dataset to view test details such as linked metadata, test configuration settings, and custom metadata values.
- **Job Properties**: Use this extended version of the Jobs dataset to view job details such as custom metadata values, each job step, and the associated schedules.
- **Metadata Object Properties**: Use this extended version of the Metadata Objects dataset to view details about each metadata object such as profile values, lineage definition, linked tests, and more.

Tableau Reporting

You can view Tableau Server reports and dashboards inside of Validatar by connecting your organization's Tableau Server account. Learn more about connecting to Tableau.

Creating Reports

Creating Reports

To create a report:

1. Select **Reports** in the main header bar.
2. Select the standard dataset you want to build the report from in the navigation pane.
3. Click **Create New Report**.
4. Add or remove columns you want to display by clicking **Edit Columns**. You'll see the number of displayed columns on the button.
5. Add table filters to your report by clicking **Edit Filters**.
6. Add the report name, description, and select who can view the report by clicking **Configure**.
    1. Report Visibility can either be public, private, or accessible by one or more user groups.
7. Click **Save**.

Subscribing to Reports

Reports created from the **Event History, Test Execution History, and Job Execution Histor**y datasets can be configured to be sent to users' emails.

Switch over to the **Subscriptions** tab and add a new subscription. Unlike job results which can be sent to anyone's email address, the recipient must be a Validatar user.

Exporting Reports

Datasets and Reports can also be viewed outside of Validatar by exporting to an Excel file. On the dataset or report, apply the appropriate columns and filters and then select **Export to File**.

Validatar automatically exports the first 10,000 records in the dataset or report. For files larger than this, the user has the option to export the entire file or just the first 10,000 records.

Managing Reports

Overview

You can create a new Test or Job Execution History report by expanding the **My Report Views** folder on the Reports tree and selecting **Manage Reports**.

You can also embed Tableau Server dashboards in Validatar in the **Manage Reports** section. To create a Tableau report, you must have the Tableau **Workbook Name** and the **View Name**.

Creating a Job or Test Execution History Report

From the Reports tree, click **Manage Reports**.
2. Click **New Report**.
4. Enter the report name.
5. Choose **Job Execution History** or **Test Execution History** as the report type.
6. Enter a description. (optional)
7. Choose the report visibility.
8. Click **Save**.

Creating a New Tableau Report

Prerequisite Required

Before you create a new report, ensure that a global admin has connected your organization's Tableau Server to Validatar. This setting is found under **Settings** > **Configuration** > **Reporting**. Learn more about this setup here.

Any Validatar user can create a new report.

1. From the Reports tree, click **Manage Reports**.
2. Click **New Report**.
3. Enter the report name.
4. Choose **Tableau** as the report type.
5. Enter a description. (optional)
6. Choose the report visibility.
7. Enter the Tableau report path in the following format:
    - [Workbook Name]/[View Name]
8. Click **Save**.

Managing Reports

1. From the Reports tree, click **Manage Reports**.
2. Select either the white space next to the name and click **Open** or click the configuration icon next to the name to view the report details.
3. Make any necessary changes, then click **Save**.
4. To delete a report, select the white space next to the name and click **Delete**.

Delete Note

You can **restore** a report immediately after deleting it. **Once you navigate away from the reports page, you cannot restore the report.**

Connecting Tableau to Validatar

Overview

Connecting to Tableau requires a **Tableau Server** account and user accounts with access to the intended workbooks and views.

Server Configuration

> **Prerequisite Required**
>
> To set up the Tableau Server connection, the user must be a **Configuration Admin** or a **Global Admin**.

1. Click **Settings** on the main menu bar.
2. On the navigation tree, select **Configuration**.
3. On the Validatar Configuration list, click **Reporting**.
4. Enter the server configuration settings.
    - **Tableau Server URL:** The URL to your Tableau Server account. Example (https://tableau.xyzcorp.com)
    - **Tableau Site (optional):** Specifies a site on a Tableau Server instance. Sites are used to separate groups of users and content on the same Tableau Server instance.
    - **Identity Store:** This is chosen when Tableau Server is installed.
        - Local Store: All user and group information is stored and managed in the Tableau Server repository.
        - External Store (AD or LDAP): An external directory service stores and manages user and group information.
5. Click **Save**.

How Users Access Workbooks

To view the embedded dashboards, a user must:

1. Have a Tableau server user account with permissions to view the workbook.
2. Have their Tableau server username added to Validatar.

Adding Tableau Account to Validatar

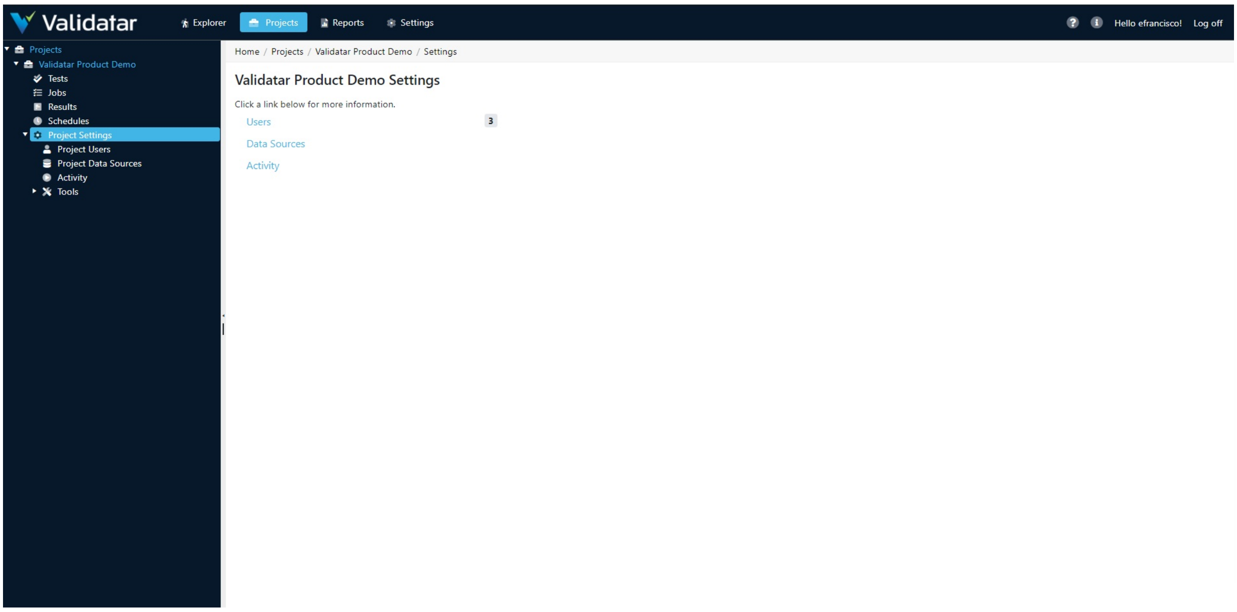
To add a user's Tableau Server account to Validatar:

1. A **Security Admin** or a **Global Admin** must navigate to **Settings** > **Users** > **User Details**.
2. Enter their **Tableau Username** in the provided field.
3. Click **Save**.

Project Settings

Overview

The **Project Settings** page reflects the same choices available within the navigation pane under the Project Settings dropdown; that is, access to settings for Data Sources, Users, Groups, Activity, and Tools. As the name implies, Project Settings are limited to a given project, whereas **Global Settings** are application-wide.



Validatar Product Demo Settings

Click a link below for more information.

Users 3

Data Sources

Activity

Included in Project Settings are:

- Project Users
- Project Data Sources
- Activity
- Tools
  - Labels
  - Import
  - Export

For information on Project Data Sources, see the Data Sources section.

Access

1. The Project Settings page may be directly accessed by expanding a project and clicking on Project Settings in the navigation pane.

Import

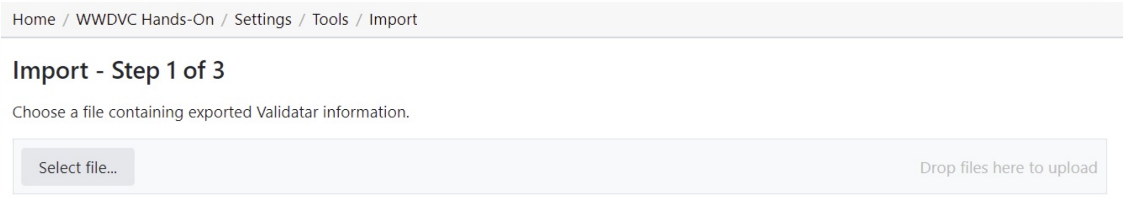
Overview

The Import feature allows an XML file that contains the contents of a project to be imported into another project. Importing is a three-step process:

- Step 1: Select a file
- Step 2: Map the data sources and choose the target import folder
- Step 3: Review the changes

Import: Step 1

Select an XML file containing exported Validatar objects. Invalid file types and XML files incorrectly structured will not be accepted. To ensure you have the correct file to import, export objects from Validatar first for reference.


Home / WWDVC Hands-On / Settings / Tools / Import

Import - Step 1 of 3

Choose a file containing exported Validatar information.

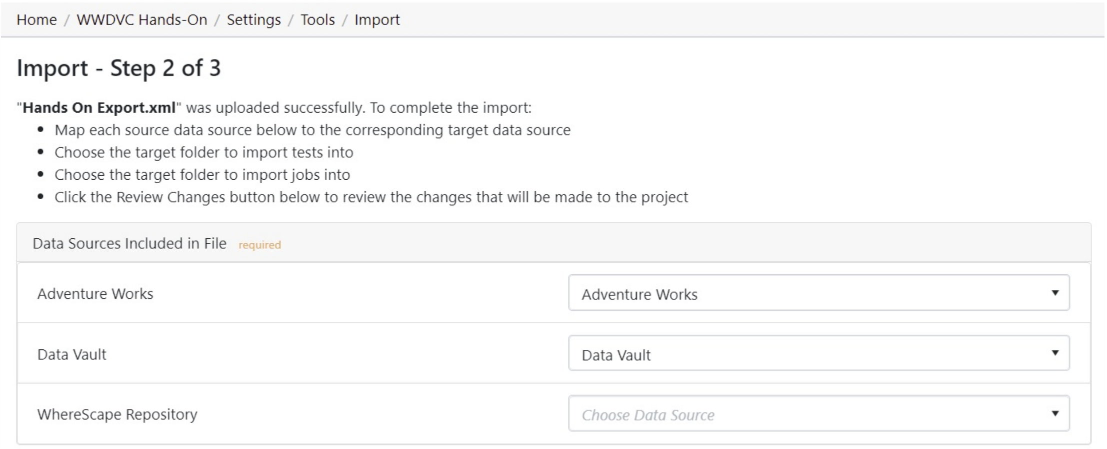
Select file...
Drop files here to upload

Import: Step 2

Mapping Data Sources

If a data source in the project you're importing into has the same name and connection string as a data source found in the imported XML file, the data sources will automatically be mapped to each other. This default mapping can be changed if necessary.

For data sources where an identical name and connection string is not found, you will have to choose the respective data source.



Import - Step 2 of 3

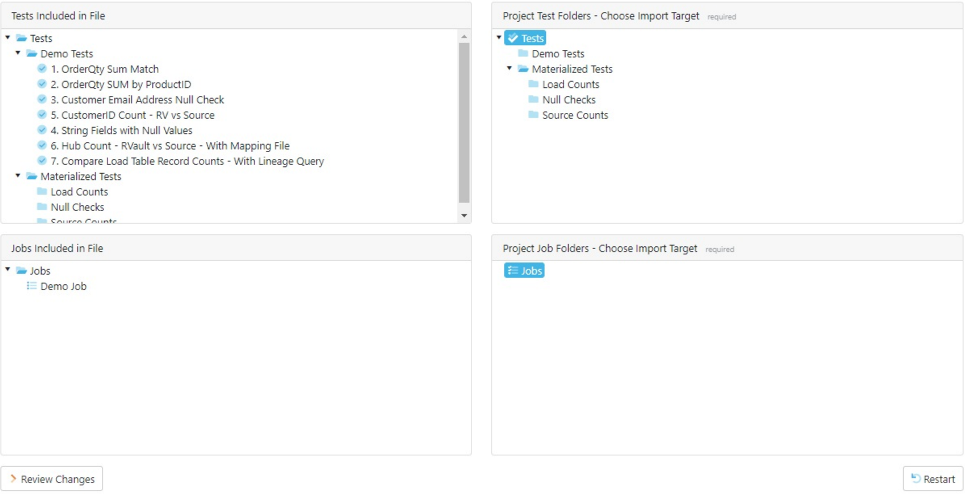
"**Hands On Export.xml**" was uploaded successfully. To complete the import:

- Map each source data source below to the corresponding target data source
- Choose the target folder to import tests into
- Choose the target folder to import jobs into
- Click the Review Changes button below to review the changes that will be made to the project

Data Sources Included in File required	
Adventure Works	Adventure Works
Data Vault	Data Vault
WhereScape Repository	Choose Data Source

Choosing the Target Folder

All the tests and jobs included in the file will appear on the left pane. Choose a single folder on the test and job levels to import every object included in the file. After the selection is made, click **Review Changes**.



Tests Included in File

- Tests
  - Demo Tests
    - 1. OrderQty Sum Match
    - 2. OrderQty SUM by ProductID
    - 3. Customer Email Address Null Check
    - 5. CustomerID Count - RV vs Source
    - 4. String Fields with Null Values
    - 6. Hub Count - RVault vs Source - With Mapping File
    - 7. Compare Load Table Record Counts - With Lineage Query
  - Materialized Tests
    - Load Counts
    - Null Checks
    - Source Counts

Project Test Folders - Choose Import Target required

- ✓ Tests
  - Demo Tests
  - Materialized Tests
    - Load Counts
    - Null Checks
    - Source Counts

Jobs Included in File

- Jobs
  - Demo Job

Project Job Folders - Choose Import Target required

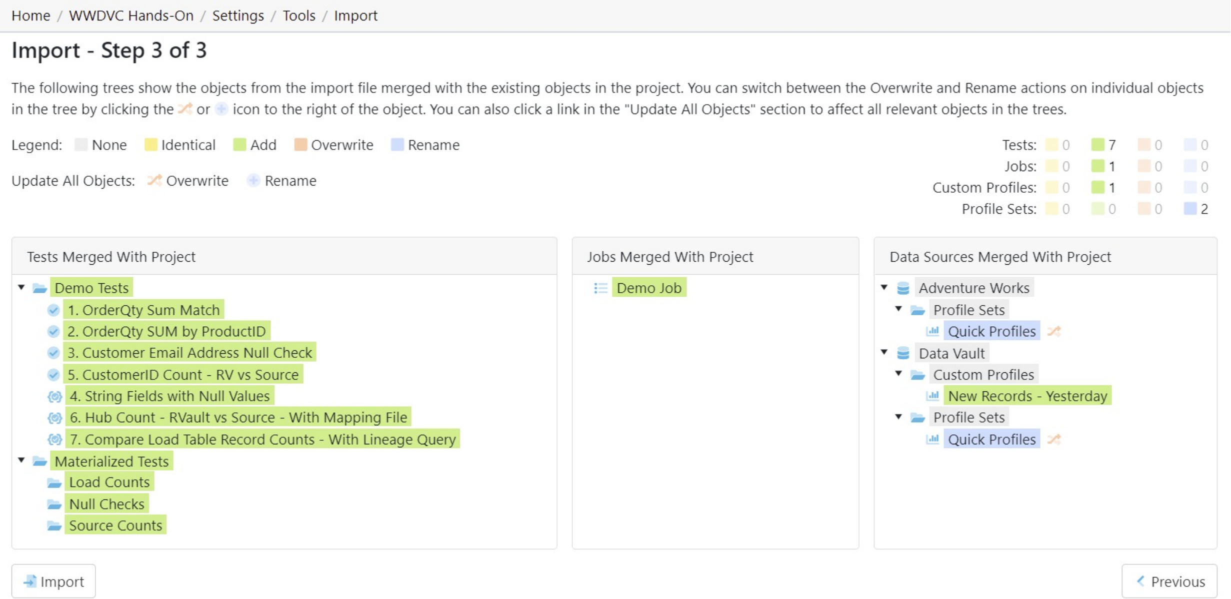
- Jobs

Import: Step 3

The final step of the import is to review the objects that will be added to the project. Each object is highlighted based on whether the object currently exists in the project or will be added.

- Identical - The object already exists in the current project. This requires the user to either overwrite or rename the object.
- Add - The object does not exist in the current project and will be added.
- Overwrite - If the object already exists, replace it with the new object by overwriting it.
- Rename - If the object already exists, duplicate the object and rename it.

To toggle between overwrite and rename, click the icon next to the object.



Import - Step 3 of 3

The following trees show the objects from the import file merged with the existing objects in the project. You can switch between the Overwrite and Rename actions on individual objects in the tree by clicking the or icon to the right of the object. You can also click a link in the "Update All Objects" section to affect all relevant objects in the trees.

Legend: None Identical Add Overwrite Rename

Update All Objects: Overwrite Rename

Tests:	0	7	0	0
Jobs:	0	1	0	0
Custom Profiles:	0	1	0	0
Profile Sets:	0	0	0	2

Tests Merged With Project

- Demo Tests
  - 1. OrderQty Sum Match
  - 2. OrderQty SUM by ProductID
  - 3. Customer Email Address Null Check
  - 5. CustomerID Count - RV vs Source
  - 4. String Fields with Null Values
  - 6. Hub Count - RVault vs Source - With Mapping File
  - 7. Compare Load Table Record Counts - With Lineage Query
- Materialized Tests
  - Load Counts
  - Null Checks
  - Source Counts

Jobs Merged With Project

- Demo Job

Data Sources Merged With Project

- Adventure Works
  - Profile Sets
    - Quick Profiles
- Data Vault
  - Custom Profiles
    - New Records - Yesterday
  - Profile Sets
    - Quick Profiles

Finalizing the Import

Once you review all changes to be made to the project, select **Import**. Navigate back to the Tests and Jobs area to confirm the import was successful.

Access

1. Navigate to **Tools** found under **Project Settings**.

2. Select **Import**.

Export

Overview

The export feature allows selected tests, template tests, profile sets, and jobs in a particular project to be exported to an XML file. The XML file can then be used to import the contents into another project.

A user may want to export Validatar objects to:

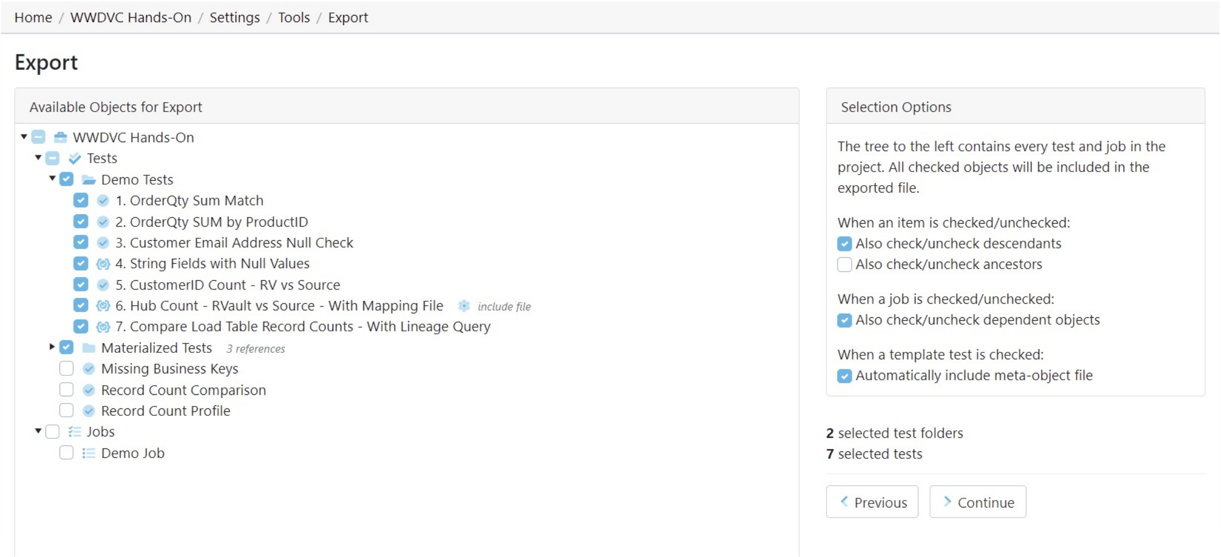
- Create a backup of a Validatar project
- Make mass changes to the contents of a test or job
- Duplicate tests or jobs for use in a different development environment (e.g. Running the same set of tests configured against the DEV environment to the TEST environment).

There are two ways to export Validatar objects

- via Folder path selection
- via Label selection (requires the project to have at least one locked label)

Exporting via Folder Path Selection

Exporting via folder path selection allows you to select available objects based on their hierarchical location in Validatar. Expand and collapse tests, jobs, and folders to show or hide certain objects.



Export

Available Objects for Export

- ⊟ WWDVC Hands-On
  - ⊟ Tests
    - ☑ Demo Tests
      - ☑ 1. OrderQty Sum Match
      - ☑ 2. OrderQty SUM by ProductID
      - ☑ 3. Customer Email Address Null Check
      - ☑ 4. String Fields with Null Values
      - ☑ 5. CustomerID Count - RV vs Source
      - ☑ 6. Hub Count - RVault vs Source - With Mapping File *include file*
      - ☑ 7. Compare Load Table Record Counts - With Lineage Query
    - ☑ Materialized Tests *3 references*
    - ☐ Missing Business Keys
    - ☐ Record Count Comparison
    - ☐ Record Count Profile
  - ☐ Jobs
    - ☐ Demo Job

Selection Options

The tree to the left contains every test and job in the project. All checked objects will be included in the exported file.

When an item is checked/unchecked:
☑ Also check/uncheck descendants
☐ Also check/uncheck ancestors

When a job is checked/unchecked:
☑ Also check/uncheck dependent objects

When a template test is checked:
☑ Automatically include meta-object file

2 selected test folders
7 selected tests

Selection Options

Every object that is checked on the Available Objects for Export list will be included in the exported XML file. You can configure how the object selection should work using the section called **Selection Options**.

When an item is checked and unchecked choose to:

- **Check and uncheck descendants**. This allows all the children objects inside of folders to be selected or deselected for export.
- **Check and uncheck ancestors**. This allows all the parent objects of the item to be selected or deselected for export.

When a job is checked and unchecked choose to:

- **Check and uncheck dependent objects**. This allows the objects within a job to be selected or deselected for export. Learn more about dependent job configuration in the next section.

When a template test is checked choose to:

- **Automatically include the meta-object file**. Checking this option ensures the import file is included with the template test if the template test is driven by an import file.

Job Configuration

When **Also check/uncheck dependent objects for a job** is selected, each job that is selected in the folder tree will have a configuration icon appear next to the job name so the user can choose which dependent objects of the job should be included in the export. Selecting the configuration icon opens the following window.

If you choose to remove an item from the item list one of two things may happen when it's time to import the file:

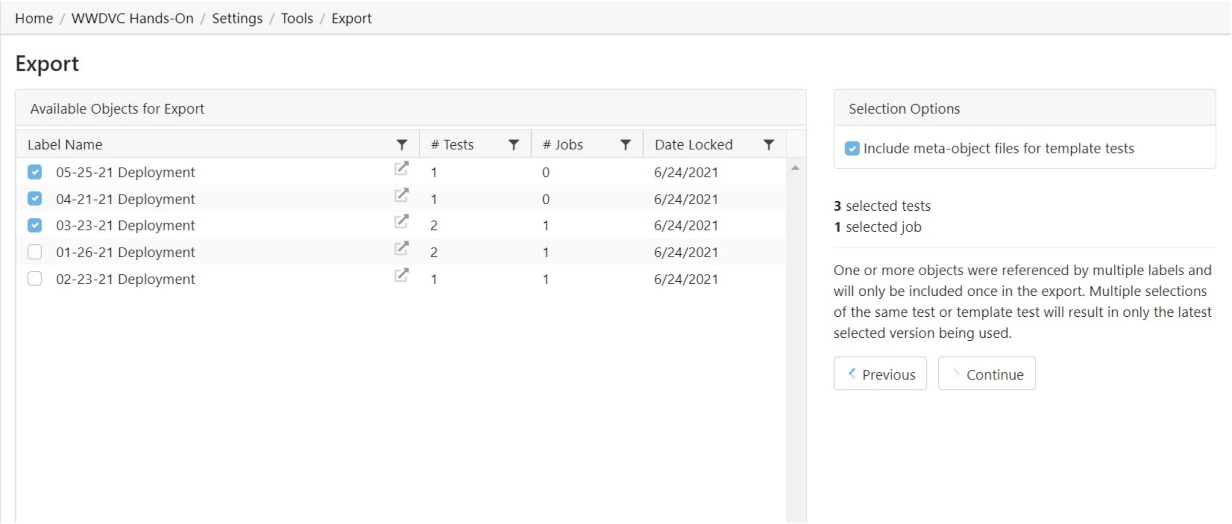
- If an item name with the same hierarchy of the removed item name is found in the target project, the job step will be linked to that item.
- If an item name is **not** found, the job step will be removed from the job upon import.

Learn more about the last export step in the Finalizing the Export section below.

Exporting via Label Selection

Exporting via label selection allows you to select all objects related to locked labels associated with tests, template tests, or jobs. Only locked labels will show up on the export list and multiple labels can be selected for export at once.

To view the configuration of the locked label, select the external view icon to the right of the label name.



Home / WWDVC Hands-On / Settings / Tools / Export

Export

Available Objects for Export

	Label Name	# Tests	# Jobs	Date Locked
☑	05-25-21 Deployment	1	0	6/24/2021
☑	04-21-21 Deployment	1	0	6/24/2021
☑	03-23-21 Deployment	2	1	6/24/2021
☐	01-26-21 Deployment	2	1	6/24/2021
☐	02-23-21 Deployment	1	1	6/24/2021

Selection Options

☑ Include meta-object files for template tests

3 selected tests
1 selected job

One or more objects were referenced by multiple labels and will only be included once in the export. Multiple selections of the same test or template test will result in only the latest selected version being used.

Selection Options

The selection options include an aggregate of how many tests and jobs will be included in the export as the labels are selected on the available objects pane.

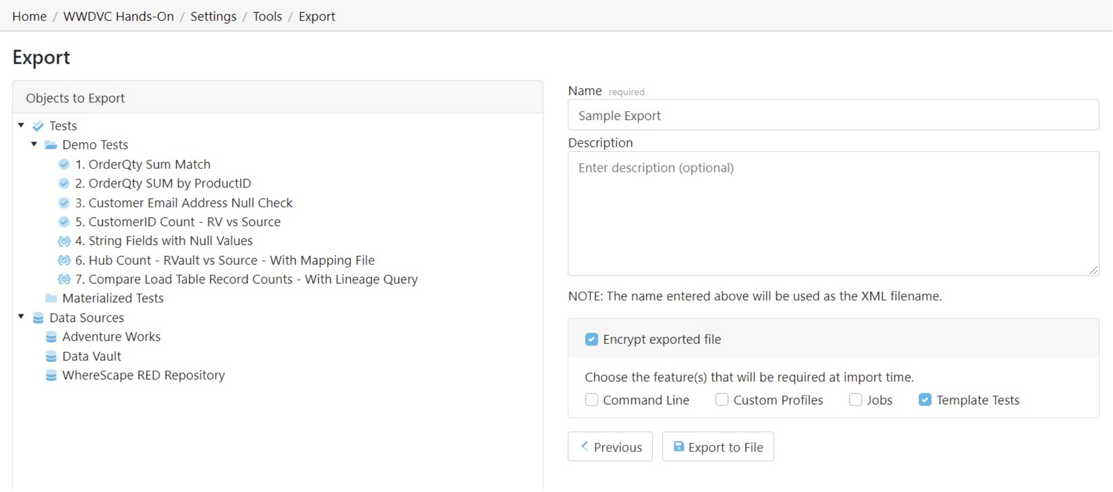
If a test, template test, or job is referenced by multiple selected labels, the object will only be included in the export once. If multiple versions of the same test or template test are selected, only the latest selected version will be used.

When checked, the **Include meta-object files for template tests** option includes the import file that drives the respective template test.

Finalizing the Export

The final step of the export includes adding a name and description (optional) to the exported XML file. You will also see a folder tree of all of the objects that will be included in the export file. This includes tests, folders, template tests, jobs, profile sets, and data sources.

Upon export, the file will be automatically downloaded and can be found in your Downloads folder. The destination of the export file cannot be changed.



Home / WWDVC Hands-On / Settings / Tools / Export

Export

Objects to Export

- ▾ ✓ Tests
  - ▾ Demo Tests
    - 1. OrderQty Sum Match
    - 2. OrderQty SUM by ProductID
    - 3. Customer Email Address Null Check
    - 5. CustomerID Count - RV vs Source
    - 4. String Fields with Null Values
    - 6. Hub Count - RVault vs Source - With Mapping File
    - 7. Compare Load Table Record Counts - With Lineage Query
  - Materialized Tests
- ▾ Data Sources
  - Adventure Works
  - Data Vault
  - WhereScape RED Repository

Name required

Sample Export

Description

Enter description (optional)

NOTE: The name entered above will be used as the XML filename.

☑ Encrypt exported file

Choose the feature(s) that will be required at import time.

☐ Command Line ☐ Custom Profiles ☐ Jobs ☑ Template Tests

Encrypting the Export File

Encrypting an Export File

If you would like to require the Validatar instance that will use the exported file to only allow the file to be imported if the instance has certain features, you can encrypt the file.

When checked, the Encrypt export file displays the following feature options the Validatar instance must have to accept the import:

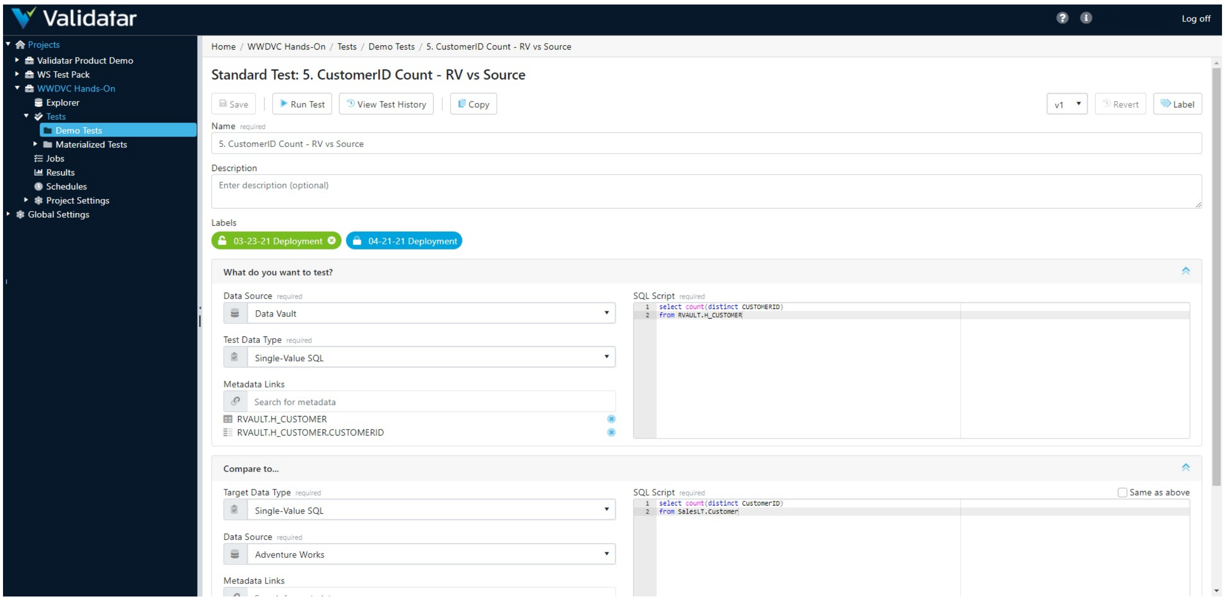
Command Line
Custom Profiles
Jobs
Template Tests

Upon import, if the Validatar instance's current license key will be evaluated to ensure the appropriate features are included in the license.

Labels Overview

Overview

Labels are a way to group Validatar objects. As items in your development environment change, new deployments happen, and test cases must be updated, labels can be used to identify objects that are a part of the same deployment.



Standard Test: 5. CustomerID Count - RV vs Source

Name required: 5. CustomerID Count - RV vs Source

Description: Enter description (optional)

Labels: 03-23-21 Deployment, 04-21-21 Deployment

What do you want to test?

Data Source required: Data Vault

Test Data Type required: Single-Value SQL

Metadata Links: Search for metadata
- RVAULT.H_CUSTOMER
- RVAULT.H_CUSTOMER.CUSTOMERID

SQL Script required

```
select count(distinct CUSTOMERID)
from RVAULT.H_CUSTOMER
```

Compare to...

Target Data Type required: Single-Value SQL

Data Source required: Adventure Works

Metadata Links

SQL Script required ☐ Same as above

```
select count(distinct CustomerID)
from SalesLT.Customer
```

When to Use Labels

There are many reasons why you would choose to use labels. The main two being:

1. The ability to identify a version of a set of tests.
   - Let's say you have a sprint release where a large number of changes will be delivered. Creating a label for each sprint will allow you to go back in time to refer to every test, template test, and job involved in that sprint.
   - To distinguish your labels, your label name might include versioning information such as the version number, delivery date, the development ticket number.
2. The ability to move a set of tests from one Validatar environment to another.
   - The simplest example here is if you've finished testing in your Validatar QA environment and you're ready to deploy your code to Production, having a label for that deployment makes it easier to move the QA tests you've completed to your Validatar Production environment to be executed.

Export Note

When exporting Validatar objects by label selection, you can not exclude any objects during the import. Be sure to include only what you plan to import in your XML document.

Label Management

Permissions

To manage labels, a user must have the **Create and Edit Labels** role found in Project User Permissions. Other users can view the label details in read-only mode by having Project Access permissions.

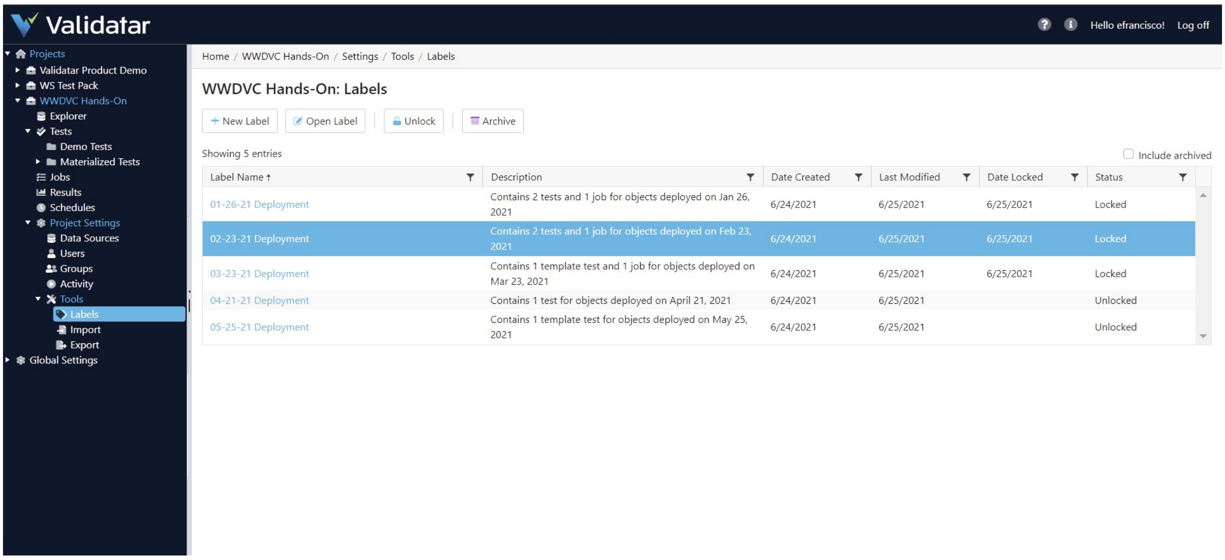
On the Label list screen you can:

- Create a new label
- Modify an existing unlocked label
- Lock or Unlock a label
- Archive or Restore a label

Creating a Label

Creating a new label is easy:

1. Select **New Label**.
2. Enter the label name and optional description and additional notes.
3. Click **Save**.

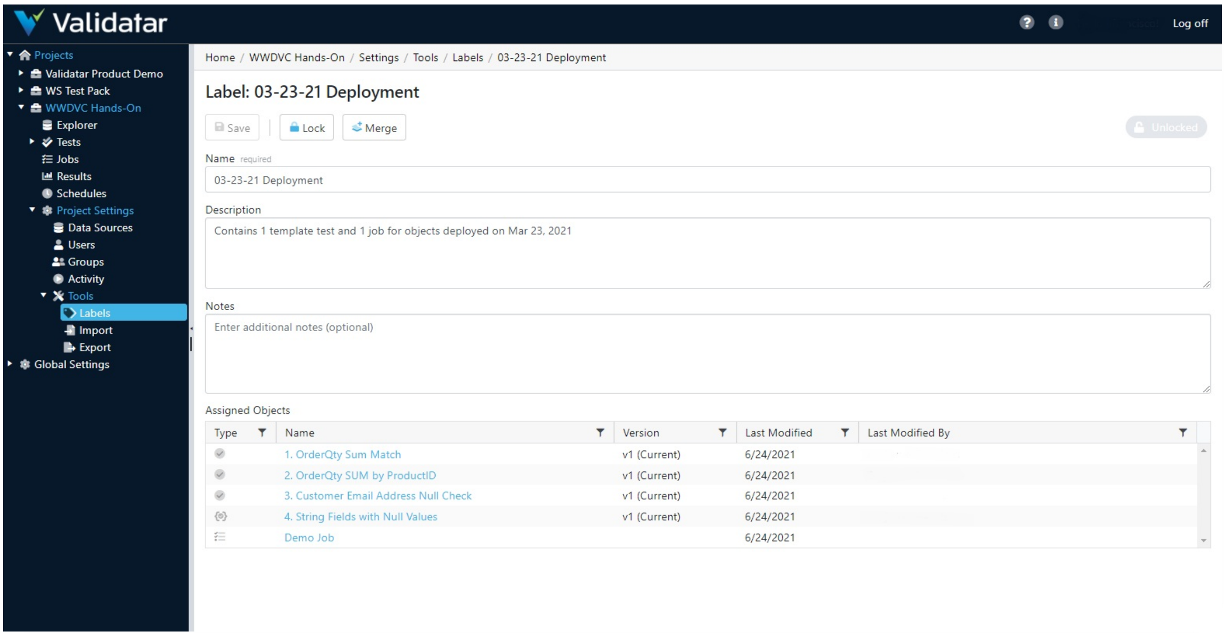


WWDVC Hands-On: Labels

Showing 5 entries

Label Name ↑	Description	Date Created	Last Modified	Date Locked	Status
01-26-21 Deployment	Contains 2 tests and 1 job for objects deployed on Jan 26, 2021	6/24/2021	6/25/2021	6/25/2021	Locked
02-23-21 Deployment	Contains 2 tests and 1 job for objects deployed on Feb 23, 2021	6/24/2021	6/25/2021	6/25/2021	Locked
03-23-21 Deployment	Contains 1 template test and 1 job for objects deployed on Mar 23, 2021	6/24/2021	6/25/2021	6/25/2021	Locked
04-21-21 Deployment	Contains 1 test for objects deployed on April 21, 2021	6/24/2021	6/25/2021		Unlocked
05-25-21 Deployment	Contains 1 template test for objects deployed on May 25, 2021	6/24/2021	6/25/2021		Unlocked

Once a label is created and is assigned to objects, you can view all the associated objects on the label details page.



Label: 03-23-21 Deployment

Name required: 03-23-21 Deployment

Description: Contains 1 template test and 1 job for objects deployed on Mar 23, 2021

Notes: Enter additional notes (optional)

Assigned Objects

Type	Name	Version	Last Modified	Last Modified By
✓	1. OrderQty Sum Match	v1 (Current)	6/24/2021	
✓	2. OrderQty SUM by ProductID	v1 (Current)	6/24/2021	
✓	3. Customer Email Address Null Check	v1 (Current)	6/24/2021	
{o}	4. String Fields with Null Values	v1 (Current)	6/24/2021	
☰	Demo Job		6/24/2021	

Unlocked vs. Locked Labels

New labels are automatically unlocked meaning they can be modified and assigned to Validatar objects such as tests, templates, jobs, and profile sets. Locked labels cannot be modified. If a label is locked after it is assigned to an object, the association to the object will remain and cannot be removed unless the label is again unlocked.

There are two ways to lock or unlock a label:

- By selecting the label and clicking the **Unlock** or **Lock** button on the label list screen.
- By opening a label and clicking the **Unlock** or **Lock** button.

The current status of a label is shown on the label list screen and in the top-right corner of the label's detail page.

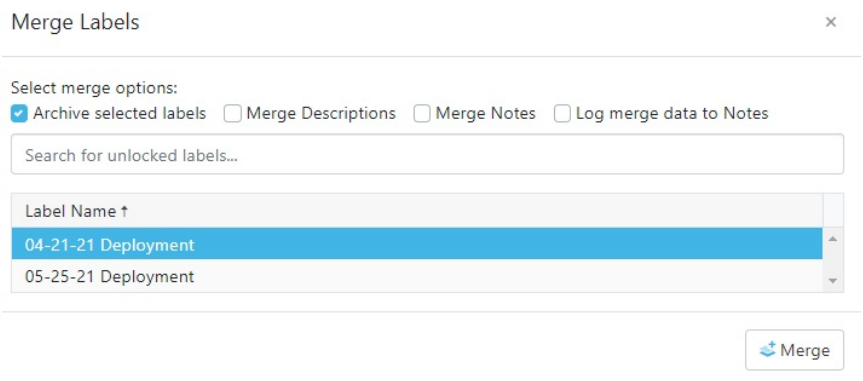
Archiving a Label

You can archive labels that you no longer want to use by selecting the **Archive** button on the label list screen. Archived labels can be viewed by checking the **Include archived** checkbox in the top-right hand corner of the screen.

To restore an archived label, select the label and click the **Restore** button.

Merging Unlocked Labels

You can copy the contents and object assignments of one or more unlocked labels to a single label by merging the labels.



Select merge options:
☑ Archive selected labels ☐ Merge Descriptions ☐ Merge Notes ☐ Log merge data to Notes

Search for unlocked labels...

Label Name ↑
04-21-21 Deployment
05-25-21 Deployment

To merge labels:

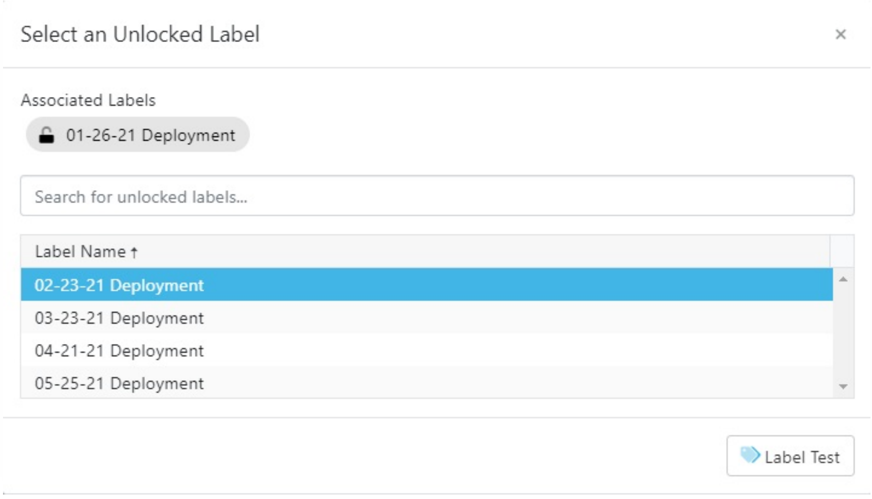
1. Create a new label or select an existing label that will be the current label.
2. On the label detail screen select **Merge**.
3. Search for and select the label(s) you want to copy the label contents of to the current label.
4. Select **Merge**.

You can merge labels that have the same objects assigned to them (e.g. both labels are associated with the same test). If you try to merge in a label that contains a different version of the object than the current label, the latest version of the object will be referenced.

Using Labels

Adding Labels to an Object

No matter if you're adding a label to a standard test, template test, or job, there's a label button in the upper-right corner of the configuration screen that will open a pop-up window to add the label(s).



Select an Unlocked Label ×

Associated Labels

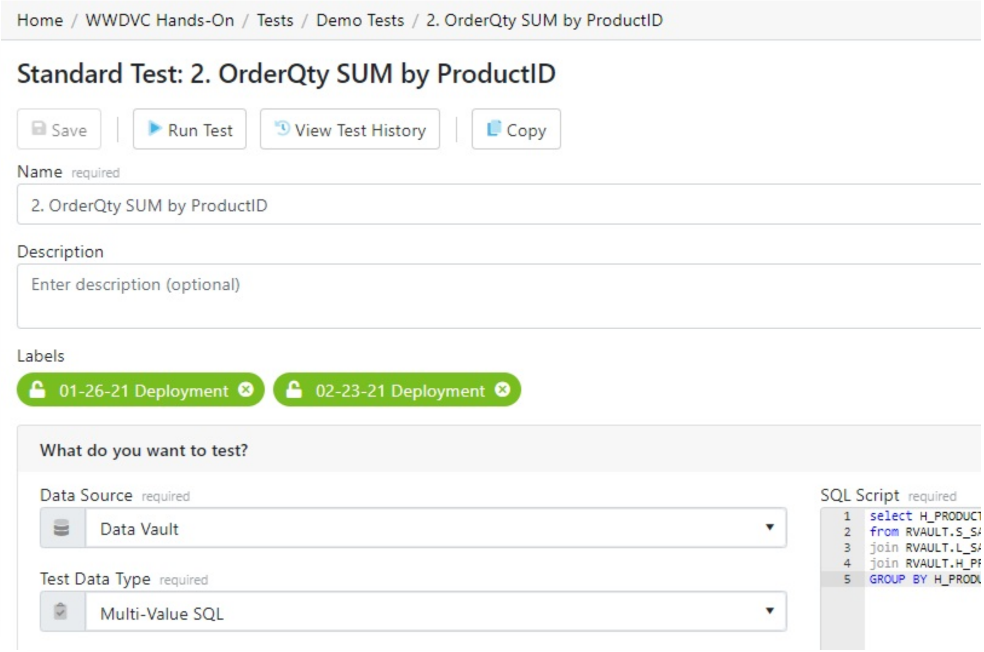
01-26-21 Deployment

Search for unlocked labels...

Label Name ↑
02-23-21 Deployment
03-23-21 Deployment
04-21-21 Deployment
05-25-21 Deployment

Once you click the Label button, the project's unlocked labels will appear in a list. Next, you can choose to add one or more labels to that object by selecting the row. Any label already associated with the object will be pinned at the top of the window and not appear in the list. After selecting the relevant labels, click the **Label Test** button.

After adding a label, a **Labels** area will appear under the Description box and list the associated labels. Click on a label to view the label details.



Home / WWDVC Hands-On / Tests / Demo Tests / 2. OrderQty SUM by ProductID

Standard Test: 2. OrderQty SUM by ProductID

Name required: 2. OrderQty SUM by ProductID

Description: Enter description (optional)

Labels: 01-26-21 Deployment, 02-23-21 Deployment

What do you want to test?

Data Source required: Data Vault

Test Data Type required: Multi-Value SQL

SQL Script required

```
select H_PRODUCT
from RVAULT.S_SA
join RVAULT.L_SA
join RVAULT.H_PR
GROUP BY H_PRODU
```

All unlocked labels are highlighted in green. If a label is locked after being added to an object, the label highlight becomes blue.

Removing Labels

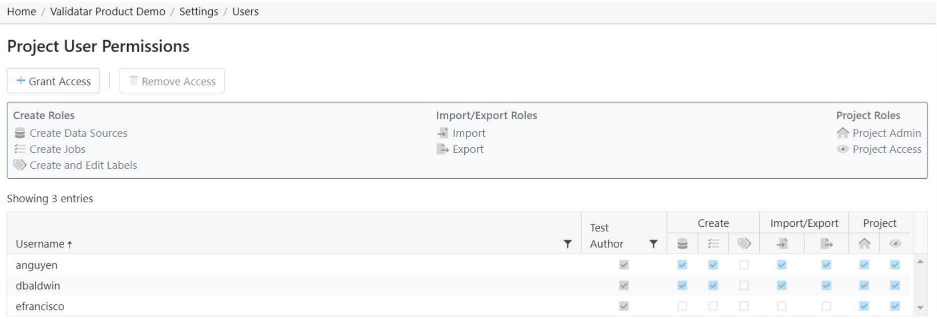
To remove a label from the object, navigate to the object detail page and click the x on the label. Note that only unlocked labels can be removed.

Learn about exporting Validatar objects by a label on the **Export** page.

Project Users

Overview

The **Project Settings Users page** lists Validatar users with access to the selected project.



Home / Validatar Product Demo / Settings / Users

Project User Permissions

Create Roles
- Create Data Sources
- Create Jobs
- Create and Edit Labels

Import/Export Roles
- Import
- Export

Project Roles
- Project Admin
- Project Access

Showing 3 entries

Username ↑	Test Author	Create			Import/Export		Project	
anguyen	☑	☑	☑	☐	☑	☑	☑	☑
dbaldwin	☑	☑	☑	☐	☑	☑	☑	☑
efrancisco	☑	☐	☐	☐	☐	☐	☑	☑

Granting Access

The **Grant User Access** page provides a list of users that do not have access to the project. The process for granting a user project access is as follows:

1. Within **Project Settings**, click **Users** and then click **Grant Access**.
2. Type the user's name to select them from the dropdown list.
3. Add the permissions by checking the appropriate boxes for each role.

Removing Access

1. Select the user from the list.
2. Click **Remove Access**.

You can restore a user's access immediately after removing it by clicking the **Restore Access** button. If you navigate away from the page, you will have to click the **Grant Access** button to re-add the user.

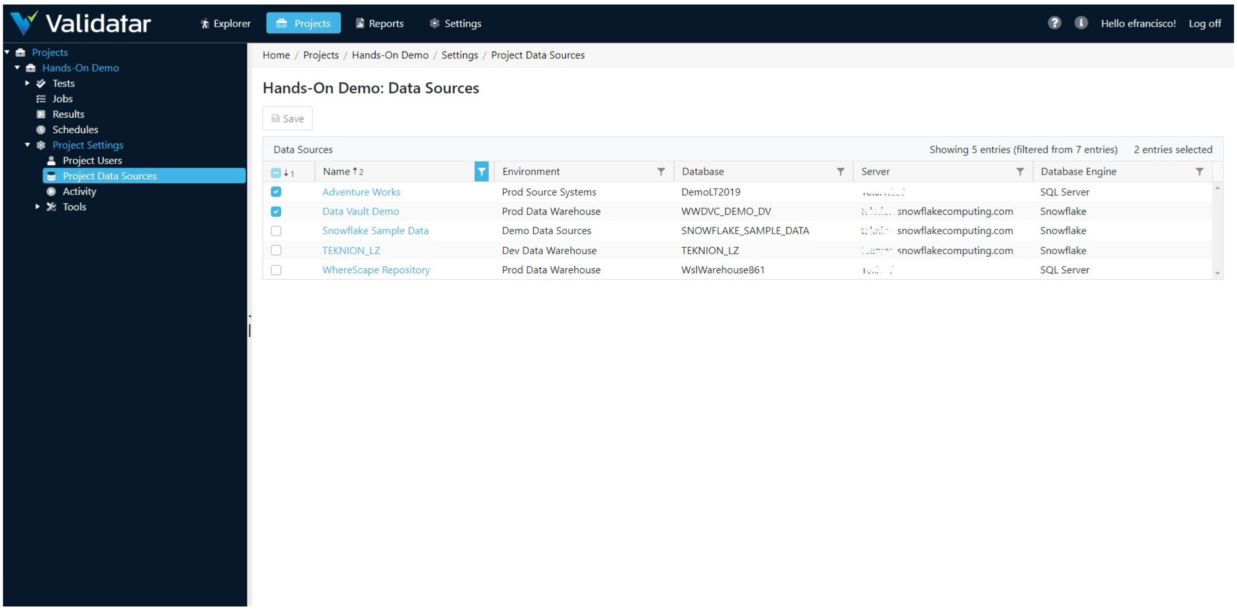
Roles

Create Roles	Import/Export Roles	Project Roles
• **Create Jobs** - the user can create Validatar jobs. • **Create and Edit Labels** - the user can create and manage labels found under Project Tools.	• **Import** - the user can import an XML file of Validatar objects into the project. • **Export** - the user can export Validatar objects out of a project to an XML file.	• **Project Admin** - the user is a project administrator. • **Project Access** - the user has read-access to the project.

Project Data Sources

Data Sources are managed and assigned to a project globally, but a project admin can also enable the use of a data source in project settings.

The Project Data Sources page shows a list of all available data sources. Check or uncheck the data sources you want to use in your project.



Home / Projects / Hands-On Demo / Settings / Project Data Sources

Hands-On Demo: Data Sources

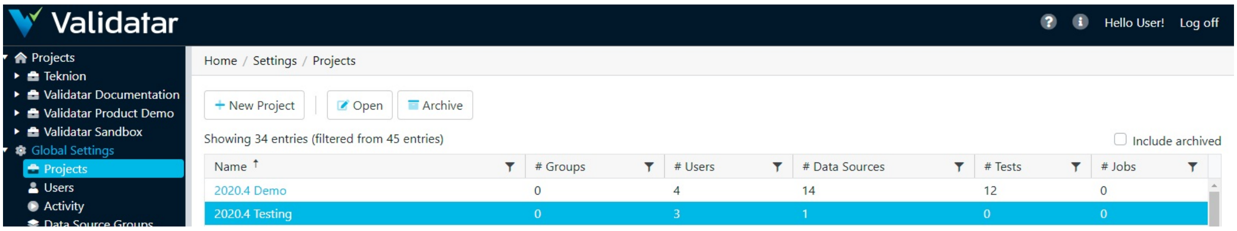
Data Sources — Showing 5 entries (filtered from 7 entries) — 2 entries selected

	Name	Environment	Database	Server	Database Engine
☑	Adventure Works	Prod Source Systems	DemoLT2019	[redacted]	SQL Server
☑	Data Vault Demo	Prod Data Warehouse	WWDVC_DEMO_DV	[redacted] snowflakecomputing.com	Snowflake
□	Snowflake Sample Data	Demo Data Sources	SNOWFLAKE_SAMPLE_DATA	[redacted] snowflakecomputing.com	Snowflake
□	TEKNION_LZ	Dev Data Warehouse	TEKNION_LZ	[redacted] snowflakecomputing.com	Snowflake
□	WhereScape Repository	Prod Data Warehouse	WslWarehouse861	[redacted]	SQL Server

Projects

Overview

Projects house tests, jobs, and project settings. The Global Settings Projects page is where you manage all of your projects in Validatar.



Showing 34 entries (filtered from 45 entries)

Name	# Groups	# Users	# Data Sources	# Tests	# Jobs
2020.4 Demo	0	4	14	12	0
2020.4 Testing	0	3	1	0	0

Action Buttons

New Project	Click to navigate to the New Project page.
Open Project	Open the currently highlighted project.
Archive	Validatar projects are not deleted. Rather, they are archived. Clicking the Archive button will archive the selected project. Archived projects can be restored.

Table Headers

Header	Description
Name	The name of the project.
# Users	The number of users associated with the project.
# Data Sources	The number of data sources assigned to the project.
# Tests	The number of tests associated with the project.
# Jobs	The number of jobs associated with the project.

Access

1. The Projects page is accessed under **Settings** in the navigation pane.

User Groups

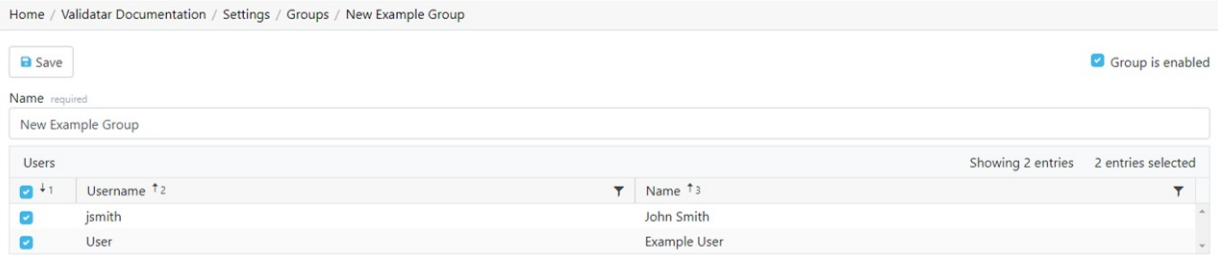
Upgrading to 2021.3

User groups were previously found in **Projects** but have moved to **Global Settings**. After upgrading, the user group list will show the user group's name followed by the name of the project to identify where the particular user group migrated from. You can now **Merge** user groups from multiple projects into one single group to avoid redundancy. You can also rename any migrated user groups to improve clarity.

Overview

Groups are used to organize users. We recommend using User Groups to assign permissions instead of individual users for easier and consistent management.

Creating Groups


Home / Validatar Documentation / Settings / Groups / New Example Group
Save
Group is enabled

☑ Group is enabled

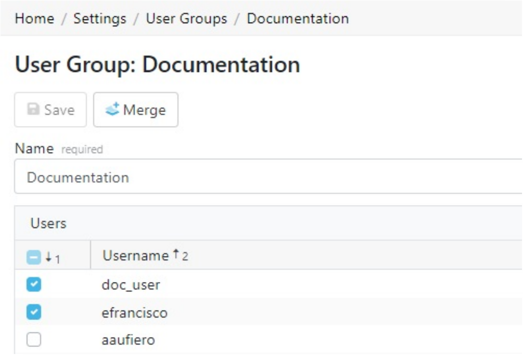
Name required: New Example Group

Users — Showing 2 entries — 2 entries selected

☑ ↓1	Username ↑2	Name ↑3
☑	jsmith	John Smith
☑	User	Example User

1. To create a new user group, select **New Group** in the toolbar.
2. To add users to a group, select the checkbox next to their username. Deselect the checkbox to remove users from a group.
3. Click **Save**.

Merging User Groups



User Group: Documentation

Name required: Documentation

Users

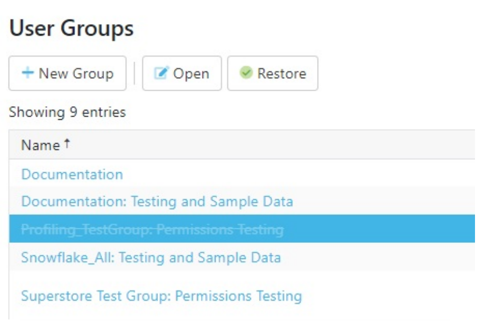
☐ ↓1	Username ↑2
☑	doc_user
☑	efrancisco
☐	aaufiero

Follow these steps to merge user groups:

1. Open the user group you want to survive.
2. Click on **Merge** in the toolbar.
3. Choose to either save or delete the user groups that you're merging into the current group.
4. Select one or more user groups from the list. (Hold down Ctrl to multi-select)
5. Click **Merge**.

Note: You cannot restore a user group after it is deleted during the merge.

Deleting User Groups



User Groups

Showing 9 entries

Name ↑
Documentation
Documentation: Testing and Sample Data
~~Profiling_TestGroup: Permissions Testing~~
Snowflake_All: Testing and Sample Data
Superstore Test Group: Permissions Testing

Follow these steps to delete user groups:

1. Click the whitespace next to the user group name to select the row.
2. Click **Delete**.

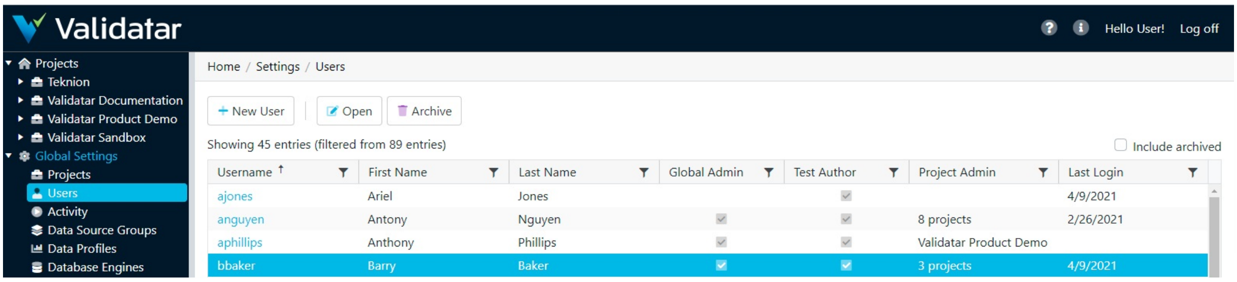
Note: You can restore the user group after it is deleted if you don't navigate away from the page. **Once you leave the User Groups page, the user group cannot be restored.**

Users

Overview

As the name implies, the Global Settings **Users** page is where you can manage every Validatar user across all projects.

Displayed Information



Showing 45 entries (filtered from 89 entries)

□ Include archived

Username	First Name	Last Name	Global Admin	Test Author	Project Admin	Last Login
ajones	Ariel	Jones		☑		4/9/2021
anguyen	Antony	Nguyen	☑	☑	8 projects	2/26/2021
aphillips	Anthony	Phillips	☑	☑	Validatar Product Demo	
bbaker	Barry	Baker	☑	☑	3 projects	4/9/2021

Action Buttons

Buttons	Description
New User	Navigate to the Global: User Details page to create a new user.
Open	Open the selected user's Global: User Details page.
Archive	Validatar users are not deleted. Rather, they are archived. Clicking the Archive button will archive the selected user. Archived users can be restored.

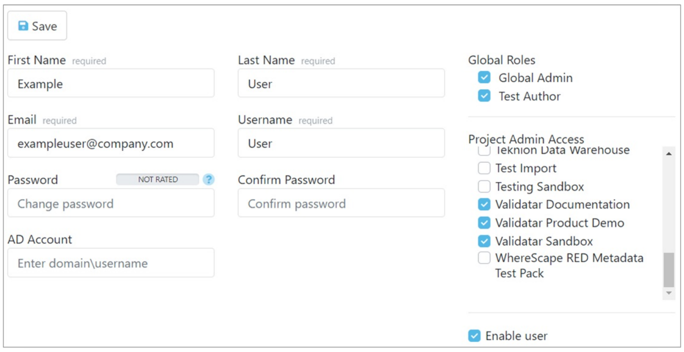
Table Headers

Header	Description
Include archived (checkbox)	Validatar users are not deleted, rather they are archived. Clicking the **Include archived** checkbox will cause those users to display.
Username	The username.
First Name	The user's first name.
Last Name	The user's last name.
Global Admin	Displays a checkmark if the user has the Global Admin role.
Test Author	Displays a checkmark if the user has the Test Author role.
Project Admin	Displays the project name where the user has the Project Admin role. If the user has Project Admin privileges for more than one project, the number of projects will display instead.
Last Login	The date the user last logged in.

Adding a New User

To add a new user, click the **New User** button on the User list page and enter all required user information.

Displayed Information



First Name required: Example	Last Name required: User	Global Roles ☑ Global Admin ☑ Test Author
Email required: exampleuser@company.com	Username required: User	Project Admin Access □ Teknion Data Warehouse □ Test Import □ Testing Sandbox ☑ Validatar Documentation ☑ Validatar Product Demo ☑ Validatar Sandbox □ WhereScape RED Metadata Test Pack
Password: Change password	Confirm Password: Confirm password	
AD Account: Enter domain\username		☑ Enable user

Save	Saves the new user.
First Name	The user's first name.
Last Name	The user's last name.
Email	The user's email address.
Username	The username.
Password; Confirm Password	Validatar requires a strong password. A strong password is defined by default as one that is at least 7 characters long and contains three of the following: • Upper-case characters (A-Z) • Lower-case characters (a-z) • Digits (0-9) **Note**: Password settings can be changed under **Global Settings** > **Configuration**.
AD Account	The AD textbox can be used to link Validatar with Active Directory.
Global Admin Roles	A global admin has the ability to create, configure, and manage settings for a Validatar instance. A full global admin inherits the following roles: • Configuration Admin: Can only see configuration settings. • Data Source Admin: Can only manage data sources and view Validatar server logs. • Security Admin: Can only manage users and user groups and view Validatar server logs. A user can have a combination of these roles without being a full global admin. There is not a limit on the number of global admins a Validatar instance can have.
Licensed Roles	There are two types of Validatar users: • Test Author: Can create new Validatar tests. • Viewer: Can view tests and test results for projects they have access to. All Validatar users are viewers. There is a maximum limit of users who can be test authors at one time based on the edition of Validatar that was purchased. Once that limit is reached, a user will have to lose the test author role for another user to gain it. Find your volume limits under your activated license key or in your Validatar Startup Guide emailed to you from the Validatar Customer Success team.
Project Admin Access	All active projects display under **Project Admin Access**. Selecting any project will grant the user admin access to that project. Admin access can also be granted via **Project** > **User Details**.

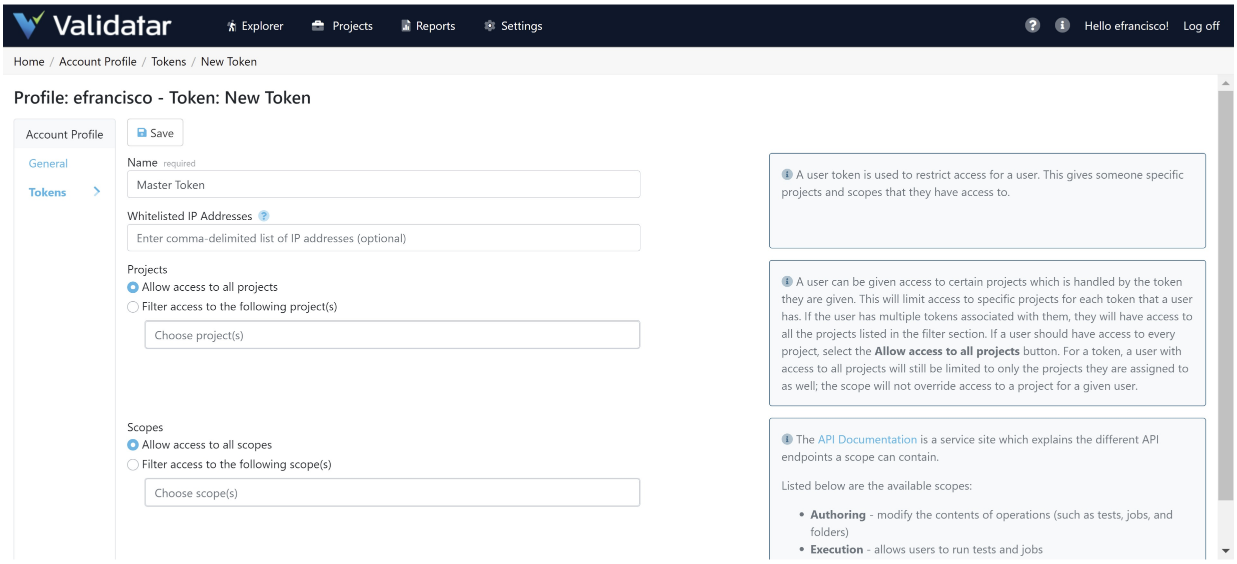
Access

1. Navigate to **Settings** on the main menu bar.
2. Click **Users**.

User Tokens

Overview

User tokens make API requests on behalf of a user. By creating a new user token, the user will be able to access specified projects and scopes. The token will encapsulate encrypted user security information.



Profile: efrancisco - Token: New Token

Name required: Master Token

Whitelisted IP Addresses: Enter comma-delimited list of IP addresses (optional)

Projects
● Allow access to all projects
○ Filter access to the following project(s)
Choose project(s)

Scopes
● Allow access to all scopes
○ Filter access to the following scope(s)
Choose scope(s)

A user token is used to restrict access for a user. This gives someone specific projects and scopes that they have access to.

A user can be given access to certain projects which is handled by the token they are given. This will limit access to specific projects for each token that a user has. If the user has multiple tokens associated with them, they will have access to all the projects listed in the filter section. If a user should have access to every project, select the **Allow access to all projects** button. For a token, a user with access to all projects will still be limited to only the projects they are assigned to as well; the scope will not override access to a project for a given user.

The API Documentation is a service site which explains the different API endpoints a scope can contain.

Listed below are the available scopes:

- **Authoring** - modify the contents of operations (such as tests, jobs, and folders)
- **Execution** - allows users to run tests and jobs

Creating User Tokens

Each token is always associated with a specific user account and will inherit the permissions of that user.

1. After logging into Validatar, click your username in the upper right-hand corner.
2. On the left pane, select **Tokens**.
3. Select **New Token**.
4. Enter a token name.
5. Enter a comma-delimited list of IP addresses where this token can have API access. (optional)
6. Specify the Projects this token can access.
    - Allow access to all projects
    - Filter access to the following project(s)
7. Specify the Scopes that are allowed.
    - Allow access to all scopes
    - Allow access to the following scope(s)
        - **Execution** - Allows users to run tests and jobs
        - **Security Admin** - Allows users to create and remove users
        - **Authoring** - Allows users to modify the contents of operations (such as tests, jobs, and folders)
8. Click **Save**.

   A masked API key will be generated after saving a user token. The API key can be unmasked, which is what you will need to connect to the API.

Managing User Tokens

A Validatar Security or Global admin can manage user tokens by navigating to **Settings** > **User Tokens**.

Managing user tokens uses the same interface as creating a new user token. An admin has the following abilities:

- To modify, disable, and delete tokens
- To view which projects the token has access to
- To view which scopes the token has access to
- Viewing the last accessed date
- View the age of the token

An admin cannot unmask a user token's API key. Only the user account the token is associated with can unmask the key.

Using Tokens to Access API

You can learn more about the API scopes using **Validatar's API documentation** on the user token configuration page.

The documentation includes information about the REST API endpoints each scope contains and examples of API responses.

Activity

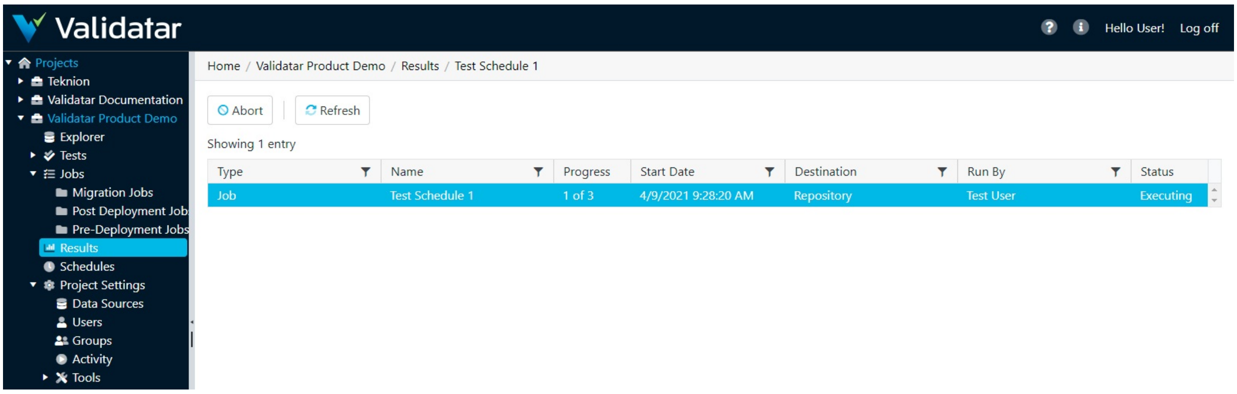
Overview

The **Activity** page lists all the jobs and tests that are currently running. There are two Activity areas in Validatar:

- Global Activity - shows activity amongst all projects in Validatar.
- Project Activity - shows activity in the project the user in currently in.

The purpose of the Activity page is to provide a comprehensive view of all activity that has taken place within Validatar. The list can be refreshed and filtered on any of the information elements to easily find activity for a particular project, user, destination, etc.

If a test or job is hung up, you can abort the execution of the test or job from the Activity page.



Home / Validatar Product Demo / Results / Test Schedule 1

Showing 1 entry

Type	Name	Progress	Start Date	Destination	Run By	Status
Job	Test Schedule 1	1 of 3	4/9/2021 9:28:20 AM	Repository	Test User	Executing

Configuring Lineage Metadata

Overview

Data lineage gives visibility into where data comes from and where it goes. As a data pro, having access to lineage allows for quicker root cause analysis and better visualization of your data process. Understanding data lineage is valuable for multiple reasons:

- Understanding the origin of data
- Quickly identifying sources of error
- Providing audit trails for better track data assets
- Improving business performance by improving business analysis
- Managing data governance compliance and much more!

Validatar provides a way to view a data source's table and column-level lineage in the Explorer. To view lineage, you must create a lineage ingestion process for a data source. Choose between File or SQL Query ingestion.

Consider where you plan to get the lineage data from when choosing an ingestion process. Use a SQL query if your lineage data is coming from one source, stored in a database. Use File ingestion if your data is coming from multiple sources or is not stored in a database.

Before creating and executing a lineage ingestion process, make sure you ingest schema metadata for the data source.

File Process

The File Ingestion process consists of 4 steps:

1. Create a New File Process
2. Generate the Import File
3. Fill out the spreadsheet
4. Upload the Import File

Create a New File Process

To create a new file process:

1. Navigate to **Settings** > **Lineage Metadata**.
2. Click the **New Process** drop-down, and select **New File Process**.
3. Enter the process name and optional description.
4. Click **Save**.

Generating a File

The lineage file holds the source to target mapping information for your data process. Generating the import file from Validatar provides all the column headers needed to ingest lineage. Existing file processes can create the import file from previous ingestion. This option is disabled for new file processes.

To generate the import file:

1. In the new file process, click **Generate Import File**.
2. Select **Headers Only** and **click Download File**.

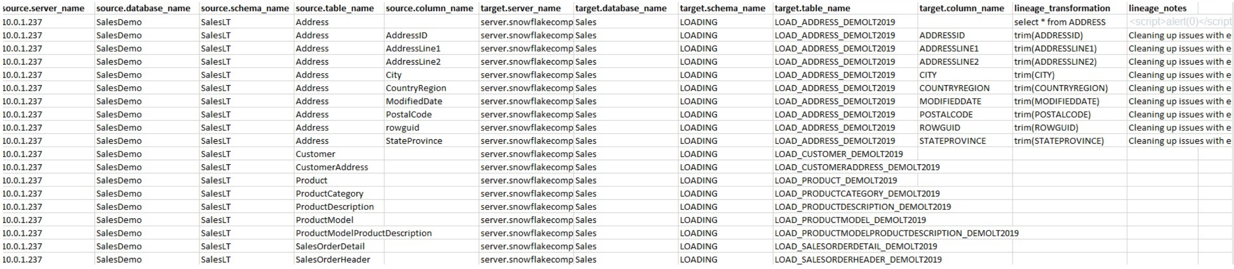
The generated import file has the following columns:

Source Information	Target Information	Lineage Notes
• source.server_name • source.database_name • source.schema_name • source.table_name • source.column_name	• target.server_name • target.database_name • target.schema_name • target.table_name • target.column_name	• lineage_transformation* • lineage_notes*

* not required

Each row represents a source to target data mapping in your data process down to the column level meaning each column in a table will have a record in the file. For table level mapping, you will leave the **source.column_name** and **target.column_name** fields blank.

Lineage_transformation will show the user what data transformation occurred on the target object. **Lineage_notes** will show any additional notes on the target object. Both can be viewed on the data object in the Explorer.



source.server_name	source.database_name	source.schema_name	source.table_name	source.column_name	target.server_name	target.database_name	target.schema_name	target.table_name	target.column_name	lineage_transformation	lineage_notes
10.0.1.237	SalesDemo	SalesLT	Address		server.snowflakecomp	Sales	LOADING	LOAD_ADDRESS_DEMOLT2019		select * from ADDRESS	<script>alert(0)</script
10.0.1.237	SalesDemo	SalesLT	Address	AddressID	server.snowflakecomp	Sales	LOADING	LOAD_ADDRESS_DEMOLT2019	ADDRESSID	trim(ADDRESSID)	Cleaning up issues with e
10.0.1.237	SalesDemo	SalesLT	Address	AddressLine1	server.snowflakecomp	Sales	LOADING	LOAD_ADDRESS_DEMOLT2019	ADDRESSLINE1	trim(ADDRESSLINE1)	Cleaning up issues with e
10.0.1.237	SalesDemo	SalesLT	Address	AddressLine2	server.snowflakecomp	Sales	LOADING	LOAD_ADDRESS_DEMOLT2019	ADDRESSLINE2	trim(ADDRESSLINE2)	Cleaning up issues with e
10.0.1.237	SalesDemo	SalesLT	Address	City	server.snowflakecomp	Sales	LOADING	LOAD_ADDRESS_DEMOLT2019	CITY	trim(CITY)	Cleaning up issues with e
10.0.1.237	SalesDemo	SalesLT	Address	CountryRegion	server.snowflakecomp	Sales	LOADING	LOAD_ADDRESS_DEMOLT2019	COUNTRYREGION	trim(COUNTRYREGION)	Cleaning up issues with e
10.0.1.237	SalesDemo	SalesLT	Address	ModifiedDate	server.snowflakecomp	Sales	LOADING	LOAD_ADDRESS_DEMOLT2019	MODIFIEDDATE	trim(MODIFIEDDATE)	Cleaning up issues with e
10.0.1.237	SalesDemo	SalesLT	Address	PostalCode	server.snowflakecomp	Sales	LOADING	LOAD_ADDRESS_DEMOLT2019	POSTALCODE	trim(POSTALCODE)	Cleaning up issues with e
10.0.1.237	SalesDemo	SalesLT	Address	rowguid	server.snowflakecomp	Sales	LOADING	LOAD_ADDRESS_DEMOLT2019	ROWGUID	trim(ROWGUID)	Cleaning up issues with e
10.0.1.237	SalesDemo	SalesLT	Address	StateProvince	server.snowflakecomp	Sales	LOADING	LOAD_ADDRESS_DEMOLT2019	STATEPROVINCE	trim(STATEPROVINCE)	Cleaning up issues with e
10.0.1.237	SalesDemo	SalesLT	Customer		server.snowflakecomp	Sales	LOADING	LOAD_CUSTOMER_DEMOLT2019			
10.0.1.237	SalesDemo	SalesLT	CustomerAddress		server.snowflakecomp	Sales	LOADING	LOAD_CUSTOMERADDRESS_DEMOLT2019			
10.0.1.237	SalesDemo	SalesLT	Product		server.snowflakecomp	Sales	LOADING	LOAD_PRODUCT_DEMOLT2019			
10.0.1.237	SalesDemo	SalesLT	ProductCategory		server.snowflakecomp	Sales	LOADING	LOAD_PRODUCTCATEGORY_DEMOLT2019			
10.0.1.237	SalesDemo	SalesLT	ProductDescription		server.snowflakecomp	Sales	LOADING	LOAD_PRODUCTDESCRIPTION_DEMOLT2019			
10.0.1.237	SalesDemo	SalesLT	ProductModel		server.snowflakecomp	Sales	LOADING	LOAD_PRODUCTMODEL_DEMOLT2019			
10.0.1.237	SalesDemo	SalesLT	ProductModelProductDescription		server.snowflakecomp	Sales	LOADING	LOAD_PRODUCTMODELPRODUCTDESCRIPTION_DEMOLT2019			
10.0.1.237	SalesDemo	SalesLT	SalesOrderDetail		server.snowflakecomp	Sales	LOADING	LOAD_SALESORDERDETAIL_DEMOLT2019			
10.0.1.237	SalesDemo	SalesLT	SalesOrderHeader		server.snowflakecomp	Sales	LOADING	LOAD_SALESORDERHEADER_DEMOLT2019			

The above is an example of a source database named "SalesDemo" located on server 10.0.1.237 mapped to a cloud database named "Sales" located in Snowflake. Notice that the first record leaves **source.column_name** and **target.column_name** blank because it is table level mapping of the "Address" table to the "LOAD_ADDRESS_DEMOLT2019" table. The remaining Address records are column-level mappings.

Once the mapping is complete, it's time to upload the file and execute the lineage process.

Uploading a File

To upload a file:

1. In the new file process, click **Upload Import File**.
2. Select the file.
3. Click **Import**.
4. Click **Save**.

Query Process

Create a New Query Process

To create a new query process:

1. Navigate to **Settings** > **Lineage Metadata**.
2. Click the **New Process** drop-down, and select **New Query Process**.
3. Enter the process name and optional description.
4. Add the **Lineage Metadata Connection**.
5. Configure the ingestion SQL.
6. Set the Refresh Schedule (optional).
7. Click **Save**.

Lineage Metadata Connection

The first step in configuring the query is to create the lineage metadata connection. You cannot use existing data source connections for this step. Instead, you will have to define the connection on the process configuration page.

For more help on configuring a connection, view Creating a Data Source.

Configure Ingestion SQL

The SQL query references the same columns as the file process. Validatar requires that all columns in the query be aliased using Validatar's field names. Copy the following query template to assist with building the query.

PL/SQL Copy

```
select
    [replace] as [source.server_name],
    [replace] as [source.database_name],
    [replace] as [source.schema_name],
    [replace] as [source.table_name],
    [replace] as [source.column_name],
    [replace] as [target.server_name],
    [replace] as [target.database_name],
    [replace] as [target.schema_name],
    [replace] as [target.table_name],
    [replace] as [target.column_name],
    [replace] as [lineage_transformation],
    [replace] as [lineage_notes]

from [replace]
```

Swap your column names for the [replace] tag, and remember to keep those columns aliased with Validatar's field names. You can also find this template by clicking **Copy Template Query** in the SQL ingestion pop-up window.

This is the simplest form of the ingestion query. In reality, your lineage data may come from multiple tables that need to be joined together. In this example, we'll look at a sample lineage query for a data warehouse built using WhereScape. This query will pull the lineage of Dimension tables using the WhereScape repository.

PL/SQL Copy

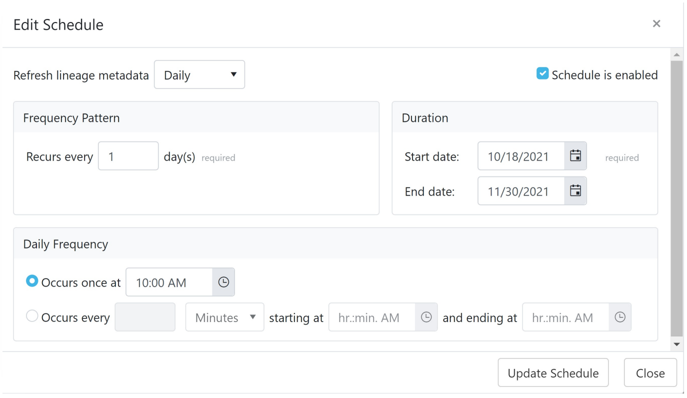
```
SELECT distinct
    '10.0.1.237'        as 'source.server_name',
 dc.dc_odbc_source  as 'source.database_name',
 sdt.dt_schema      as 'source.schema_name',
 dc_src_table       AS 'source.table_name',
    ''                  as 'source.column_name',
    '10.0.1.237'        as 'target.server_name',
 dc.dc_odbc_source  as 'target.database_name',
 dt.dt_schema       as 'target.schema_name',
 dt_table_name      AS 'target.table_name',
    ''                  as 'target.column_name',
    ''                  as 'lineage_transformation',
    ''                  as 'lineage_notes'
FROM dbo.ws_dim_tab d
INNER JOIN ws_dim_col on dc_obj_key = dt_obj_key
inner join ws_obj_object on oo_obj_key = dt_obj_key
inner join ws_dbc_target dt on dt_target_key = oo_target_key or case when oo_target_key = 0 then 'dbo' else
'' end = dt.dt_schema
inner join ws_dbc_connect dc on dt.dt_connect_key = dc.dc_obj_key

inner join ws_obj_object so on so.oo_name = dc_src_table
inner join ws_dbc_target sdt on sdt.dt_target_key = so.oo_target_key or case when so.oo_target_key = 0 then
'dbo' else '' end = sdt.dt_schema
inner join ws_dbc_connect sdc on sdt.dt_connect_key = dc.dc_obj_key

WHERE COALESCE(LTRIM(RTRIM(dc_src_table)),'') <> ''
```

This query finds the database, schema, and table names by joining the source and target objects on their respective object keys using WhereScape tables. Since this query is for table-level lineage, the **source.column_name** and **target.column_name** columns are left empty. You would union similar scripts to get the lineage for other object types.

Similar to the file ingestion, **lineage_transformation** and **lineage_notes** are not required in the ingestion query.

Set Refresh Schedule

Set a refresh schedule to ensure the most recent lineage data is in Validatar. Choose the frequency and duration of the schedule.



Edit Schedule

Refresh lineage metadata: Daily

☑ Schedule is enabled

Frequency Pattern

Recurs every 1 day(s) required

Duration

Start date: 10/18/2021 required

End date: 11/30/2021

Daily Frequency

● Occurs once at 10:00 AM

○ Occurs every Minutes starting at hr.:min. AM and ending at hr.:min. AM

Possible Errors

The following errors may occur while trying to ingest lineage metadata.

Error	Description
Target object was not found	There was no matching table or column found in the target data source.
Source object was not found	There was no matching table or column found in the source data source.
Duplicate Mapping	The file or query contains a duplicate mapping. Make sure that a source to target mapping only appears once.
Invalid Column	The file or query contains an unknown column. Remove any invalid columns.

Data Source Groups

Removal of Data Source Groups

The Data Source Groups feature has been removed in Validatar 2021.4. All previously existing Data Source Groups have been migrated to a Catalog Custom Field called Data Source Groups. The custom field can then be added to the applicable data sources.

Data Profiles

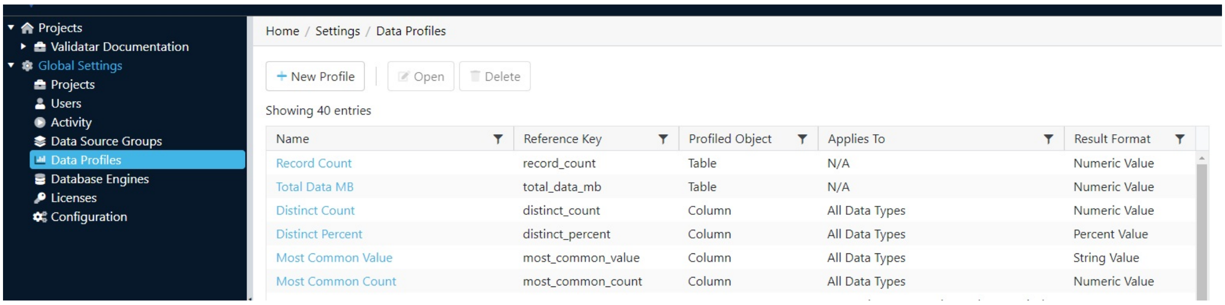
What are Data Profiles?

Data Profiles are information about existing data that help to determine the accuracy, completeness, and quality of your data. Data profiling is typically used within the broader context of ELT, monitoring, and Data Governance. When done properly, data profiling can play a significant part in data cleansing, enriching, and maintaining quality data within an organization.

Some common data profiles that are useful in many cases are:

- Record count
- Distinct value count
- Nulls
- Range of values
- Average values

Validatar contains a standard set of 40 data profiles. They are listed on the **Data Profiles** page with filterable columns. Any data profile on the list can be deleted by clicking on the white space next to the Name and then clicking the Delete button at the top of the page.



Showing 40 entries

Name	Reference Key	Profiled Object	Applies To	Result Format
Record Count	record_count	Table	N/A	Numeric Value
Total Data MB	total_data_mb	Table	N/A	Numeric Value
Distinct Count	distinct_count	Column	All Data Types	Numeric Value
Distinct Percent	distinct_percent	Column	All Data Types	Percent Value
Most Common Value	most_common_value	Column	All Data Types	String Value
Most Common Count	most_common_count	Column	All Data Types	Numeric Value

Creating a New Data Profile

When you create a new data profile, you're creating the existence of the profile. To configure the data profile definition, navigate to **Settings** > **Database Engines** > **Profiling** > **Choose the data profile**. This allows for different database engines to use the same data profile name, but to be configured differently.

1. Click **New Profile** on the Data Profiles page.
2. Configure the settings in Table 1.
3. This will bring up a new page with space to enter the Name (required), Reference Key (required), Description (optional), Profiled Object (required), and Results Format (required). Once the required fields are populated, the Save button will become clickable and you can save the new profile.

Table 1

Setting	Description
Name	The data profile name.
Reference Key	The data profile's unique key identifier.
Description (optional)	A description of the data profile.
Profile Object	Choose a table or column-level profile. • Table level - Evaluates the entire table. • Column level - Evaluates one column.
Result Format	The data type and format the result should be in. Options are: • Date/Time Array • Date/Time Value • Numeric Array • Numeric Value • Percent Value • String Array • String Value
Restrict data types (checkbox)	When checked, the data profile is only valid for the selected data types. The data type options are: • Approximate Numeric • Boolean/Logical • Date/Time • Exact Numeric • Geospatial • GUID • String • Other

Data Profile List

The complete list of the standard profiles pre-written in Validatar is:

Record Count	Lower Quartile	Null Percent	Maximum (Date)
Total Data MB	Upper Quartile	Blank Count	Longest Value
Distinct Count	Minimum (String)	Blank Percent	Shortest Value
Distinct Percent	Maximum (String)	Numeric Count	Distribution (String)
Most Common Value	Standard Deviation	Numeric Percent	Distribution (Numeric)
Most Common Count	Max Length	Zero Count	Top 10 Values
Minimum (Numeric)	Min Length	Zero Percent	Bottom 10 Values
Maximum (Numeric)	Mean Length	Negative Count	Binned (Numeric)
Mean (Numeric)	Length Distribution	Negative Percent	Year Distribution
Median (Numeric)	Null Count	Minimum (Date)	Year Month Distribution

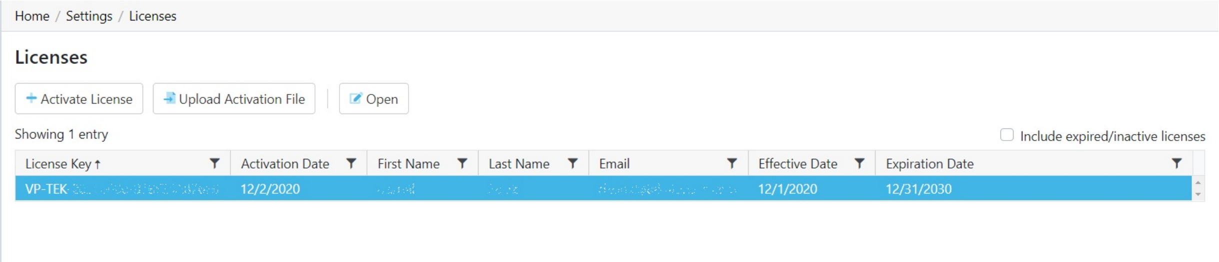
Data Profile Details

Each data profile name is clickable and when opened, displays a details page for that profile. The details include Name, Reference Key, Description, Profiled Object, and Results Format. There is also an optional Restrict Data Type checkbox and if checked, will display a list of options. Multiple options can be checked within the Data Type options list. The data profile details page can be updated and saved.

Licenses

Overview

A Validatar license contains information about the edition of Validatar being used, when the license expires, and the features included in the Validatar purchase.



Home / Settings / Licenses

Licenses

Showing 1 entry

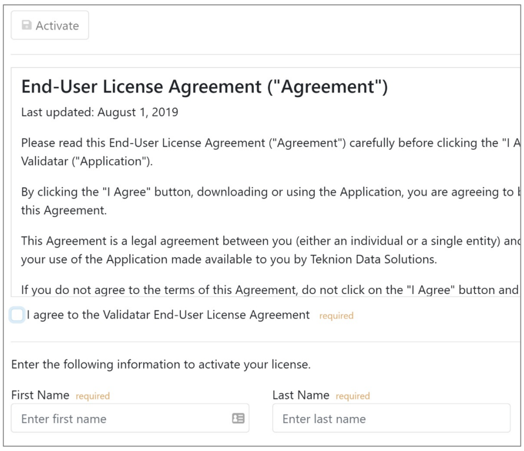
□ Include expired/inactive licenses

License Key ↑	Activation Date	First Name	Last Name	Email	Effective Date	Expiration Date
VP-TEK [redacted]	12/2/2020	[redacted]	[redacted]	[redacted]	12/1/2020	12/31/2030

Activating a License

To activate a license:

1. Navigate to **Global Settings > Licenses**.
2. Select **Activate License**.
3. Read the Validatar End-User License Agreement and select “I Agree”.
4. Fill out the license information form. The Validatar Customer Success Team would have sent your license to you.
5. Select **Activate**.



End-User License Agreement ("Agreement")

Last updated: August 1, 2019

Please read this End-User License Agreement ("Agreement") carefully before clicking the "I A
Validatar ("Application").

By clicking the "I Agree" button, downloading or using the Application, you are agreeing to b
this Agreement.

This Agreement is a legal agreement between you (either an individual or a single entity) and
your use of the Application made available to you by Teknion Data Solutions.

If you do not agree to the terms of this Agreement, do not click on the "I Agree" button and

□ I agree to the Validatar End-User License Agreement required

Enter the following information to activate your license.

First Name required: Enter first name

Last Name required: Enter last name

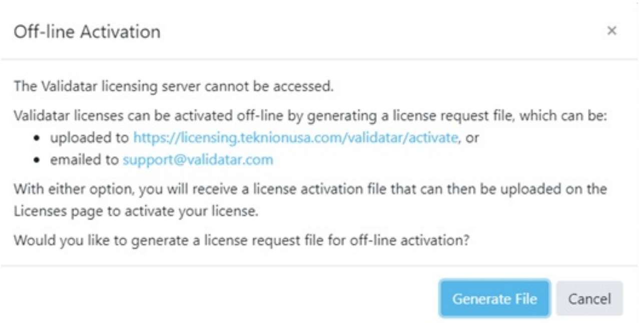
Activating a License without Internet

In instances where the server does not have internet connectivity, you can activate a license using the following steps:

- Generating a license request file.
- Send the license request file back to the Validatar team who will provide you with a license activation file in return.
- Upload the license activation file to Validatar.

To generate and upload a license file:

1. Complete the **Activating a License** steps listed above.
2. When you click **Activate** the following pop-up will appear:



Off-line Activation ×

The Validatar licensing server cannot be accessed.

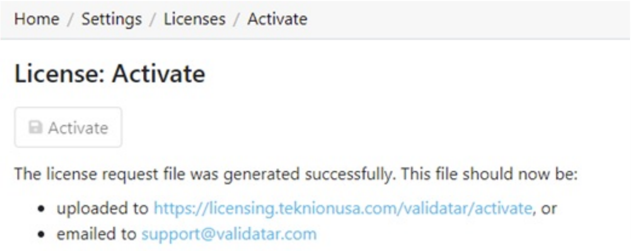
Validatar licenses can be activated off-line by generating a license request file, which can be:

- uploaded to https://licensing.teknionusa.com/validatar/activate, or
- emailed to support@validatar.com

With either option, you will receive a license activation file that can then be uploaded on the Licenses page to activate your license.

Would you like to generate a license request file for off-line activation?

3. Click the **Generate File** button to start the offline activation process.
   - This will create and download a file named Validatar-license-request-yyyyMMdd-hhnnss.json (with the date/time components in the filename replaced) and the page will change to the following:



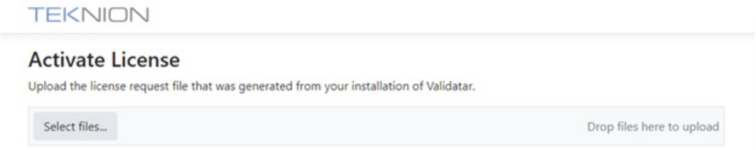
License: Activate

The license request file was generated successfully. This file should now be:

- uploaded to https://licensing.teknionusa.com/validatar/activate, or
- emailed to support@validatar.com

   - The generated JSON file contains the user information entered on the screen as well as a computed signature that guarantees the file was correctly generated from the client's instance of Validatar and was not modified after the file was created.

4. Upload the request file.
   - If you choose to upload the file via the activation link, you will see the following page:

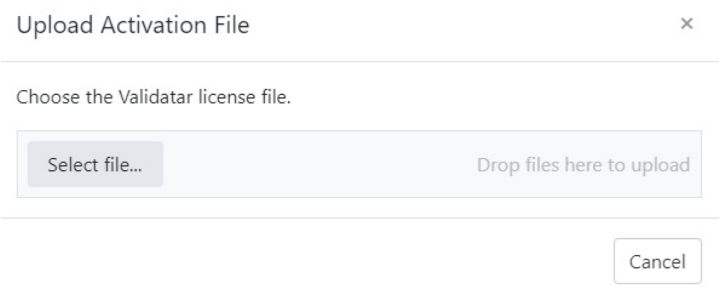


Activate License

Upload the license request file that was generated from your installation of Validatar.

     - Select the request file that was downloaded from Validatar.
   - If you choose to send the Validatar team the request file directly, we will complete **Step 4** for you and send you the **activation file**.

5. Navigate to the **Licenses** page and click **Upload Activation File** that shows the following pop-up:



Upload Activation File ×

Choose the Validatar license file.

6. Select the activation file that was either downloaded to your computer or sent to you by the Validatar team.
7. You should then get a message that says "The license file was activated successfully".

Common Errors

When uploading the generated request file, you may receive one of the following errors:

- "The uploaded file contained an invalid key."
- "The uploaded file was not a valid license request."
- "The uploaded file did not contain a valid signature."
- "The license key in the uploaded file has already been activated."

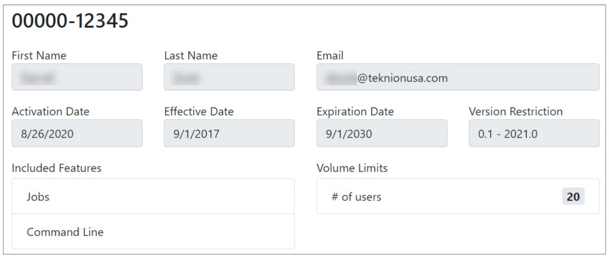
- "The uploaded license request was not processed successfully."

When uploading the activation file, you may receive a message that says "The uploaded license file was not valid."

In both cases, contact the Validatar team for further support.

License Details

The **License Details** page displays specifics about a given license.



00000-12345

First Name	Last Name	Email	
[redacted]	[redacted]	[redacted]@teknionusa.com	

Activation Date	Effective Date	Expiration Date	Version Restriction
8/26/2020	9/1/2017	9/1/2030	0.1 - 2021.0

Included Features	Volume Limits	
Jobs	# of users	20
Command Line		

- **Version Restriction** - The versions for which this license is applicable.
- **Included Features -** Included features may change depending on the release and license.
  - **Jobs**: Jobs and related features are included; for example, since jobs are available, this instance of Validatar supports scheduling.
  - **Command Line**: Command line access is available.
  - **Template Tests**: Users can create multiple dynamic tests at once using the Template Test functionality.
  - **Custom Profiles**: Users can create custom profiles to go along with the 40 standard profiles Validatar ships with.
- **Volume Limits**: The number of users that can be enabled at one time.

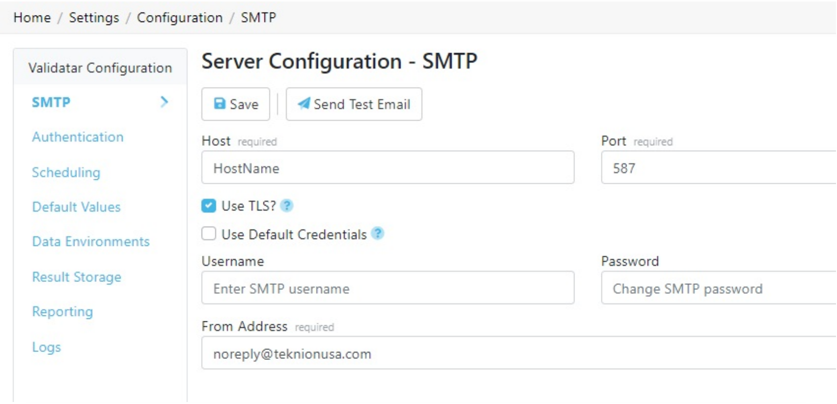
Access

1. Find all license keys by navigating to **Global Settings** > **Licenses** in the navigation pane.

SMTP Settings

Overview

Simple Mail Transfer Protocol (SMTP) is a communication protocol used for email transmission. SMTP must be configured in order to receive job email notifications and to reset passwords. Contact your organization's IT or Network Administrator for SMTP details.



Server Configuration - SMTP

Host required: HostName

Port required: 587

☑ Use TLS?

□ Use Default Credentials

Username: Enter SMTP username

Password: Change SMTP password

From Address required: noreply@teknionusa.com

The settings that must be configured are listed below.

Setting	Description
Save	Save any changes made to the SMTP configuration settings.
Send Test Email	Sends a test email to your user's email address to verify SMTP settings.
Host	The server name for the SMTP server.
Port	The port that the mail server expects communications to come through. The port should typically be set to one of the following values: • 25: clear text SMTP port • 587: encrypted SMTP using TLS
User TLS?	Check this option to encrypt SMTP traffic using the Transport Layer Security (TLS) protocol.
Use Default Credentials	Check this option to authenticate using the service account Validatar runs under instead of entering specific credentials.
Username (optional)	SMTP username
Password (optional)	Changes SMTP password
From Address	The displayed sender's address.

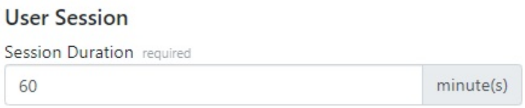
Authentication

Overview

The **Authentication** section contains settings related to profile security, such as automatic logouts and account passwords.

User Session

- **Session Duration:** Determines how long, in minutes, a user can remain idle in Validatar before they are automatically logged out.



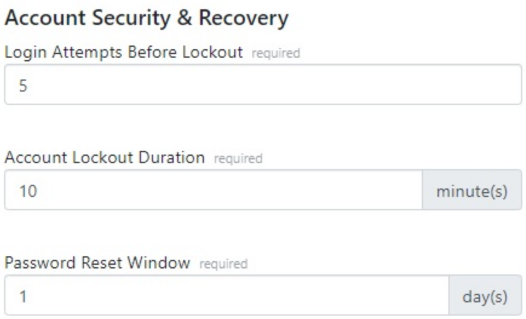
User Session

Session Duration required

60 minute(s)

Account Security & Recovery

- **Login Attempts Before Lockout:** The number of incorrect password attempts allowed before locking a user account.
- **Account Lockout Duration:** How long, in minutes, an account is locked after the maximum number of failed login attempts is reached.
- **Password Reset Window:** The length of time, in days, a user has to click the Password Reset email link before it expires.



Account Security & Recovery

Login Attempts Before Lockout required

5

Account Lockout Duration required

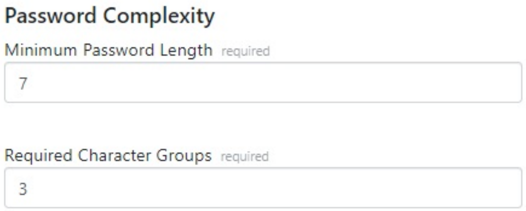
10 minute(s)

Password Reset Window required

1 day(s)

Password Complexity

- **Minimum Password Length:** The minimum character length of all passwords.
- **Required Character Groups:** The minimum number of character groups that are required for an acceptable password. Character groups are necessary to enforce password complexity.
    - Set to 1, then only upper-case characters are required.
    - Set to 2, then upper-case and lower-case characters are required.
    - Set to 3, then upper-case characters, lower-case characters, and digits are required.
    - Set to 4, then upper-case characters, lower-case characters, digits, and punctuation characters are required.



Password Complexity

Minimum Password Length required

7

Required Character Groups required

3

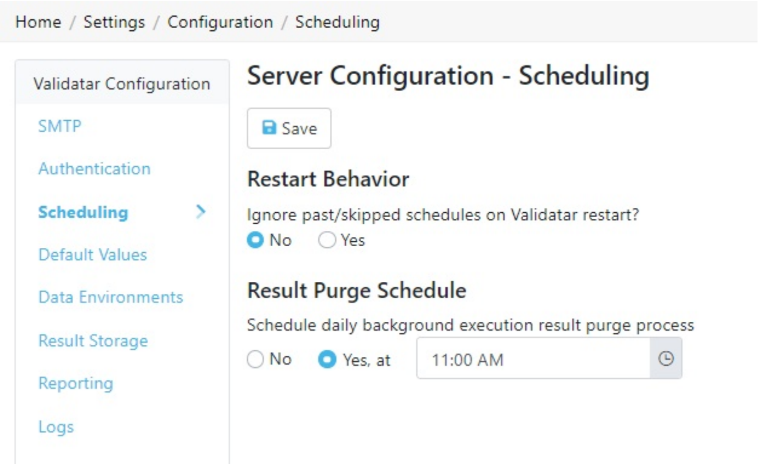
Scheduling

Overview

The **Scheduling** section allows the user to

- Choose to ignore past or skipped schedules when the Validatar site is restarted in IIS (Internet Information Services) and
- Choose to schedule a daily result purge. The result purge clears the Validatar database of test results but keeps the history of test executions.

If the Result Purge Schedule is set to yes, Validatar will purge the results of any tests or profile sets that have been stored longer than the defined number of days either on the individual test, profile set, or default setting at the specified time.



Home / Settings / Configuration / Scheduling

Server Configuration - Scheduling

Restart Behavior

Ignore past/skipped schedules on Validatar restart?

● No ○ Yes

Result Purge Schedule

Schedule daily background execution result purge process

○ No ● Yes, at 11:00 AM

Default Values

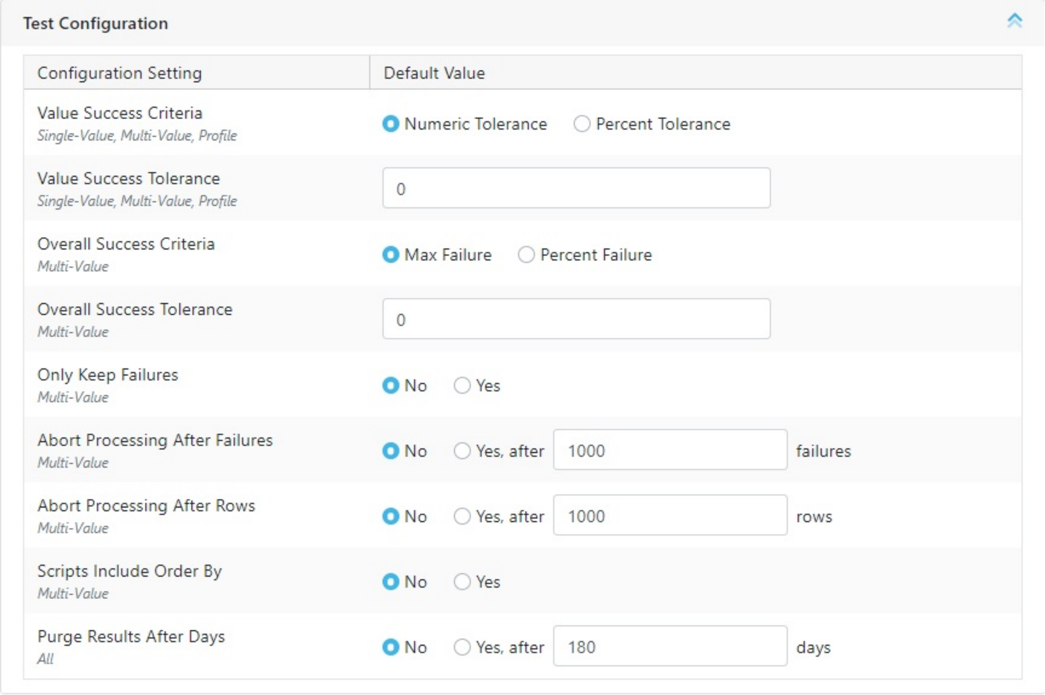
Overview

The **Default Values** section allows you to select test and profile set configuration settings on a server level. This allows every new test and profile set created to be standardized.

> **Configuration Note**
>
> The configuration setting on an individual test or profile set overrides the default setting.

Test Configuration

The test configuration settings reflect the result configuration section found at the bottom of standard and template tests.



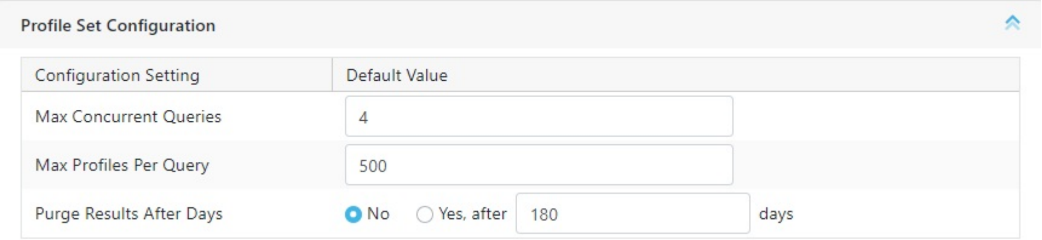
Test Configuration

Configuration Setting	Default Value
Value Success Criteria *Single-Value, Multi-Value, Profile*	● Numeric Tolerance ○ Percent Tolerance
Value Success Tolerance *Single-Value, Multi-Value, Profile*	0
Overall Success Criteria *Multi-Value*	● Max Failure ○ Percent Failure
Overall Success Tolerance *Multi-Value*	0
Only Keep Failures *Multi-Value*	● No ○ Yes
Abort Processing After Failures *Multi-Value*	● No ○ Yes, after 1000 failures
Abort Processing After Rows *Multi-Value*	● No ○ Yes, after 1000 rows
Scripts Include Order By *Multi-Value*	● No ○ Yes
Purge Results After Days *All*	● No ○ Yes, after 180 days

The test data type that applies to each setting is italicized under the setting name.

Profile Set Configuration

The profile set settings reflect what's on the Profile Set Configuration tab inside of a profile set.



Profile Set Configuration

Configuration Setting	Default Value
Max Concurrent Queries	4
Max Profiles Per Query	500
Purge Results After Days	● No ○ Yes, after 180 days

Notifications

New Feature!

You can now be notified about activity happening Validatar.

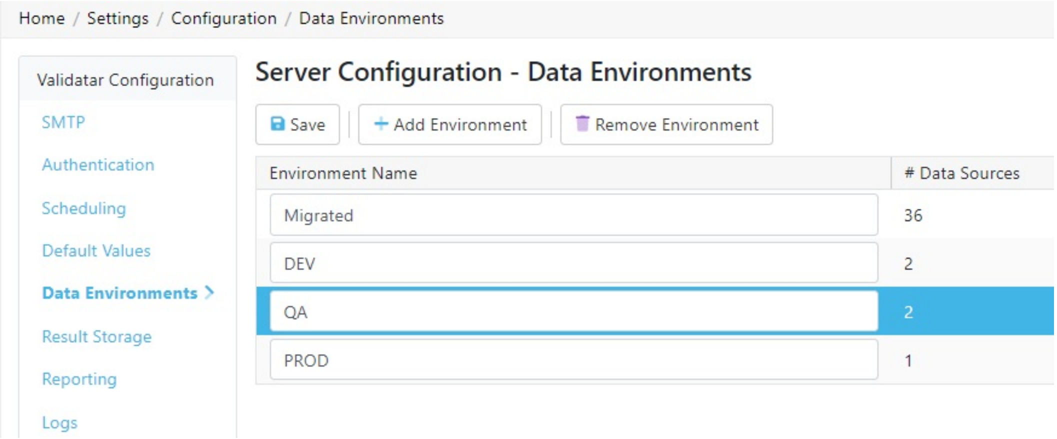
Overview

Preferences

Data Environments

Overview

Data Environments help to define a data source's usage relating to development and deployments.



Server Configuration - Data Environments

Environment Name	# Data Sources
Migrated	36
DEV	2
QA	2
PROD	1

Standard environment names include:

- Development or DEV
- Test or QA
- Production or PROD
- UAT

Usage of these environments helps separate architectures for deployments and are used in development. Adding an environment to your data source allows you to use the same name for different database connections but differentiate them by attaching the development environment.

Adding an Environment

1. Navigate to **Settings** > **Configuration** > **Data Environments**.
2. Click **Add Environment**.
3. Enter environment name.
4. Add as many environments as needed.
5. Click **Save**.

Removing an Environment

1. Navigate to **Settings** > **Configuration** > **Data Environments**.
2. Select the environment from the list.
3. Click **Remove Environment**.

Assigning an Environment to a Data Source

1. A **Data Source Admin** or **Global Admin** must navigate to **Settings** > **Data Sources** > **Select the Data Source**.
2. On the General tab, select an Environment from the dropdown.
3. Click **Save**.

You can find the total number of data sources assigned to an Environment on the Data Environments list page.

Migrated Environment

If you're upgrading from an older version to 2021.3, all data sources will inherit the **Migrated** environment automatically. You can change the name after the upgrade is complete.

Result Storage

Overview

During installation, a SQL Server database server is set up, which stores the results of test and profile set executions locally. If you are interested in storing results in the cloud, we offer the option to store result records in Snowflake.

Choosing a Location

- **Local Storage:** Test Results and Profile Results will be stored in the Validatar repository.
- **Snowflake:** Test Results and Profile Results will be stored in Snowflake. This option is not available until the Snowflake configuration has been completed and verified below.

Configuring Snowflake

Prerequisites

1. SnowSQL needs to be installed on the Validatar web server before changing the storage location to Snowflake.
2. A Snowflake admin must run the generated setup script in Snowflake to initialize the environment.

> **SnowSQL Note**
>
> Files are transferred to Snowflake using SnowSQL. SnowSQL version 1.2.16 or higher is required.

Setting up SnowSQL on Windows

Click here to view Snowflake's documentation on installing SnowSQL.

Note: The installer may fail to initialize log. If this occurs,

- Go to: `%USERPROFILE%\.snowsql\`
- Update the config file by replacing this line: `log_file = ../snowsql_rt.log`
- With this line: `log_file = %USERPROFILE%\.snowsql\log\snowsql_rt.log`

Configure the Connection

All parameters are required to configure the connection.

Connection Parameters	Description
Account Name	The account name is found in the URL used to log into Snowflake, excluding "*https://*" or "*.snowflakecomputing.com*". The account name must include the region.
Database Name	The database that will store the results. The database does not have to exist beforehand.
Schema Name	The schema that will store the results. The schema does not have to exist beforehand.
Role Name	The assigned user role used to. The role cannot be one of the following Snowflake system-defined roles: • ACCOUNTADMIN • SECURITYADMIN • USERADMIN • SYSADMIN • PUBLIC
Username	
Warehouse Name	The warehouse that will store the results. The warehouse does not have to exist beforehand.
Temporary File Path	Temporary files will be created in this folder before being imported into Snowflake.

Running the Setup Scripts

Complete the following steps to configure Snowflake.

1. After configuring the Snowflake Connection in Step 2, click **Generate Setup Script**.
2. Copy the setup script to the clipboard.
3. Paste the setup script into an open Snowflake worksheet.
4. Setup a password for the user by substituting [[REPLACE]] in the setup script with the actual password.
5. Run the setup script in Snowflake.
6. In Validatar, generate the public/private keys that will authenticate SnowSQL.
7. Click View Public Key/Script and copy the script to the clipboard.
8. Paste the key script into an open Snowflake worksheet
9. Run the key script.

You're done with the Snowflake setup! Next is validating the connection.

Public/Private Key Note

You can regenerate the public/private keys as frequently as you want, but you will need to regenerate and rerun the script to assign the new public key to the Validatar user account in Snowflake.

Validate the Connection

The last step to set up Snowflake is validating the connection. Click **Validate Settings**, and Validatar will complete the following steps:

1. Verifying that the Snowflake account exists
2. Verifying that SnowSQL is installed
3. Verifying the Snowflake login
4. Verifying the database, role, and warehouse
5. Creating the schema, table, stages, and pipes
6. Verifying access to the temporary file path
7. Verifying that a sample file can be uploaded/imported to Snowflake

Global Search

Overview

Easily find tests, jobs, and metadata objects using the search bar in the menu header. Not only can you search by object name, but you can search by Data Source, the SQL script used in a test, who created a test, and more.

Prerequisites

There are a few steps to complete before Global Search can be used.

1. Install & Configure Apache Solr
2. Configure Global Search settings in Validatar

Search Bar Display

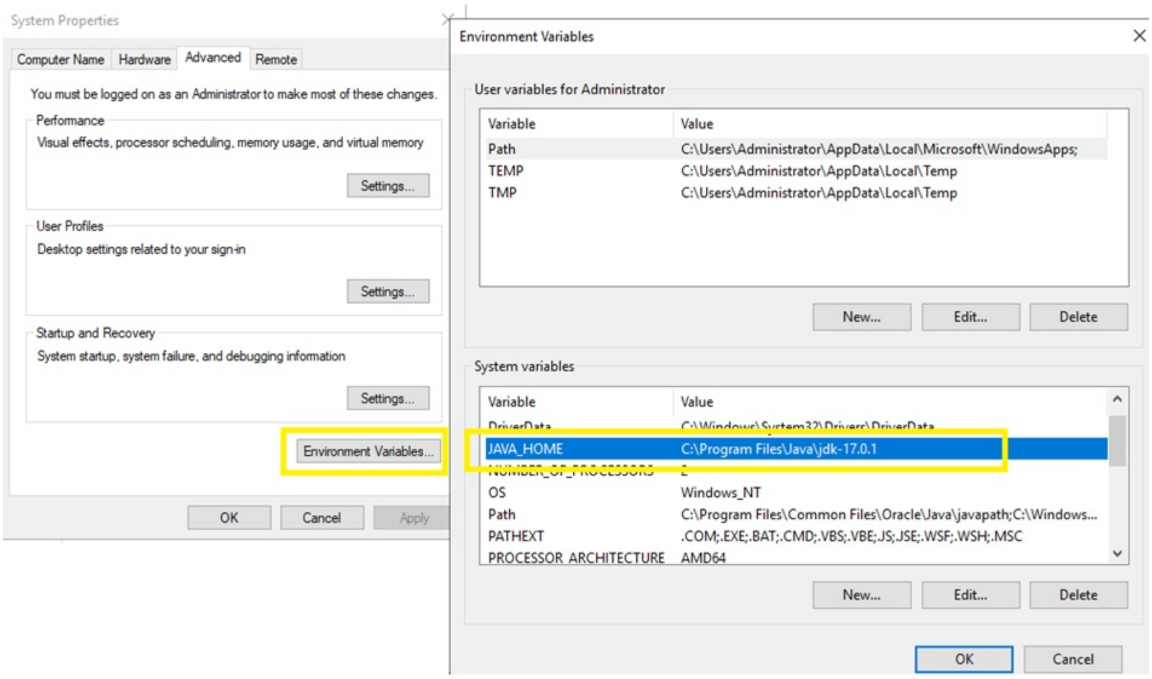
The search bar will not appear in the Validatar header if Global Search isn't configured.

Solr Install for Windows

1. Install Java

If not already installed,

1. Download Java: https://download.oracle.com/java/17/latest/jdk-17_windows-x64_bin.msi
2. Install the downloaded MSI
3. Create a JAVA_HOME system environment variable that references the root Java folder from the above install.



You must be logged on as an Administrator to make most of these changes.

Visual effects, processor scheduling, memory usage, and virtual memory

Desktop settings related to your sign-in

System startup, system failure, and debugging information

User variables for Administrator

Variable	Value
Path	C:\Users\Administrator\AppData\Local\Microsoft\WindowsApps;
TEMP	C:\Users\Administrator\AppData\Local\Temp
TMP	C:\Users\Administrator\AppData\Local\Temp

System variables

Variable	Value
DriverData	C:\Windows\System32\Drivers\DriverData
JAVA_HOME	C:\Program Files\Java\jdk-17.0.1
NUMBER_OF_PROCESSORS	2
OS	Windows_NT
Path	C:\Program Files\Common Files\Oracle\Java\javapath;C:\Windows...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
PROCESSOR_ARCHITECTURE	AMD64

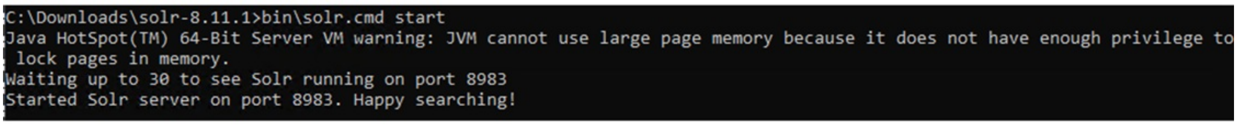
2. Install Solr

1. Download Solr from here: https://solr.apache.org/downloads.html. Use the link for the latest binary release zip file (https://www.apache.org/dyn/closer.lua/lucene/solr/8.11.1/solr-8.11.1.zip?action=download)
2. Unzip the Solr file
3. Open a command line prompt, navigate to the folder where you unzipped Solr, and start Solr using the following command:

None

Copy

```
bin\solr.cmd start
```

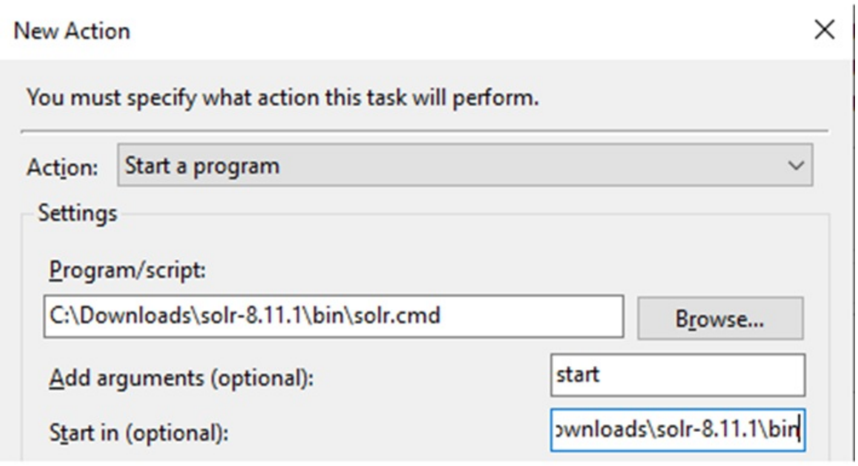


```
C:\Downloads\solr-8.11.1>bin\solr.cmd start
Java HotSpot(TM) 64-Bit Server VM warning: JVM cannot use large page memory because it does not have enough privilege to
 lock pages in memory.
Waiting up to 30 to see Solr running on port 8983
Started Solr server on port 8983. Happy searching!
```

4. At this point, you should be able to open a browser on that server and navigate to http://localhost:8983 to open the Solr UI

Server Restart Note

This command will need to be run any time the server restarts, so you'll probably want to create a task using Windows Task Scheduler with a trigger of "At startup".



New Action

You must specify what action this task will perform.

Action: Start a program

Settings

Program/script: C:\Downloads\solr-8.11.1\bin\solr.cmd

Add arguments (optional): start

Start in (optional): ›wnloads\solr-8.11.1\bin

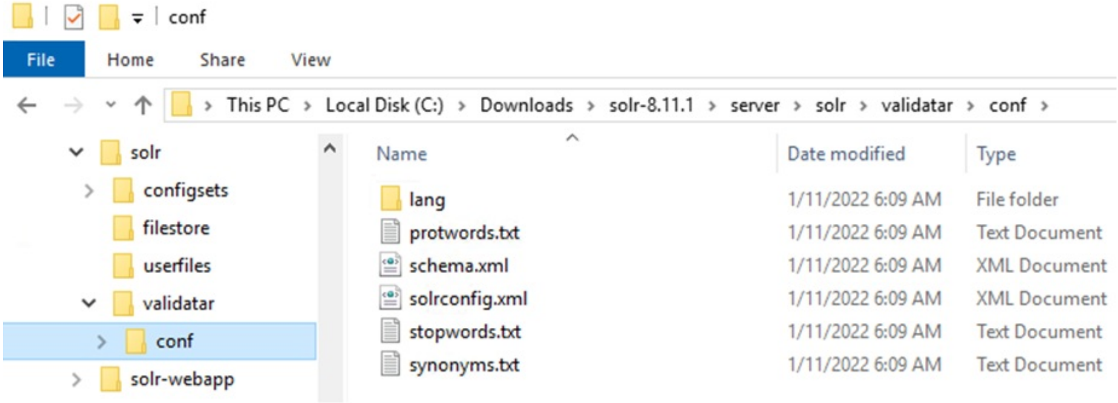
3. Add Validatar core to Solr

1. A "core" in Solr is where the data indexed by Validatar resides. If you're using one instance of Solr for multiple Validatar instances (dev and prod), you'll want to use a different core for each instance. Alternatively, if you're installing Solr on both ANB-DALSQL03 and ANB-DALSQL04, then you'll just need one core on each server that will be used by the Validatar instance on that server.

2. Create a folder for the new core inside the Solr root folder. On my server the root folder is here: C:\Downloads\solr-8.11.1\server\solr. You can use whatever name you want – I used "validatar", so if you use something else, replace any reference to that name in this step and the following steps.

3. Download the following file: https://admin.validatar.com/assets/releases/2022-2/solr-root.zip

4. Copy the "conf" folder from the zip file into the "validatar" folder. The folder should look like this:



conf
File
Home
Share
View
This PC › Local Disk (C:) › Downloads › solr-8.11.1 › server › solr › validatar › conf ›
solr
configsets
filestore
userfiles
validatar
conf
solr-webapp

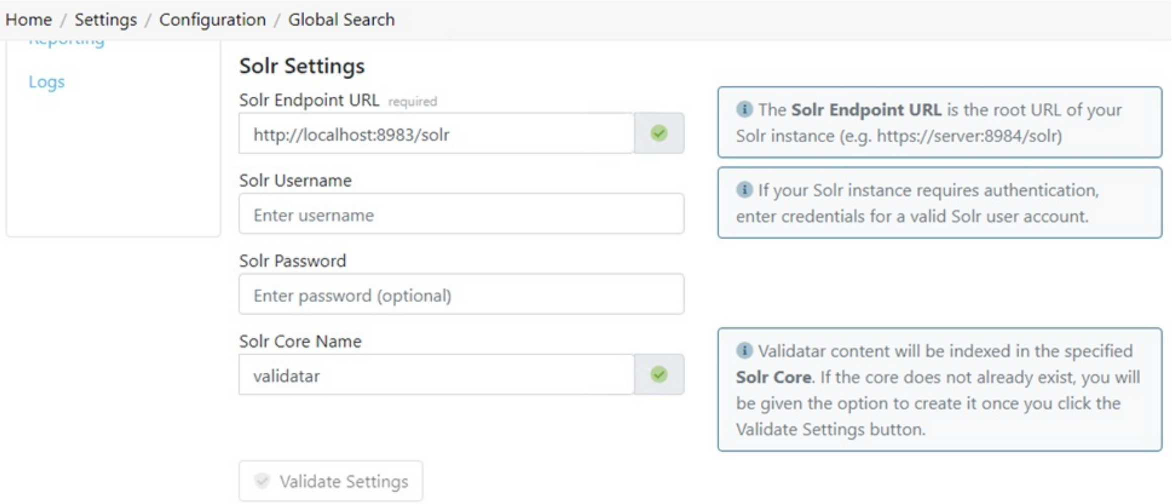
Name	Date modified	Type
lang	1/11/2022 6:09 AM	File folder
protwords.txt	1/11/2022 6:09 AM	Text Document
schema.xml	1/11/2022 6:09 AM	XML Document
solrconfig.xml	1/11/2022 6:09 AM	XML Document
stopwords.txt	1/11/2022 6:09 AM	Text Document
synonyms.txt	1/11/2022 6:09 AM	Text Document

5. Open a browser on the server and navigate to: http://localhost:8983/solr/#/~cores

6. Enter “validatar” in the name and instanceDir text boxes, then click the “Add Core” button

Configure Global Search in Validatar

1. Open a browser and navigate to Validatar
2. Go to **Settings** > **Configuration** > **Global Search**
3. Enter **http://localhost:8983/solr** in the Solr Endpoint URL textbox and “validatar” in the Solr Core Name textbox. Leave the username/password fields empty.
4. Click the **Validate Settings** button to verify that everything is configured correctly. If everything is valid, you’ll see two green checkboxes



Home / Settings / Configuration / Global Search

Reporting

Logs

Solr Settings

Solr Endpoint URL required

http://localhost:8983/solr ✓

ⓘ The **Solr Endpoint URL** is the root URL of your Solr instance (e.g. https://server:8984/solr)

Solr Username

Enter username

ⓘ If your Solr instance requires authentication, enter credentials for a valid Solr user account.

Solr Password

Enter password (optional)

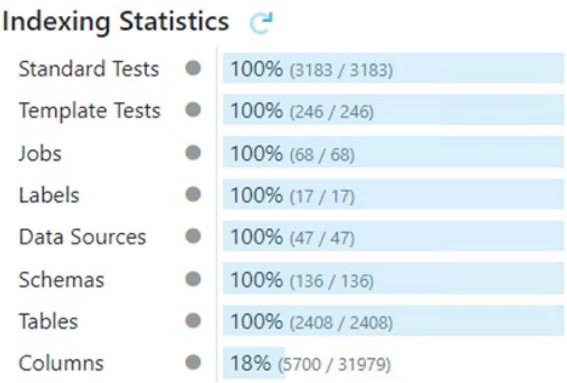
Solr Core Name

validatar ✓

ⓘ Validatar content will be indexed in the specified **Solr Core**. If the core does not already exist, you will be given the option to create it once you click the Validate Settings button.

Validate Settings

5. Scroll up and check the **Enable Indexing** box, then click the **Save** button at the top.
6. After a few seconds, Validatar will start indexing data. You can see the indexing progress by clicking the refresh button to the right of the Indexing Statistics heading.



Indexing Statistics

Standard Tests	●	100% (3183 / 3183)
Template Tests	●	100% (246 / 246)
Jobs	●	100% (68 / 68)
Labels	●	100% (17 / 17)
Data Sources	●	100% (47 / 47)
Schemas	●	100% (136 / 136)
Tables	●	100% (2408 / 2408)
Columns	●	18% (5700 / 31979)

Indexing Statistics Note

The indexed objects include archived and disabled objects in the Validatar Database.

Reporting

Overview

You can embed Tableau Server dashboards in Validatar by configuring the **Reporting** section.

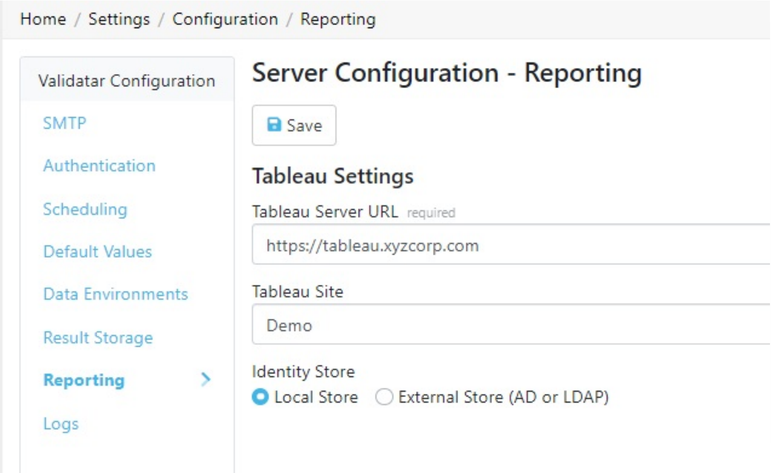


Tableau Server URL required

https://tableau.xyzcorp.com

Tableau Site

Demo

Identity Store

● Local Store ○ External Store (AD or LDAP)

Here's what you need to complete the setup:

- **Tableau Server URL:** The URL to your Tableau Server account. Example (https://tableau.xyzcorp.com)
- **Tableau Site (optional):** Specifies a site on a Tableau Server instance. Sites are used to separate groups of users and content on the same Tableau Server instance.
- **Identity Store:** This is chosen when Tableau Server is installed.
  - Local Store: All user and group information is stored and managed in the Tableau Server repository.
  - External Store (AD or LDAP): An external directory service stores and manages user and group information.

Logs

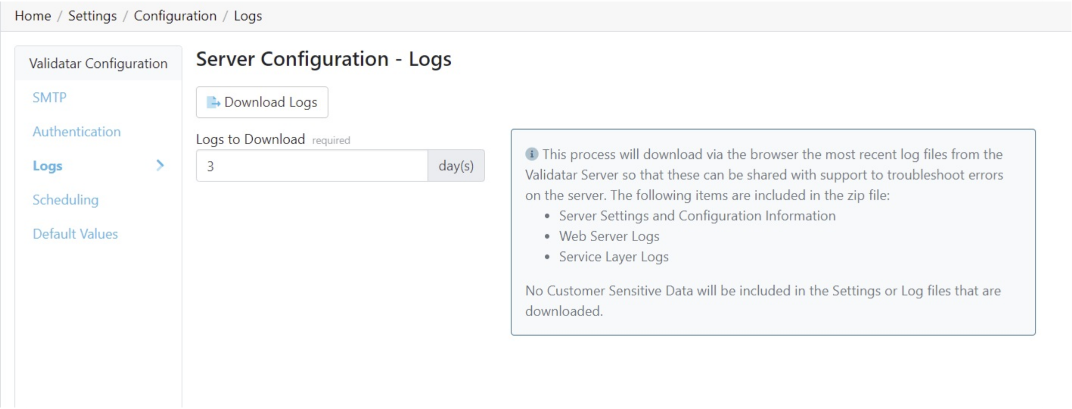
Overview

The **Logs** section allows the user to download recent log files that can be shared with support to troubleshoot errors on the server. Enter how far back (in days) you want the log file to go in the text box, then select Download Logs.

The following items are included in the zip file:

- Server Settings and Configuration Information
- Web Server Logs
- Service Layer Logs

No customer-sensitive data will be included in the settings or log files that are downloaded.



Logs to Download required
3 day(s)

This process will download via the browser the most recent log files from the Validatar Server so that these can be shared with support to troubleshoot errors on the server. The following items are included in the zip file:

- Server Settings and Configuration Information
- Web Server Logs
- Service Layer Logs

No Customer Sensitive Data will be included in the Settings or Log files that are downloaded.

Data Quality Monitoring Project

Overview

Are you connecting to Snowflake data sources? Use the Data Quality Monitoring (DQM) quick start project to jumpstart your QA testing. Oftentimes, the most challenging part of data testing is knowing where to start. The DQM project aims to give you standard monitoring, formatting, and easy-to-use tests straight out of the box. To aid in the standardization of data quality testing, it is important to utilize and continuously monitor data profiling stats over time.

You'll find several types of tests in the DQM project:

- **Comprehensive Monitoring** will detect any changes in data profiles such as record counts, duplicate values, null values, and more
- You can set Expectations for data values and be notified if that value falls out of range or hasn't been updated.
- Field Format

How to Use the Quick Start Project

1. Choose which Data Source to create the project from
2. Using the Explorer, add DQ Monitoring custom field values

Quick start projects are used to test one data source.

Comprehensive Monitoring

Profiling your data not only presents the current state of your data but helps you to monitor changes over time. Tests in the Comprehensive Monitoring folder are standard table and column-level profiles that indicate deviations from your typical state of data quality.

List of Column Tests

Test Name	Description
Blanks Showing Up in a Previously Non-Blank Column	
Column Average Changed more than X Percent	
Column Max Changed more than X Percent	
Distinct Count Changed more than X Percent	
Duplicates Showing Up in Previously Unique Column	
Negative Showing Up in Previously Non-Negative Column	
Nulls Showing Up in Previously Non-Null Column	
Zero Showing Up in Previously Non-Zero Column	

Custom Fields: Object Inclusion

The values you input for each data quality monitoring custom field drives how each test is configured. The custom fields are used to create the test name, filter which metadata objects the test can be run on, and the value success criteria for the test.

Schema Level

Choose how to enable DQM tests for the chosen schema. Tests included can be

- All Table and Column Tests

- All Table Level Tests Only
- All Column Level Tests Only
- Only Specific Tables and Columns
- None

Expectations

Custom Fields: Object Expectations

Commonly Used Regex

Overview

Regular Expressions also referred to as Regex, are a sequence of characters that attempts to identify patterns. Regex is commonly used to perform validations such as email or phone number validation in a set of data.

Regex Rules

There are a few categories of characters, operators, or constructs that define regular expressions. Provided are the categories referenced in the .Net regex engine. Use the following categories to build your regular expression.

Character Classes

A character class matches any one of a set of characters.

Character Class	Description	Example Pattern	Matches
[*character_group*]	A case-sensitive match of any single character in *character_group*.	[ah]	"a" in "ball" "a", "h" in "watch"
[^*character_group*]	Matches any single character that is **not** in *character_group.* Case-sensitive.	[^ba]	"o", "r", "d" in "board"
[*first* - *last*]	Matches any single character in the range from *first* to *last*.	[A-Z]	"A", "B" in "ABC123"
.	Wildcard that matches any single character except \n (new line). To match a literal period, escape the character (\.).	b.e	"bje" in "object" "ble" in "noble"
\p{*name*}	Matches any single character in the Unicode general category or named block specified by *name.*	\p{Lu}	"S", "A" in "Swiss Alps"
\P{name}	Matches any single character that is not in the Unicode general category or named block specified by *name*.	\P{Lu}	"l", "p", "s" in "Alps"
\w	Matches any word character.	\w	"A", "B", "C", "2", "5" in "ABC 2.5"
\W	Matches any non-word character.	\W	" ", "." in "ABC 2.5"
\s	Matches any white-space character.	\w\s	"C " in "ABC 2.5"
\S	Matches any non-white-space character.	\s\S	" 2" in "ABC 2.5"
\d	Matches any decimal digit.	\d	"3" in "3 toys"
\D	Matches any character other than a decimal digit.	\D	" ", "t", "o", "y", "s" in "3 toys"

Quantifiers

A quantifier specifies how many instances of a previous element (character, group, or character class) must be present in the string for a match.

Quantifier	Description	Example Pattern	Matches
*	Matches the previous element 0 or more times.	\d*\.\d	".0", "15.5", "415.5"
+	Matches the previous element 1 or more times.	"fe+"	"fe" in "fell" "fee" in "feel"
?	Matches the previous element 0 or 1 time.	"rai?n"	"ran", "rain"
{*n*}	Matches the previous element exactly *n* times.	",\d{3}"	",056" in "1,056.4"",435", ",692", ",718" in "9,435,692,718.00"
{*n*,}	Matches the previous element at least *n* times.	"\d{2,}"	"29", "435", "2031"
{*n*,*m*}	Matches the previous element at least *n* times but no more than *m* times.	"\d{3,5}"	"789", "5347" "90875" in "908756"
*?	Matches the previous element 0 or more times, but as few times as possible.	\d*?\.\d	".0", "15.5", "415.5"
+?	Matches the previous element 1 or more times, but as few times as possible.	"\fe+?"	"fe" in "feel" "fe" in "fell"
??	Matches the previous element 0 or 1 time, but as few times as possible.	"rai??n"	"ran","rain"
{*n*}?	Matches the previous element exactly *n* times.	",\d{3}?"	",056" in "1,056.4" ",435", ",692", ",718" in "9,435,692,718.00"
{*n*,}?	Matches the previous element at least *n* times, but as few times as possible.	"\d{2,}?"	"29", "435", "2031"
{*n*,*m*}?	Matches the previous element between *n* and *m* times, but as few times as possible.	"\d{3,5}?"	"435", "2031" "908", "756" in "908756"

Anchors

Anchors cause a match to pass or fail depending on the current position in the string.

Anchor	Description	Example Pattern	Matches
^	The match must start at the beginning of the string, or the beginning of the line.	^\d{3}	"913" in "913-333-"
$	The match must occur at the end of the string or before the new line at the end of the string. Or it must occur before the end of the line or before the new line at the end of the line.	-\d{3}$	"333" in "-913-333"
\A	The match must occur at the start of the string.	\A\d{3}	"913" in "913-333-"
\Z	The match must occur at the end of the string or before a new line at the end of the string.	-\d{3}\Z	"-333" in "-913-333"
\z	The match must occur at the end of the string.	-\d{3}\z	"-333" in "-913-333"
\G	The match must occur at the point where the previous match ended.	\G\(\d\)	"(1)", "(3)", "(5)" in "(1)(3)(5)[7](9)"
\b	The match must occur on a boundary between a \w (alphanumeric) and a \W (non-alphanumeric) character.	\bscheme\b	"scheme" in "That was a clever scheme"
\B	The match must not occur on a \b boundary.	\Band\w*\b	"ands", "ander" in "and sands android lander"

Escaped Characters

Escaped Character	Description
\a	Matches a bell character
\t	Matches a tab
\r	Matches a carriage return
\v	Matches a vertical tab
\f	Matches a form feed
\n	Matches a new line
\e	Matches an escape
\	When followed by a character that's not an escaped character,'\' is the literal match of the character.

Alternation

Alternation modifies a regular expression to enable either/or matching.

Alternation	Description
a|*b*	Matches either *a* or *b*
(?(*exp*)*yes*|*no*)	*yes* if *exp* is matched; *no* if *exp* isn't matched
(?(*name*)*yes*|*no*)	*yes* if *name* is matched; *no* if *name* isn't matched

9 Commonly Used Regex

Copy and paste the following regular expressions into Validatar to search for patterns in your data. Learn more about using regex with Validatar here.

1. Digits

- **Whole Numbers:** `^\d+$`
- **Decimal Numbers:** `^\d*\.\d+$`
- **Whole + Decimal Numbers:** `^[\-+]?\d*(\.\d+)?$`
- **Negative, Positive Whole + Decimal Numbers:** `^-?\d*(\.\d+)?$`
- **Whole + Decimal + Fractions:** `[-]?[0-9]+[,.]?[0-9]*([\/][0-9]+[,.]?[0-9]*)*`

2. Alphanumeric Characters

- **Alphanumeric without space:** `^[a-zA-Z0-9]+$`
- **Alphanumeric with space:** `^[a-zA-Z0-9 ]*$`

3. Email

- **Common Email Address:** `^([a-zA-Z0-9._%-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,6})*$`

4. Dates

- **Format YYYY-MM-dd:** `^(19|20)\d\d([- /.])(0[1-9]|1[012])\2(0[1-9]|[12][0-9]|3[01])$`
- **Format MM-dd-YYYY:** `^(0[1-9]|1[012])[- /.](0[1-9]|[12][0-9]|3[01])[- /.](19|20)\d\d$`
- **Format dd-MM-YYYY:** `^(0[1-9]|[12][0-9]|3[01])[- /.](0[1-9]|1[012])[- /.](19|20)\d\d$`

5. Time

- **Format HH:MM 12-hour, optional leading 0:** `^(0?[1-9]|1[0-2]):[0-5][0-9]$`
- **Format HH:MM 12-hour, optional leading 0, AM/PM:** `^((1[0-2]|0?[1-9]):([0-5][0-9]) ?([AaPp][Mm]))$`
- **HH:MM 24-hour with leading 0:** `^(0[0-9]|1[0-9]|2[0-3]):[0-5][0-9]$`
- **HH:MM 24-hour, optional leading 0:** `^([0-9]|0[0-9]|1[0-9]|2[0-3]):[0-5][0-9]$`
- **HH:MM:SS 24-hour:** `^(?:[01]\d|2[0123]):(?:[012345]\d):(?:[012345]\d)$`

6. Match Duplicates

- **Search Duplicates:** `(\b\w+\b)(?=.*\b\1\b)`

7. Phone Numbers

- **US and International Phone Numbers with Separators:** `^(\+\d{1,2}\s)?\(?\d{3}\)?[\s.-]\d{3}[\s.-]\d{4}$`
- **US Phone Numbers Only with Separators:** `^(\+0?1\s)?\(?\d{3}\)?[\s.-]\d{3}[\s.-]\d{4}$`
- **Unformatted Phone Numbers:** `^(\+\d{1,2}\s)?\(?\d{3}\)?[\s.-]?\d{3}[\s.-]?\d{4}$`

8. Identification

- **Social Security Number with dashes:** `^(?!219-09-9999|078-05-1120)(?!666|000|9\d{2})\d{3}-(?!00)\d{2}-(?!0{4})\d{4}$`
- **Social Security Number without dashes:** `^(?!219099999|078051120)(?!666|000|9\d{2})\d{3}(?!00)\d{2}(?!0{4})\d{4}$`

9. Zip Codes

- **US Postal Code:** `^\d{5}([\-]?\d{4})?$`

Connection Strings

Setup Note

The parameters of a connection string can change depending on the version of the database engine or driver. Reference your database provider's documentation for the correct connection string syntax if you do not see it listed here.

Overview

Connecting a data source to Validatar is done using connection strings. It specifies information about the data source and the means of connecting to it.

The connection string you use depends on the database engine you're connecting to, the chosen Validatar connection type, and whether or not you decide to use Windows Authentication.

Password Note

Always encrypt your password using the placeholder **[[PASSWORD]]** in the connection string and typing your password in the provided password box.

Validatar uses three different connection types: SQL Server, ODBC, and OLE DB.

Connecting using SQL Server

Use the SQL Server connection type to connect to SQL Server and Azure SQL databases. **Note:** Windows comes with the SQL Server driver already installed.

To escape any special characters (semicolons, single-quote character, double-quote character):

- The value must be enclosed in double quotation marks.
- If the value contains both a semicolon and double-quote character, the value must be enclosed in single quotation marks.

SQL Server Authentication

```sql
Server=myServerAddress;Database=myDatabase;User Id=myUsername;Password=[[PASSWORD]];
```

Windows Authentication

Integrated Security specifies that you are using Windows Authentication. User credentials are not required. When using Integrated Security, acceptable values are 'true', 'false', 'yes', or 'no'.

```sql
Server=myServerAddress;Database=myDatabase;Integrated Security=yes;
```

Windows Authentication Note

Validatar uses the service (domain) account Validatar is running under for Windows Authentication **not the current user's domain credentials**. Check with your web server admin and database admin to verify that the Validatar service account has read-access to the databases you want to connect to via Windows Authentication.

Connecting to a SQL Server Instance

If there are multiple instances on the server, you must identify the instance name in the connection string.

SQL Copy

```
Server=myServerName\myInstanceName;Database=myDatabase;User Id=myUsername;Password=[[PASSWORD]];
```

Specifying a Port

You can specify the port number after the server name.

SQL Copy

```
Server=myServerName,portNumber;Database=myDatabase;User Id=myUsername;Password=[[PASSWORD]];
```

Synonyms

Some connection string keywords are interchangeable with other synonyms.

Keyword	Synonym(s)
Server	Data Source, Address, Addr, Network Address
Database	Initial Catalog
Integrated Security	Trusted_Connection
Password	PWD
User ID	UID

Connecting using ODBC

Prerequisite Required

The appropriate **ODBC driver** must be installed on the server before connecting to a data source. Be sure to use the correct driver name in the connection string.

Netezza

SQL Copy

```
Driver=
{NetezzaSQL};Server=myServerAddress;port=myPortNumber;Database=myDataBase;username=myUsername;password=
[[PASSWORD]];
```

Snowflake

SQL Copy

```
Driver=
{SnowflakeDSIIDriver};Server=myServerAddress;Warehouse=myWarehouse;Database=myDatabase;Uid=myUserName;Pwd=
[[PASSWORD]];
```

Optional parameters

- Schema
- Role

Syntax Note

Make sure there are no spaces between the parameter value and semicolon in the connection string, or the connection will fail.

Oracle

The simplest way to connect to an Oracle database is to create a system DSN on the server. The user must have admin rights to create a system DSN.

- Open ODBC Data Sources 64-bit in Windows
- Click the System DSN tab
- Click Add
- Select the Oracle ODBC driver
- Fill out all relevant information
- Create the DSN

```SQL
dsn=dsnName; Uid=myUserName; Pwd=[[PASSWORD]];
```

PostgreSQL

```SQL
Driver={PostgreSQL};Server=IP address;Port=5432;Database=myDataBase;Uid=myUsername;Pwd=[[PASSWORD]];
```

Teradata

```SQL
DRIVER={Teradata Database ODBC Driver 16.20};DBCName=156.43.66.95;UID=username;PWD=[[PASSWORD]];
```

Connecting using OLE DB

Prerequisite Required

The appropriate **OLE DB driver** must be installed on the server before connecting to a data source.

Netezza

```SQL
Provider=NZOLEDB;Password=[[PASSWORD]];User ID=myUsername;Data Source=myServerAddress;Initial
Catalog=myDataBase;Port=PortNumber;
```

Oracle

Oracle Provider

```SQL
Provider=OraOLEDB;Data Source=MyOracleDB;User Id=myUsername;Password=[[PASSWORD]];
```

Microsoft Provider

```SQL
Provider=MSDAORA;Data Source=MyOracleDB;User Id=myUsername;Password=[[PASSWORD]];
```

PostgreSQL

SQL Copy

```sql
Provider=PostgreSQL OLE DB Provider;Data Source=myServerAddress;location=myDataBase;User
ID=myUsername;password=[[PASSWORD]];
```

Teradata

SQL Copy

```sql
Provider=TDOLEDB;Data Source=myServerAddress;User ID=myUsername;Password=[[PASSWORD]];
```

Troubleshooting Errors

Error	Possible Solutions
Invalid username or password	Verify the credentials by connecting to the database via DBMS or creating a DSN. If using Windows Authentication, check with your server admin and database admin that the service account Validatar is running under has permissions to the appropriate databases.
Could not connect to the database server	If the timeout takes a couple of seconds this is probably an access issue. If the timeout takes a few more seconds this probably is a networking or firewall issue. Check with your networking team to ensure Validatar can connect to the database server.
The connection string is incorrect	Verify that all required parameters for that database engine are included in the connection string. If you're connecting to Snowflake, be sure there are no spaces between the parameter value and semicolon as this will cause the connection to fail.

Data Source and Explorer Changes

Data Sources Changes

Data sources are now managed globally instead of under project settings. That's right, no more creating duplicate data sources for each project using a particular database. With this change you can:

- Create a data source once and use it across Validatar projects.
- Create and manage data environments for data sources. Go to **Settings** > **Configuration** > **Data Environments** to manage the environments. Assign the environment to a data source by navigating to the data source's settings.

Implications of Moving Data Sources

- The names of existing data sources have been concatenated to include the data source name and the project from where it was migrated.
- All data sources that existed before upgrading to 2021.3 are assigned to the **Migrated** environment.
- You can assign data sources to projects in the new **Projects** subsection found in Data Source Settings.
- We've added the ability to **merge duplicate data sources** together if they have an identical connection string. Validatar will preserve the connections to tests, profile sets, and execution history.

Explorer Changes

Explorer has also moved out of Project Settings. The new explorer area brings all of your data sources together, regardless of project, and allows you do discover their structure and full data profile. With this change you can:

- Perform a search to find database objects
- Have profile data about a data source stored in one place
  - This assumes that any duplicate data sources have been merged.

Implications of Moving the Explorer

- Quickly discover and navigate to objects in a data source by drilling into the data source on the Explorer's navigation tree. Click on the object to navigate to its data profile.

What You Need to Do

Now that you know what's changed and why the change happened, visit the following pages to learn how to clean up your environment following the upgrade:

- Data Source Clean Up
- User Group Clean Up

2022.3 Product Updates

Validatar has a new release! From major updates to minor bug fixes, we're constantly working to improve your Validatar experience so you can continue to build trust in your data. Learn about what's new below.

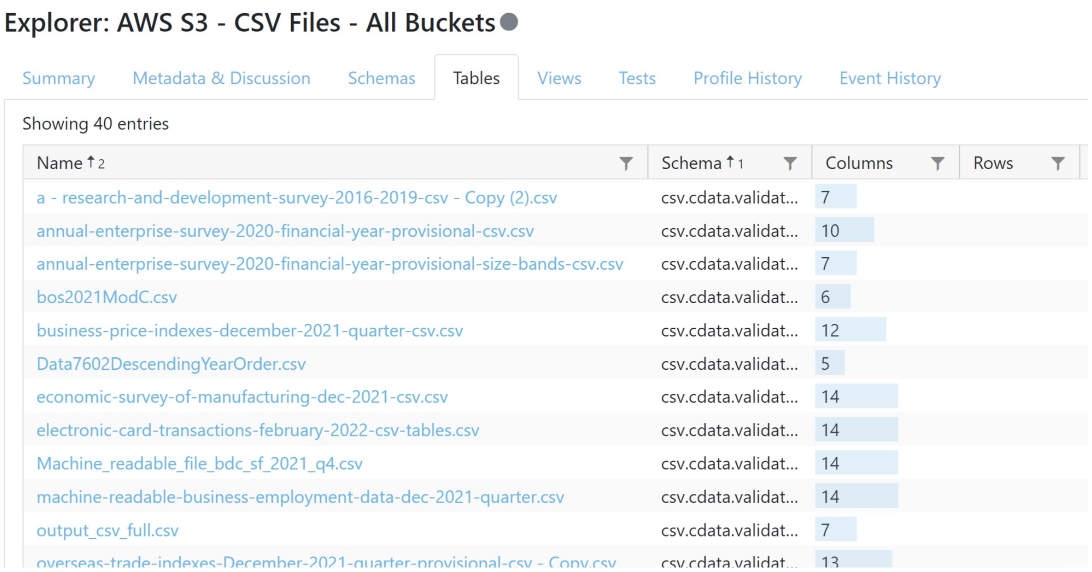
Release Date

Validatar 2022.3 was released on **September 30, 2022**.

Connect to Remote Data

- We want you to be able to connect to data no matter where it's located. Firewall? Across Clouds? No problem. **Connect to remote data sources** by installing and using a Data Agent.
- Install the Data Agent on any computer that can access your data
- Available for both Validatar Cloud and Server editions.

More Than Database Testing



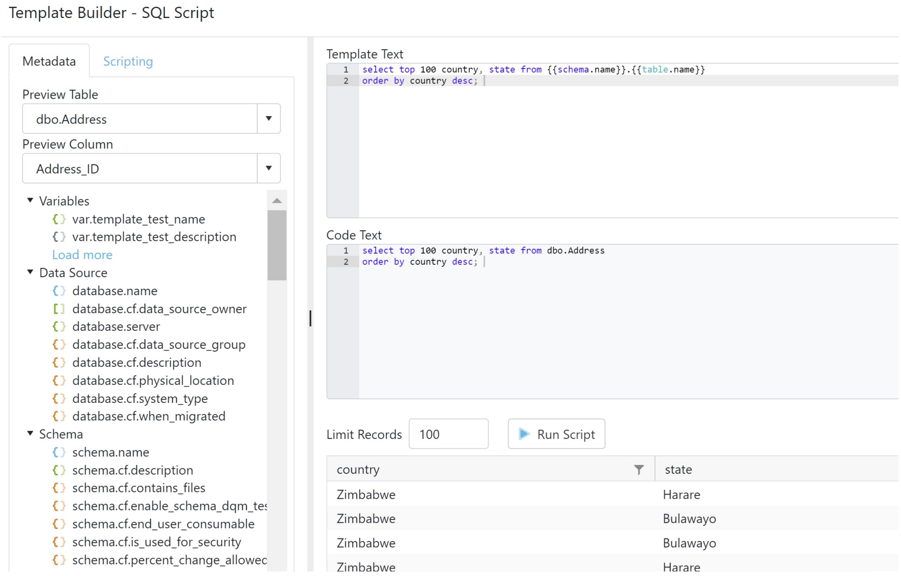
Explorer: AWS S3 - CSV Files - All Buckets

Showing 40 entries

Name ↑2	Schema ↑1	Columns	Rows
a - research-and-development-survey-2016-2019-csv - Copy (2).csv	csv.cdata.validat...	7	
annual-enterprise-survey-2020-financial-year-provisional-csv.csv	csv.cdata.validat...	10	
annual-enterprise-survey-2020-financial-year-provisional-size-bands-csv.csv	csv.cdata.validat...	7	
bos2021ModC.csv	csv.cdata.validat...	6	
business-price-indexes-december-2021-quarter-csv.csv	csv.cdata.validat...	12	
Data7602DescendingYearOrder.csv	csv.cdata.validat...	5	
economic-survey-of-manufacturing-dec-2021-csv.csv	csv.cdata.validat...	14	
electronic-card-transactions-february-2022-csv-tables.csv	csv.cdata.validat...	14	
Machine_readable_file_bdc_sf_2021_q4.csv	csv.cdata.validat...	14	
machine-readable-business-employment-data-dec-2021-quarter.csv	csv.cdata.validat...	14	
output_csv_full.csv	csv.cdata.validat...	7	
overseas-trade-indexes-December-2021-quarter-provisional-csv - Copy.csv	csv.cdata.validat...	13	

- You've mentioned the need for testing flat files and we heard you! You can now **test delimited flat files** using Python scripts.
- Test AWS S3 buckets, Azure Blob Storage, and other files.
- With the integration of Python, you can even connect to BI/Reporting tools like Tableau Server/Online.
- Integrate better with ETL Tools and Data Catalogs.
- Database Engines have been renamed to Data Source Templates.

Build Advanced Tests Quicker, Easier



Template Builder - SQL Script

Preview Table: dbo.Address

Preview Column: Address_ID

- Variables
  - var.template_test_name
  - var.template_test_description
  - Load more
- Data Source
  - database.name
  - database.cf.data_source_owner
  - database.server
  - database.cf.data_source_group
  - database.cf.description
  - database.cf.physical_location
  - database.cf.system_type
  - database.cf.when_migrated
- Schema
  - schema.name
  - schema.cf.description
  - schema.cf.contains_files
  - schema.cf.enable_schema_dqm_tes
  - schema.cf.end_user_consumable
  - schema.cf.is_used_for_security
  - schema.cf.percent_change_allowec

Template Text

```
select top 100 country, state from {{schema.name}}.{{table.name}}
order by country desc;
```

Code Text

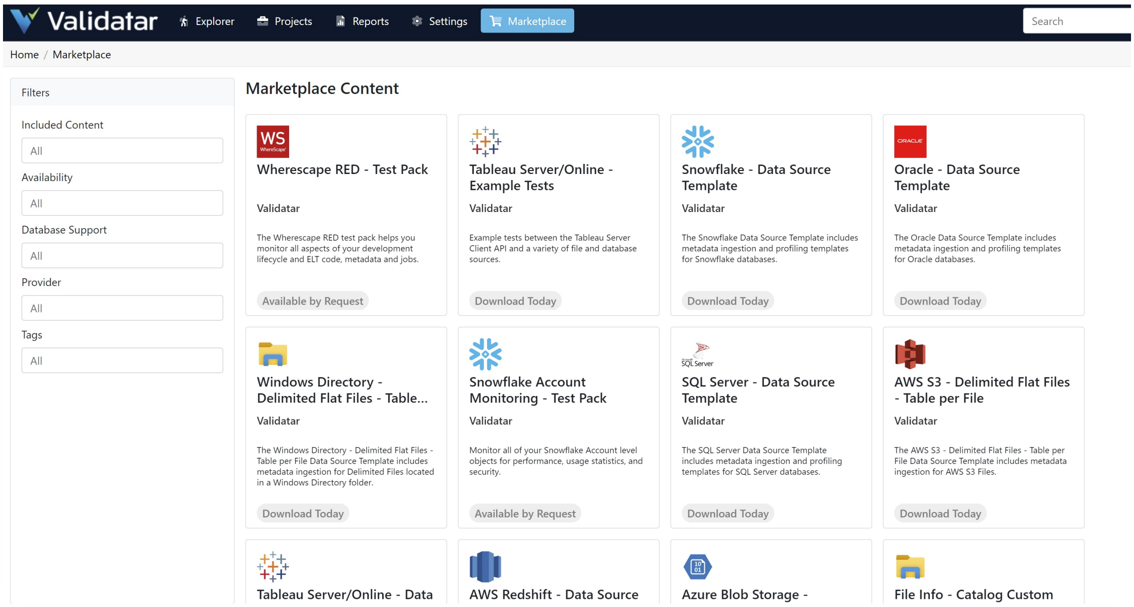
```
select top 100 country, state from dbo.Address
order by country desc;
```

Limit Records: 100

country	state
Zimbabwe	Harare
Zimbabwe	Bulawayo
Zimbabwe	Bulawayo
Zimbabwe	Harare

- We've already simplified test script writing by adding the Build Script feature on standard tests. Now we've expanded that capability to Template Tests!
- Use the Template Builder to preview template references populated in real time. No need to save the test before seeing what the materialized child test will look like anymore.
- Quickly see the results from the template scripts.
- Enhanced editor with metadata references and advanced scripting functions.

Validatar Marketplace



Filters

Included Content: All

Availability: All

Database Support: All

Provider: All

Tags: All

Marketplace Content

Wherescape RED - Test Pack
Validatar
The Wherescape RED test pack helps you monitor all aspects of your development lifecycle and ELT code, metadata and jobs.
Available by Request

Tableau Server/Online - Example Tests
Validatar
Example tests between the Tableau Server Client API and a variety of file and database sources.
Download Today

Snowflake - Data Source Template
Validatar
The Snowflake Data Source Template includes metadata ingestion and profiling templates for Snowflake databases.
Download Today

Oracle - Data Source Template
Validatar
The Oracle Data Source Template includes metadata ingestion and profiling templates for Oracle databases.
Download Today

Windows Directory - Delimited Flat Files - Table...
Validatar
The Windows Directory - Delimited Flat Files - Table per File Data Source Template includes metadata ingestion for Delimited Files located in a Windows Directory folder.
Download Today

Snowflake Account Monitoring - Test Pack
Validatar
Monitor all of your Snowflake Account level objects for performance, usage statistics, and security.
Available by Request

SQL Server - Data Source Template
Validatar
The SQL Server Data Source Template includes metadata ingestion and profiling templates for SQL Server databases.
Download Today

AWS S3 - Delimited Flat Files - Table per File
Validatar
The AWS S3 - Delimited Flat Files - Table per File Data Source Template includes metadata ingestion for AWS S3 Files.
Download Today

Tableau Server/Online - Data

AWS Redshift - Data Source

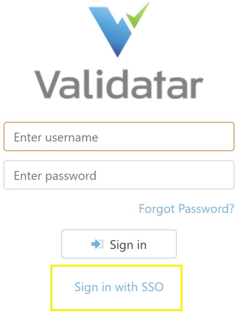
Azure Blob Storage -

File Info - Catalog Custom

- Get started with test and data source configuration even faster by shopping our new Marketplace.
- The Marketplace has pre-built Data Source Templates, Test Packs, and Custom Fields you can import into your own Validatar instance.

- After import, content can be customized to fit your testing needs.

Single Sign On (SSO)



- Manage users and authentication from a single location.
- SAML 2.0 Supported
  - Okta
  - Azure ADFS
  - AWS IAM

- Included in Enterprise Edition or as an Add-On for Team Edition.

Other Enhancements

- The metadata import process now supports custom metadata fields.
- You can now keep the Result page filters set as you navigate away from the page.
- Better password reset error messaging.

Bug Fixes

We've corrected bugs

- Where exported logs failed if one of the referenced folders did not exist.
- That caused the data grid in the Explorer to not display correctly when a metadata object doesn't contain any tests.
- That prevented new custom profiles from linking to the related data set profile during the Project Import.
- Where a report subscription could not be deleted.
- Where column statistics were incorrectly showing N/A in the Explorer.

2022.2 Product Changes

Validatar has a new release! From major updates to minor bug fixes, we're constantly working to improve your Validatar experience so you can continue to build trust in your data. Learn about what's new below.

Release Date

Validatar 2022.2 was released on **May 6, 2022**.

Get Started with More Options

- Starting in 2022.2, you have the option to install Validatar using Linux and PostgreSQL.
- Quickly learn how to best test and observe your data using our new step-by-step in-app guides.

Test Enhancements

Validatar is more than a source to target testing tool. We've highlighted our expansive testing capabilities by introducing the labels **Test Data Set** and **Control Data Set** to the testing interface. Learn more about this enhancement and other changes below.

- We've given you more control over how Validatar should evaluate test results.
  - You can now decide how the engine handles missing values returned from the test.
  - We've expanded the result success criteria options to include exact matches, percent of test and control values, and custom calculations for more nuanced scenarios.
- We've added a **Jobs tab** to the test details screen that lists all the jobs associated with the test.
- Validatar immediately cancels tests when it detects syntax issues in the test or control data set SQL.
- Use the new **Child Tests tab** inside of a Template Test to quickly navigate to the materialized child tests. Vice versa, we've added a **link back to the Template Test** to the child tests detail screen.
- Perfect your SQL scripts directly in Validatar using the new **Build Script** feature, which prevents you from running the entire test to check for syntax or semantic errors.
- Metadata isn't just logically linked to tests anymore. Navigate directly to the metadata object by clicking on the Metadata Link in the test.
- We've replaced the View Runtime Configuration button with a **Test Version tab** on the test results screen.

Changed Test Result Warning

After upgrading, existing tests will have the **Missing Key Settings** configuration set to **Fail** by default. If you're expecting a record to pass even though the test or control value is missing from the result set, change the Missing Key Setting on the test.

Job and Schedule Enhancements

- You can now **run an entire test and job folder** as a single job step. Just choose **Folders(Run the entire folder)** from the available job objects dropdown.
- You'll now receive a warning if you attempt to delete a test that has a job currently referencing it.
- We've added **Sent to Top** and **Sent to Bottom** buttons to the job steps list panel.
- You can now **add a name to job schedules**. Any existing schedules will be named Schedule 1, Schedule 2, etc.
- We've added the ability for you to **copy jobs**.
- We've expanded template test configuration options to include **not executing child tests inside a job**.

Reporting Improvements

We've extensively broadened the types of reports you can create in Validatar by changing our Standard Reports to **Standard Datasets**.

- The datasets are used to create custom reports needed to drive KPIs and trends.
- Our new datasets are driven by secured views, which you can access in the Validatar database and use for external reporting.
- Dive deeper into details for tests, jobs, profile results, and metadata objects across all Validatar projects.
- Don't worry. We still offer the Test and Job Execution and Event History reports.

Data Source Enhancements

- We've added a Sample Profile to our standard profiles that return the top 50 column values as a sample set.

Other Enhancements

- Automatically purge Event history with the new Event Purge Schedule.
- When using Move/Copy in Tests, the folders in the tree are now closed by default to allow for easier navigation.

Bug Fixes

We've corrected bugs

- Where the profile execution script was incorrectly linked to the profile execution in the user interface.
- That prevented Custom Metadata exports from including the correct data source objects.
- That prohibited the user from filtering string values correctly in Key-Value test results.
- That prevents job schedules from running when SMTP settings are incorrect.
- That prevents metadata from being ingested if a SQL Server database has multiple column-level extended properties.

2022.1 Product Changes

Validatar has a new release! From major updates to minor bug fixes, we're constantly working to improve your Validatar experience so you can continue to build trust in your data. Learn about what's new below.

Release Date

Validatar 2022.1 was released on **January 25, 2022**.

Global Search

We're introducing the ability to look up tests, jobs, metadata objects, and more by using a global search bar at the top of your Validatar screen.

- Global search requires Apache Solr to be installed and configured.

Social Metadata

Users can now post comments, ask questions, and raise potential issues about their data using our new Discussion feature. Mention your favorite team member (using the @ symbol) and get the discussion going!

- Use the **Metadata & Discussion** tab on tests, jobs, labels, and metadata objects in the Explorer to get the conversation going.

Reporting Enhancements

We've heard your requests and we've made improvements to our Reports section.

- **Create New Reports Views** from the Manage Reports screen instead of on the Report.
- **Tableau Report Path** is now required when creating a Tableau report.
- We've added a report called **Event History** that shows a historical log of events that have occurred across all Validatar projects.
- Introducing **Subscriptions** that allow users and user groups to subscribe to receive reports via email.

Notifications

Get notifications anytime you're mentioned in a comment or if there's a change to a test, job, or label you've created or modified.

- Notifications can be shown in real-time in Validatar
- or receive email notifications immediately or wrapped up in a daily report
- Configure your notification settings by opening your User Profile and selecting Notifications from the Account Profile list.

Custom Field Enhancements

We've expanded Custom Fields to not only apply to metadata objects but to Validatar objects as well.

- The original custom fields are now called **Catalog Custom Fields**.
- We've introduced **Project Custom Fields** which are also managed on a global level and can be viewed on the **Metadata & Discussion** tab on tests, jobs, and labels.

Event History

You can now view a log of events on metadata and Validatar objects.

- View metadata event history in the Explorer on the **Event History** tab for the respective object.
- View test, job, and label event history on the object's configuration screen using the **History** tab.
- View Global Event history by navigating to **Settings** > **Activity** > **Event History**.
- There's also a new Standard Report called **Event History Report** that you can configure and subscribe to.

Other Improvements

- On the test details screen, you can now view the list of jobs associated with that test by selecting the **Jobs** tab.
- We've cleaned up the Lineage import process to enhance the user experience.
- Previously, Views were listed alongside Tables in the Explorer. We've split Views into their own tab to show the distinction of the two metadata objects.

Bug Fixes

- We’ve fixed a bug where changing Project permissions for a User Group didn't immediately update the permissions.
- We've corrected an issue where overriding the default script for Data Source Metadata ingestion wasn't saving correctly.
- We've corrected a bug that didn't allow an existing Data Source's name to be edited and saved.

Releases Before 2022.1

Validatar Release Version	PDF
2021.4	View here
2021.3	View here
2021.2.1	View here
2021.2	View here

Previous Versions

Download Now!

Not on the latest version of Validatar but still need assistance? Download the documentation for previous Validatar versions below!

Version	PDF
2021.4	Click here for PDF
2021.3	Click here for PDF
2021.2	Click here for PDF
2020.4, 2021.1	Click here for PDF

Validatar EULA

Validatar Server EULA

End-User License Agreement ("Agreement")

Please read this End-User License Agreement ("Agreement") carefully before clicking the "I Agree" button, downloading or using Validatar ("Application").

By clicking the "I Agree" button, downloading or using the Application, you are agreeing to be bound by the terms and conditions of this Agreement.

This Agreement is a legal agreement between you (either an individual or a single entity) and Validatar and it governs your use of the Application made available to you by Validatar.

If you do not agree to the terms of this Agreement, do not click on the "I Agree" button and do not download or use the Application.

The Application is licensed, not sold, to you by Validatar for use strictly in accordance with the terms of this Agreement.

License

Validatar grants you a revocable, non-exclusive, non-transferable, limited license to download, install and use the Application strictly in accordance with the terms of this Agreement.

Restrictions

You agree not to, and you will not permit others to:

- license, sell, rent, lease, assign, distribute, transmit, host, outsource, disclose or otherwise commercially exploit the Application or make the Application available to any third party.
- modify, make derivative works of, disassemble, decrypt, reverse compile or reverse engineer any part of the Application.
- remove, alter or obscure any proprietary notice (including any notice of copyright or trademark) of Validatar or its affiliates, partners, suppliers or the licensors of the Application.

Intellectual Property

The Application, including without limitation all copyrights, patents, trademarks, trade secrets and other intellectual property rights are, and shall remain, the sole and exclusive property of Validatar.

Your Suggestions

Any feedback, comments, ideas, improvements or suggestions (collectively, "Suggestions") provided by you to Validatar with respect to the Application shall remain the sole and exclusive property of Validatar.

Validatar shall be free to use, copy, modify, publish, or redistribute the Suggestions for any purpose and in any way without any credit or any compensation to you.

Modifications to Application

Validatar reserves the right to modify, suspend or discontinue, temporarily or permanently, the Application or any service to which it connects, with or without notice and without liability to you.

Updates to Application

Validatar may from time to time provide enhancements or improvements to the features/functionality of the Application, which may include patches, bug fixes, updates, upgrades and other modifications ("Updates").

Updates may modify or delete certain features and/or functionalities of the Application. You agree that Validatar has no obligation to (i) provide any Updates, or (ii) continue to provide or enable any particular features and/or functionalities of the Application to you.

You further agree that all Updates will be (i) deemed to constitute an integral part of the Application, and (ii) subject to the terms and conditions of this Agreement.

Third-Party Services

The Application may display, include or make available third-party content (including data, information, applications and other products services) or provide links to third-party websites or services ("Third-Party Services").

You acknowledge and agree that Validatar shall not be responsible for any Third-Party Services, including their accuracy, completeness, timeliness, validity, copyright compliance, legality, decency, quality or any other aspect thereof. Validatar does not assume and shall not have any liability or responsibility to you or any other person or entity for any Third-Party Services.

Third-Party Services and links thereto are provided solely as a convenience to you and you access and use them entirely at your own risk and subject to such third parties' terms and conditions.

Privacy Policy

Validatar collects, stores, maintains, and shares information about you in accordance with its Privacy Policy, which is available at www.validatar.com/privacy-policy. By accepting this Agreement, you acknowledge that you hereby agree and consent to the terms and conditions of our Privacy Policy.

Term and Termination

This Agreement shall remain in effect until terminated by you or Validatar.

Validatar may, in its sole discretion, at any time and for any or no reason, suspend or terminate this Agreement with or without prior notice. In the event of Validatar terminating the Agreement other than for your failure to comply with any provision of this Agreement, you will have until the end of the purchased subscription period to discontinue use of the Application.

This Agreement will terminate immediately, without prior notice from Validatar, in the event that you fail to comply with any provision of this Agreement. You may also terminate this Agreement by deleting all copies of the Application.

Upon termination of this Agreement, you shall cease all use of the Application and delete all copies of the Application from your server, computer(s) and/or mobile devices.

Termination of this Agreement will not limit any of Validatar's rights or remedies at law or in equity in case of breach by you (during the term of this Agreement) of any of your obligations under the present Agreement.

Indemnification

You agree to indemnify and hold Validatar and its parents, subsidiaries, affiliates, officers, employees, agents, partners and licensors (if any) harmless from any claim or demand, including reasonable attorneys' fees, due to or arising out of your: (a) use of the Application; (b) violation of this Agreement or any law or regulation; or (c) violation of any right of a third party.

No Warranties

The Application is provided to you "AS IS" and "AS AVAILABLE" and with all faults and defects without warranty of any kind. To the maximum extent permitted under applicable law, Validatar, on its own behalf and on behalf of its affiliates and its and their respective licensors and service providers, expressly disclaims all warranties, whether express, implied, statutory or otherwise, with respect to the Application, including all implied warranties of merchantability, fitness for a particular purpose, title and non-infringement, and warranties that may arise out of course of dealing, course of performance, usage or trade practice. Without limitation to the foregoing, Validatar provides no warranty or undertaking, and makes no representation of any kind that the Application will meet your requirements, achieve any intended results, be compatible or work with any other software, applications, systems or services, operate without interruption, meet any performance or reliability standards or be error free or that any errors or defects can or will be corrected.

Without limiting the foregoing, neither Validatar nor any of Validatar's provider makes any representation or warranty of any kind, express or implied: (i) as to the operation or availability of the Application, or the information, content, and materials or products included thereon; (ii) that the Application will be uninterrupted or error-free; (iii) as to the accuracy, reliability, or currency of any information or content provided through the Application; or (iv) that the Application, its servers, the content, or e-mails sent from or on behalf of Validatar are free of viruses, scripts, trojan horses, worms, malware, timebombs or other harmful components.

Some jurisdictions do not allow the exclusion of or limitations on implied warranties or the limitations on the applicable statutory rights of a consumer, so some or all of the above exclusions and limitations may not apply to you.

Limitation of Liability

Notwithstanding any damages that you might incur, the entire liability of Validatar and any of its suppliers under any provision of this Agreement and your exclusive remedy for all of the foregoing shall be limited to the amount actually paid by you for the Application.

To the maximum extent permitted by applicable law, in no event shall Validatar or its suppliers be liable for any special, incidental, indirect, or consequential damages whatsoever (including, but not limited to, damages for loss of profits, for loss of data or other information, for business interruption, for personal injury, for loss of privacy arising out of or in any way related to the use of or inability to use the Application, third-party software and/or third-party hardware used with the Application, or otherwise in connection with any provision of this Agreement), even if Validatar or any supplier has been advised of the possibility of such damages and even if the remedy fails of its essential purpose.

Some states/jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so the above limitation or exclusion may not apply to you.

Severability

If any provision of this Agreement is held to be unenforceable or invalid, such provision will be changed and interpreted to accomplish the objectives of such provision to the greatest extent possible under applicable law and the remaining provisions will continue in full force and effect.

Waiver

Except as provided herein, the failure to exercise a right or to require performance of an obligation under this Agreement shall not effect a party's ability to exercise such right or require such performance at any time thereafter nor shall be the waiver of a breach constitute waiver of any subsequent breach.

For U.S. Government End Users

The Application and related documentation are "Commercial Items", as that term is defined under 48 C.F.R. §2.101, consisting of "Commercial Computer Software" and "Commercial Computer Software Documentation", as such terms are used under 48 C.F.R. §12.212 or 48 C.F.R. §227.7202, as applicable. In accordance with 48 C.F.R. §12.212 or 48 C.F.R. §227.7202-1 through 227.7202-4, as applicable, the Commercial Computer Software and Commercial Computer Software Documentation are being licensed to U.S. Government end users (a) only as Commercial Items and (b) with only those rights as are granted to all other end users pursuant to the terms and conditions herein.

Export Compliance

You may not export or re-export the Application except as authorized by United States law and the laws of the jurisdiction in which the Application was obtained.

In particular, but without limitation, the Application may not be exported or re-exported (a) into or to a nation or a resident of any U.S. embargoed countries or (b) to anyone on the U.S. Treasury Department's list of Specially Designated Nationals or the U.S. Department of Commerce Denied Person's List or Entity List.

By installing or using any component of the Application, you represent and warrant that you are not located in, under control of, or a national or resident of any such country or on any such list.

Amendments to this Agreement

Validatar reserves the right, at its sole discretion, to modify or replace this Agreement at any time. If a revision is material we will provide at least 30 days' notice prior to any new terms taking effect. What constitutes a material change will be determined at our sole discretion.

By continuing to access or use our Application after any revisions become effective, you agree to be bound by the revised terms. If you do not agree to the new terms, you are no longer authorized to use the Application.

Governing Law

The laws of Texas, United States, excluding its conflicts of law rules, shall govern this Agreement and your use of the Application. Your use of the Application may also be subject to other local, state, national, or international laws.

This Agreement shall not be governed by the United Nations Convention on Contracts for the International Sale of Good.

Contact Information

If you have any questions about this Agreement, please contact us at www.validatar.com/contact.

Entire Agreement

The Agreement constitutes the entire agreement between you and Validatar regarding your use of the Application and supersedes all prior and contemporaneous written or oral agreements between you and Validatar.

You may be subject to additional terms and conditions that apply when you use or purchase other Validatar services, which Validatar will provide to you at the time of such use or purchase.

Powered by Document360