QualiDI- ETL Test Automation Tool

User Manual

QualiDI

TABLE OF CONTENTS

This document is proprietary to Bitwise.

The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

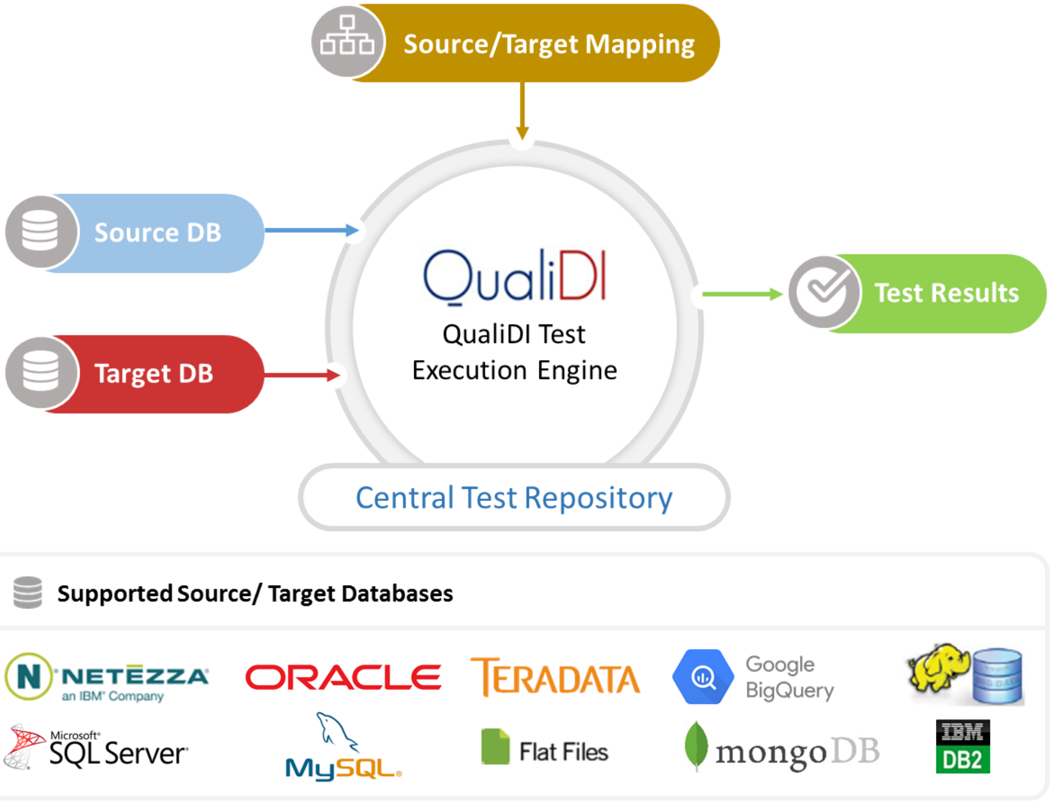
CHAPTER 1. INTRODUCTION TO QUALIDI FOR AUTOMATED ETL TEST AUTOMATION

QualiDI is our proprietary ETL test automation tool for highly regulated industries where accurate data is paramount. QualiDI improves testing effectiveness, shortens development cycles and reduces costs. It automates every aspect of the testing lifecycle, including finding bad and non-compliant data; data integration testing; testing across platforms; managing test cycles through dashboards and reports; and test scheduling.

The result: higher ROI and accelerated time to market.

1.1. Overview of QualiDI ETL Test Automation Tool

The QualiDI for Automated ETL Testing functional diagram shows that QualiDI is Enterprise-wide, end-to-end platform for centralizing testing of one or more ETL tools.



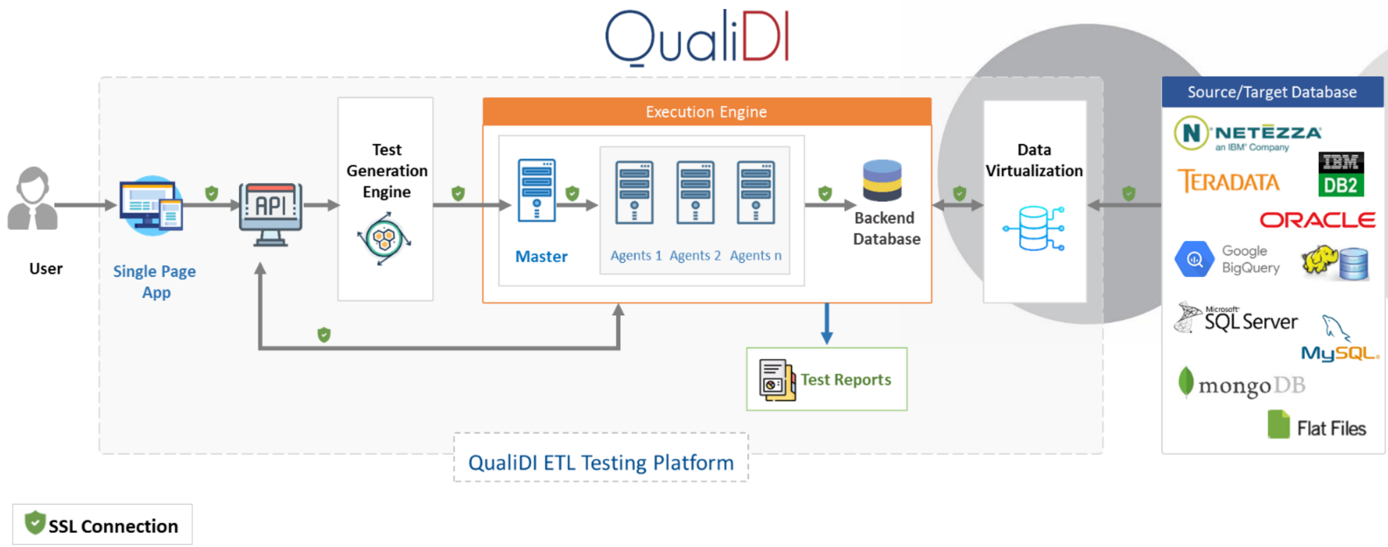
As depicted above, to get going, we simply need to connect our source and target databases and enter the mappings into QualiDI.

1.2. Architecture of QualiDI

QualiDI Architecture is designed to be highly scalable:

1. The End User can be an ETL Tester, Developer or Test Lead.
2. QualiDI is made as a single page app that avoids unnecessary page transitions, page loads, etc., hence providing a smooth, fast and a user friendly experience.

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.



3. QualiDI exposes secured APIs, can address multiple requests simultaneously, however, it would be executed in an isolated environment enabling parallelism.
4. *Execution Engine*: Execution of the test case is done with the help of Master-Slave configuration which is highly scalable and enables concurrent test case execution.
    a. *Master*: Manages the agents as well as provides you with updates on the tasks. Fallback and failures are handled here.
    b. *Agents*: Work in isolated environment to execute the test cases. The number of agents will be directly proportional to the concurrent execution.
5. *Backend Database*: Center of the repository to maintain, process and provide the information.
6. *Source/Target*: QualiDI can support the most type of databases and data sources as per your requirements. These include:
    a. Azure_MySQL
    b. AZURE_SQL Server
    c. CloudSQL
    d. Flatfile (Delimited)
    e. Flatfile (Fixedlength)
    f. Mongodb
    g. MySQL
    h. Oracle
    i. SQL Server

In addition, QualiDI works with any JDBC-compliant database or data source and Flat files (any type of delimited file and/or fixed-width files)

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

CHAPTER 2. KEY FEATURES & BENEFITS

QualiDI enables users to reduce costs, achieve higher ROIs and accelerate time to market.

2.1. Key Features

QualiDI offers an array of important features that significantly increase testing efficiency, including:

1. Meaningful auto test data generation utilizing constraints and referential integrity
2. Automated Test Case (query) generation for direct mappings
3. Add manually written Test Case(s) (Source-Target query pairs)
4. Central test case repository enabling test schedules for regression testing and monitoring through dashboards and reports
5. Test execution maintained in batches for regression and retesting
6. Test scheduling
7. Abort Test Suite or individual Test Case(s)
8. Email notifications
9. Test execution results dashboards and reports available at a click
10. Built-in defect tracking and monitoring, interfacing with a third-party defect tracking tool (JIRA)

2.2. Key Benefits

QualiDI is a proven approach that supports a company’s business strategy, operational excellence, compliance and governance through more reliable, timely data at lower cost. Following are the key benefits:

- ✓ Shift-Left Testing
- ✓ Big Data Testing
- ✓ Enables Agile Development
- ✓ Deliver Projects and Functionality Faster
- ✓ Reduce Defects
- ✓ Seamless Integration with Bitwise TDM Tool
- ✓ Improve Efficiency & Cost Savings
    - Up to **25%** Reduction in Test Cycle time
    - Up to **60%** Reduction in Test Execution
    - Up to **70%** Reduction in On-Demand Test-Data Preparation & Regression Testing Time
    - Up to **85%** Reduction in test case execution documentation

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

CHAPTER 3. KNOWING ADMINISTRATOR MODULE

Administrator module helps *Super User* to control the application level administrative tasks such as managing users, accesses, license etc.

A *Super User* is a user that you get shipped with the application in case of trial version and is created at the time of installation for the enterprise version. This user has 100% access to the application.

3.1. Accessing Administrator Module

To login to the admin module you can browse the URL http://<domain/ip>:<port>/admin e.g. http://MyDomain/admin

3.2. Manage Users

QualiDI supports two different mechanism of logging into application and user management:

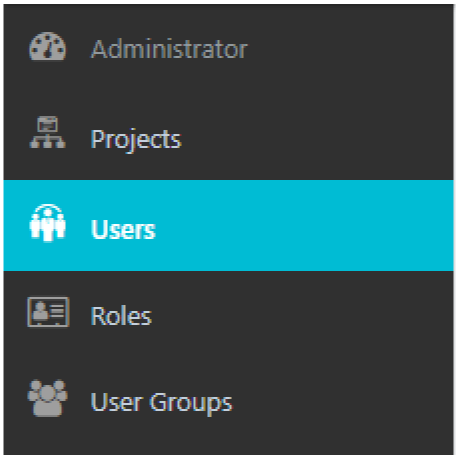
1. **Standard Login**: The identity of user is managed by admin suite.
2. **Kerberos or LDAP integration**: QualiDI uses your on-premises available identity manager and protocol with simple configurations at the time of installation. (Available for enterprise version only)

Only one mechanism is supported at a time. However, in both the cases, it is compulsory to add the user in the user section in order to make it available for use.

Adding User:

Prerequisites: You are a Super User. In trial version, all users added in the application are Super User

1. Navigate to administrator section
2. Click on **Users** at left drawer navigation



3. Click on **Add** + tab at the top right corner
4. Enter the details and click **Save** button to add the user, if you want to add more users immediately, click **Save & Add More** button
5. Newly added users will be visible on listing ≡ tab

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

Note:

1. In case of Standard login, user **Email** will act as **Username**.
2. In all other cases, **Username** and **Email** should be same as that in your identity manager.

Update User:

1. Navigate to administrator section.
2. Click on **Users** at left drawer navigation.
3. Click on **Edit** icon ✎ next to the user you want to update.
4. Update the information and click on **Save** button.

Remove existing user:

1. Navigate to administrator section.
2. Click on **Users** at left drawer navigation.
3. Click on **Delete** icon 🗑 next to the user you want to remove from QualiDI.
   Removing existing user will result in user not being able to access the application.

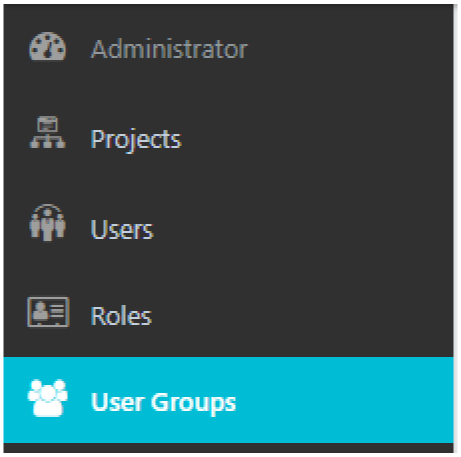
3.3. Manage User Groups

User groups are bundle of users that can be created for ease of assigning projects, roles etc.

The users added in User section can be used to create a User Group.

Adding User Group:

1. Navigate to administrator section.
2. Click on **User Groups** at left drawer navigation.



3. Click on **Add** + tab at the top right corner.
4. Enter the **Group Name** and **Description.**
5. Select users from the **Add Users** multi-select box and click **Add** button to add the user to the group.
6. Click **Create Group** button.

7. Newly added User Group will be visible on listing ≡ tab.

Update User Groups:

1. Navigate to administrator section.
2. Click on **User Groups** at left drawer navigation.
3. Click on **Edit** icon ✎ next to the user group you want to update.
4. You can add new users by selecting **Add Users** or/and you can remove users by clicking **Remove user** icon − next to existing users in **Group members** list
5. Click on **Update Group** button to save your changes.

Remove User Group:

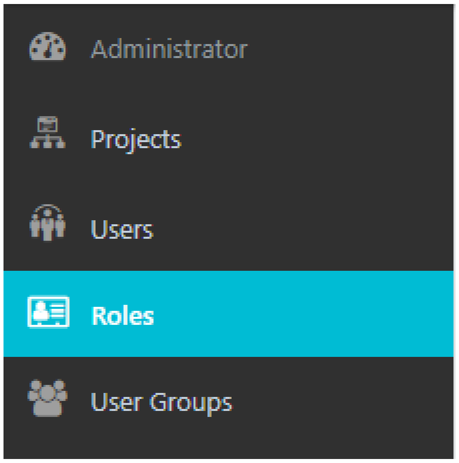
1. Navigate to administrator section.
2. Click on **User Groups** at left drawer navigation.
3. Click on **Delete** icon 🗑 next to the user group you want to remove.

3.4. Manage Roles

A role can be defined to control the accesses and permissions of any user.

Adding Roles:

1. Navigate to administrator section.
2. Click on **Roles** at left drawer navigation.



3. Click on **Add** + tab at the top right corner.
4. Enter the **Role Name** and **Description.**
5. Select the privileges that you want to give to his role form **Privileges Section**.
6. Click **Create Role** button.
7. Newly added Role will be visible on listing ≡ tab.

Update Roles:

1. Navigate to administrator section.
2. Click on **Roles** at left drawer navigation.
3. Click on **Edit** icon ✎ next to the Role you want to update.
4. You can add new privileges by selecting respective check boxes or/and you can remove existing privileges by deselecting the respective check boxes.
5. Click on **Update Role** button to save your changes.

Remove Roles:

1. Navigate to administrator section.
2. Click on **Roles** at left drawer navigation.
3. Click on **Delete** icon 🗑 next to the role you want to delete.

By default, QualiDI comes with three inbuilt roles that can be assigned to users at the time of project assignments. Following are the inbuilt roles:

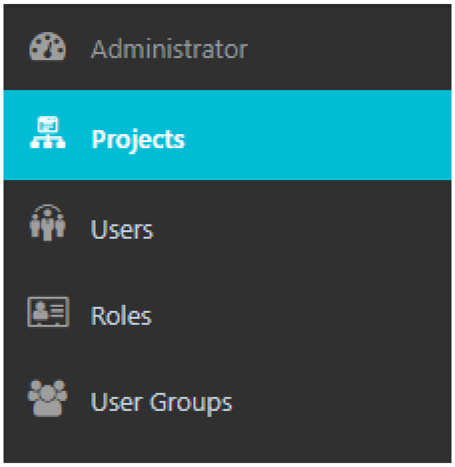
1. Project Administrator
2. Business Analyst
3. Quality Assurance

3.5. Manage Projects

A project is a collection of releases, mappings, tests and so forth that is used by a defined team to track and develop a common set of related work.

Adding Projects:

1. Navigate to administrator section.
2. Click on **Projects** at left drawer navigation.



3. Click on **Add** + tab at the top right corner.
4. Enter the **Project Name** and **Description.**

5. Select the **Visibility Level** radio button, information about the option can be seen by hovering mouse on info icon ℹ.
6. Select the tools you want this project available for.
7. Select the users from **Add Users** multi-select box and/or select the user group from **User Group** multi-select box.
8. Assign one or more **role** to the selected user(s) and/or user group(s) by selecting the same from **User Role** multi-select box.
9. Click on **Add** button next to the **User Roles** to assign user/user group with specified role to the project.
10. Newly added user/user group will be visible on list **Add new user to project**.
11. Click on **Create Project** button to save new project.
12. Newly added Project will be visible on listing ≡ tab.

A new project can be created even without adding any users or user group to it.

Update Project:

1. Navigate to administrator section.
2. Click on **Projects** at left drawer navigation.
3. Click on **Edit** icon ✎ next to the Project you want to update.
4. You can update the required information and add/remove users, user groups or even update the roles of existing users/user groups in the project.
5. Click on **Confirm Changes** button to save your changes.

Remove Projects:

1. Navigate to administrator section.
2. Click on **Projects** at left drawer navigation.
3. Click on **Delete** icon 🗑 next to the project you want to delete.

3.6. Maintain Library

The Library contains all the template like items that can be used for Test Case generation in QualiDI. The scope of these artifacts is global and will be available to all the projects by default.

QualiDI uses Column and Table Level Query Templates to generate test cases for Straight Pulls. QualiDI is shipped with some of the most important and frequently used Query Templates. This library can be maintained by Super Users and can be used by any users with the appropriate role.

Adding Library: Query Template:

1. Navigate to administrator section.
2. Click on **Library** at left drawer navigation.

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

3. Next click on **Query Template**.
4. Click on **Add** + tab at the top right corner.
5. Enter the **Title** and **Description.**
6. Select the **Type** for the query template you want to add.
7. Start typing your **Source** and **Target Template.**
8. Click **Save Template** button.
9. Newly added **Query Template** will be visible on listing ≡ tab. These are classified in accordions named as **TABLE LEVEL** and **COLUMN LEVEL**.

Query templates support certain *variables* that are replaced with actual values at the time of Test Query generation.

While defining these query templates system is intelligent to suggest you the relative variables names as you type. The name of variables itself explain the purpose, following is the list of these variables:

1. SourceTable
2. TargetTable
3. SourceColumn
4. TargetColumn
5. SourceColumns
6. TargetColumns
7. SourceVariable
8. TargetVariable

To use these variables, enclose them with two curly braces, e.g.: {{SourceTable}}

Update Library: Query Template:

1. Navigate to administrator section.
2. Click on **Library** at left drawer navigation.
3. Next click on **Query Template**.
4. Click on **Edit** icon ✎ next to the Query Template you want to update.
5. You can make the required changes to the query templates.
6. Click on **Save Template** button to save your changes.

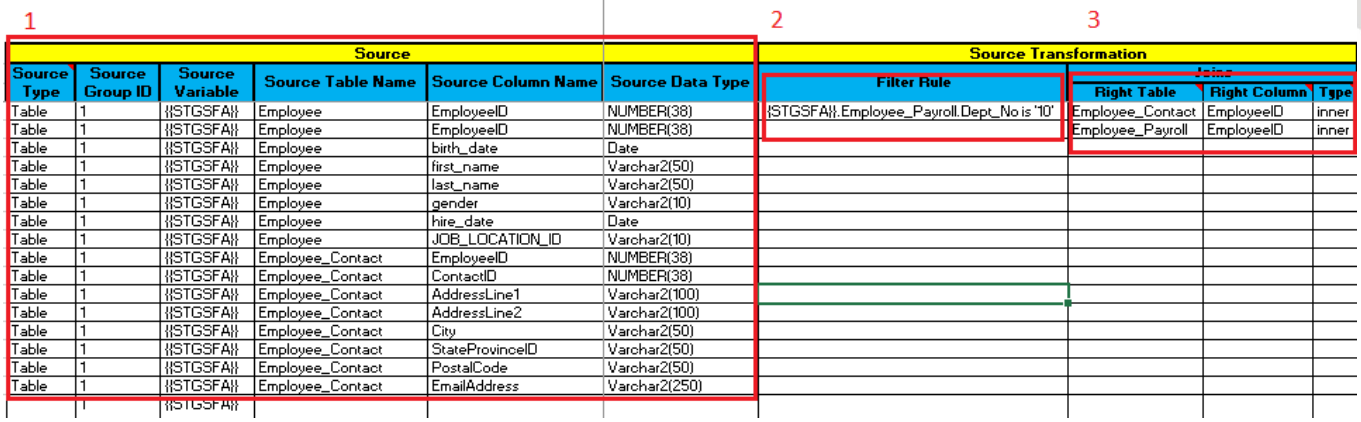
Remove Library: Query Template:

1. Navigate to administrator section.
2. Click on **Library** at left drawer navigation.
3. Next click on **Query Template**.
4. Click on **Delete** icon 🗑 next to the query template you want to delete.

CHAPTER 4. QUALIDI MAPPING SHEET

Mapping sheet is a core part of QualiDI application. It is an Excel Sheet having source to target mappings. Left most columns have information about source tables and fields along with joins and filters to be applied on source data.
Right side columns have information about target tables and fields.

Below is the brief information about columns in Source part of the sheet.



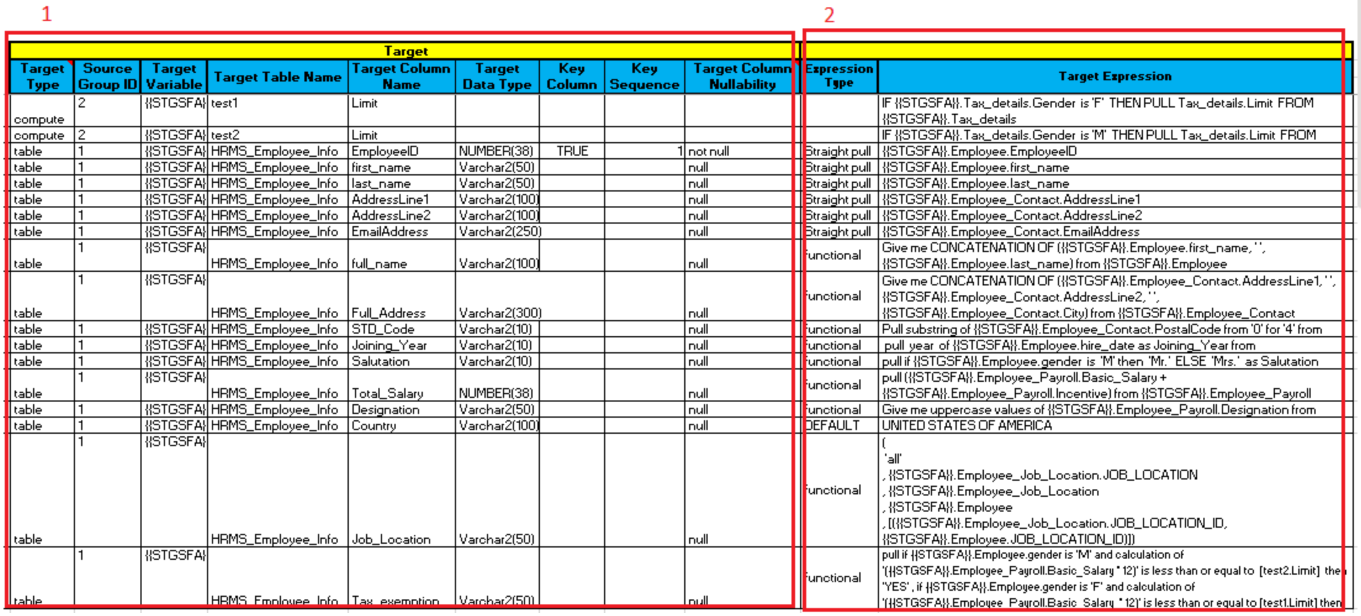
Source						Source Transformation			
							Joins		
Source Type	Source Group ID	Source Variable	Source Table Name	Source Column Name	Source Data Type	Filter Rule	Right Table	Right Column	Type
Table	1	{{STGSFA}}	Employee	EmployeeID	NUMBER(38)	{STGSFA}}.Employee_Payroll.Dept_No is '10'	Employee_Contact	EmployeeID	inner
Table	1	{{STGSFA}}	Employee	EmployeeID	NUMBER(38)		Employee_Payroll	EmployeeID	inner
Table	1	{{STGSFA}}	Employee	birth_date	Date				
Table	1	{{STGSFA}}	Employee	first_name	Varchar2(50)				
Table	1	{{STGSFA}}	Employee	last_name	Varchar2(50)				
Table	1	{{STGSFA}}	Employee	gender	Varchar2(10)				
Table	1	{{STGSFA}}	Employee	hire_date	Date				
Table	1	{{STGSFA}}	Employee	JOB_LOCATION_ID	Varchar2(10)				
Table	1	{{STGSFA}}	Employee_Contact	EmployeeID	NUMBER(38)				
Table	1	{{STGSFA}}	Employee_Contact	ContactID	NUMBER(38)				
Table	1	{{STGSFA}}	Employee_Contact	AddressLine1	Varchar2(100)				
Table	1	{{STGSFA}}	Employee_Contact	AddressLine2	Varchar2(100)				
Table	1	{{STGSFA}}	Employee_Contact	City	Varchar2(50)				
Table	1	{{STGSFA}}	Employee_Contact	StateProvinceID	Varchar2(50)				
Table	1	{{STGSFA}}	Employee_Contact	PostalCode	Varchar2(50)				
Table	1	{{STGSFA}}	Employee_Contact	EmailAddress	Varchar2(250)				
	1	{{STGSFA}}							

1. **Source information**
   a. Source Type – Below are acceptable values
      Table
      File
      Compute
   b. Source Group ID: This field signifies group of source table on which source level filters and joins should be applied. In above screenshot, filter and join will be applied for fields from source group ID – 1, as filter condition and join condition is applied tables present into group id 1.
   c. Variable name- This are the variables which allow us to execute test cases across multiple environments. Values for these variables get replaced as specified in Configuration file. Details about configuration files are mentioned in section 7.7
   d. Source Name – Here one can mention name of the source file or table.
   e. Field name - Here one can mention name of the field present in source file or table.
   f. Field Data Type – This is the data type of the field.
2. **Filter Information**
   Here one can specify filter condition for particular source group ID. User can write it in any row having corresponding group ID.
   Filter will be applied for all the columns mentioned in similar source group ID.
   In above example, Employee_Payroll.Dept_No is '10' filter is applied for all fields having source group ID '1'.
3. **Join Information**
   Here one can specify Join to be applied on source side. User can write the 'Right side' table and key in front of the field with which we want to have join.
   In above snapshot, Employee.EmployeeID join is taken against Employee_Contact.EmployeeID as we as, against Employee_Payroll. EmployeeID
   Joins Types
   a. Inner

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

b. Left
c. Right

Below is the brief information about columns in Target part of the sheet.



Target										
Target Type	Source Group ID	Target Variable	Target Table Name	Target Column Name	Target Data Type	Key Column	Key Sequence	Target Column Nullability	Expression Type	Target Expression
compute	2	{{STGSFA}}	test1	Limit						IF {{STGSFA}}.Tax_details.Gender is 'F' THEN PULL Tax_details.Limit FROM {{STGSFA}}.Tax_details
compute	2	{{STGSFA}}	test2	Limit						IF {{STGSFA}}.Tax_details.Gender is 'M' THEN PULL Tax_details.Limit FROM
table	1	{{STGSFA}}	HRMS_Employee_Info	EmployeeID	NUMBER(38)	TRUE	1	not null	Straight pull	{{STGSFA}}.Employee.EmployeeID
table	1	{{STGSFA}}	HRMS_Employee_Info	first_name	Varchar2(50)			null	Straight pull	{{STGSFA}}.Employee.first_name
table	1	{{STGSFA}}	HRMS_Employee_Info	last_name	Varchar2(50)			null	Straight pull	{{STGSFA}}.Employee.last_name
table	1	{{STGSFA}}	HRMS_Employee_Info	AddressLine1	Varchar2(100)			null	Straight pull	{{STGSFA}}.Employee_Contact.AddressLine1
table	1	{{STGSFA}}	HRMS_Employee_Info	AddressLine2	Varchar2(100)			null	Straight pull	{{STGSFA}}.Employee_Contact.AddressLine2
table	1	{{STGSFA}}	HRMS_Employee_Info	EmailAddress	Varchar2(250)			null	Straight pull	{{STGSFA}}.Employee_Contact.EmailAddress
table	1	{{STGSFA}}	HRMS_Employee_Info	full_name	Varchar2(100)			null	unctional	Give me CONCATENATION OF ({{STGSFA}}.Employee.first_name, ' ', {{STGSFA}}.Employee.last_name) from {{STGSFA}}.Employee
table	1	{{STGSFA}}	HRMS_Employee_Info	Full_Address	Varchar2(300)			null	unctional	Give me CONCATENATION OF ({{STGSFA}}.Employee_Contact.AddressLine1, ' ', {{STGSFA}}.Employee_Contact.AddressLine2, ' ', {{STGSFA}}.Employee_Contact.City) from {{STGSFA}}.Employee_Contact
table	1	{{STGSFA}}	HRMS_Employee_Info	STD_Code	Varchar2(10)			null	unctional	Pull substring of {{STGSFA}}.Employee_Contact.PostalCode from '0' for '4' from
table	1	{{STGSFA}}	HRMS_Employee_Info	Joining_Year	Varchar2(10)			null	unctional	pull year of {{STGSFA}}.Employee.hire_date as Joining_Year from
table	1	{{STGSFA}}	HRMS_Employee_Info	Salutation	Varchar2(10)			null	unctional	pull if {{STGSFA}}.Employee.gender is 'M' then 'Mr.' ELSE 'Mrs.' as Salutation
table	1	{{STGSFA}}	HRMS_Employee_Info	Total_Salary	NUMBER(38)			null	unctional	pull ({{STGSFA}}.Employee_Payroll.Basic_Salary + {{STGSFA}}.Employee_Payroll.Incentive) from {{STGSFA}}.Employee_Payroll
table	1	{{STGSFA}}	HRMS_Employee_Info	Designation	Varchar2(50)			null	unctional	Give me uppercase values of {{STGSFA}}.Employee_Payroll.Designation from
table	1	{{STGSFA}}	HRMS_Employee_Info	Country	Varchar2(100)			null	DEFAULT	UNITED STATES OF AMERICA
table	1	{{STGSFA}}	HRMS_Employee_Info	Job_Location	Varchar2(50)			null	unctional	('all' , {{STGSFA}}.Employee_Job_Location.JOB_LOCATION , {{STGSFA}}.Employee_Job_Location , {{STGSFA}}.Employee , [({{STGSFA}}.Employee_Job_Location.JOB_LOCATION_ID, {{STGSFA}}.Employee.JOB_LOCATION_ID)])
table	1	{{STGSFA}}	HRMS_Employee_Info	Tax_exemption	Varchar2(50)			null	unctional	pull if {{STGSFA}}.Employee.gender is 'M' and calculation of '({{STGSFA}}.Employee_Payroll.Basic_Salary * 12)' is less than or equal to [test2.Limit] then 'YES' , if {{STGSFA}}.Employee.gender is 'F' and calculation of '({{STGSFA}}.Employee_Payroll.Basic_Salary * 12)' is less than or equal to [test1.Limit] then

1. Target information

a. Target Type - – Below are acceptable values
   - Table
   - File
   - Compute

b. Source Group ID – This field signifies group of source table on which source level filters and joins should be applied. In above screenshot, filter and join will be applied for fields from source group ID – 1, as filter condition and join condition is applied tables present into group id 1.

c. Target Variable - This are the variables which allow us to execute test cases across multiple environments. Values for these variables get replaced as specified in Configuration file. Details about configuration files are mentioned in *Section 7.7*

d. Target Table Name - Here one can mention name of the target file or table.

e. Target Column Name - Here one can mention name of the column present in target file or table.

f. Target Data Type – This is the data type of the column.

g. Key Column – Here one can mention "TRUE" for the primary key against which system will validate source and target data at the time of Test Execution. If no primary key base comparison is required, the key column can be left empty.

h. Key Sequence – In case of multiple primary key combinations; user can provide the priority/sequence of the key columns.

i. Target Column Nullability – Here user can specify the nullability of the field. Below are acceptable values
   - Not Null
   - Null

2. Transformation rules

User needs to provide transformation rules that are applied on source data to populate target data.

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

a. Expression Type – Below are the acceptable values
   Straight Pull – Data from Source is populated as it is in target
   Default – Constant value
   Functional – Various Transformations are applied on source data to populate target data.
b. Expressions – User can use the English like syntax to specify the transformations to be applied on source data. It includes- String operations, mathematical calculations, various operators etc.

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

CHAPTER 5. CREATING AND EXECUTING ETL TEST CASES

This chapter takes the example of ETL job where, we extract employee data from two different data sources, select specific data to include in a MySQL database, and show how we can make use of QualiDI for ETL test Automation.

For writing your Transformation rules you can use out-of-box business natural language that is processed by or Business Rule Engine.

You will work with the ETL job as you learn how to test the same using QualiDI

The sequence of developing automated tests involves following steps:

1. Create the test project, the container of all test related artifacts. For further information, see *Setting up Test Project*
2. Connect to the databases which hold the data. For further information, see *Creating the Database connections*
3. Add/create the new mapping sheet that will hold all the mappings in it. For further information, see *Creating Mapping Sheet for ETL Test-Suite*
4. Create the connection groups, For further information, see *Grouping Source and Target Databases*
5. Add/create the test suite for the automated test execution. For further information, see *Creating Test Suite*
6. Execute the test suite for the results and reports. For further information, see *Execution of Test Cases/Suite*
7. Access the reports to see test suite/cases execution status. For further information, see *Reports*

5.1. Understanding Example (Use Case):

Let us consider an example of HRMS (Human Resource Management System) use case.

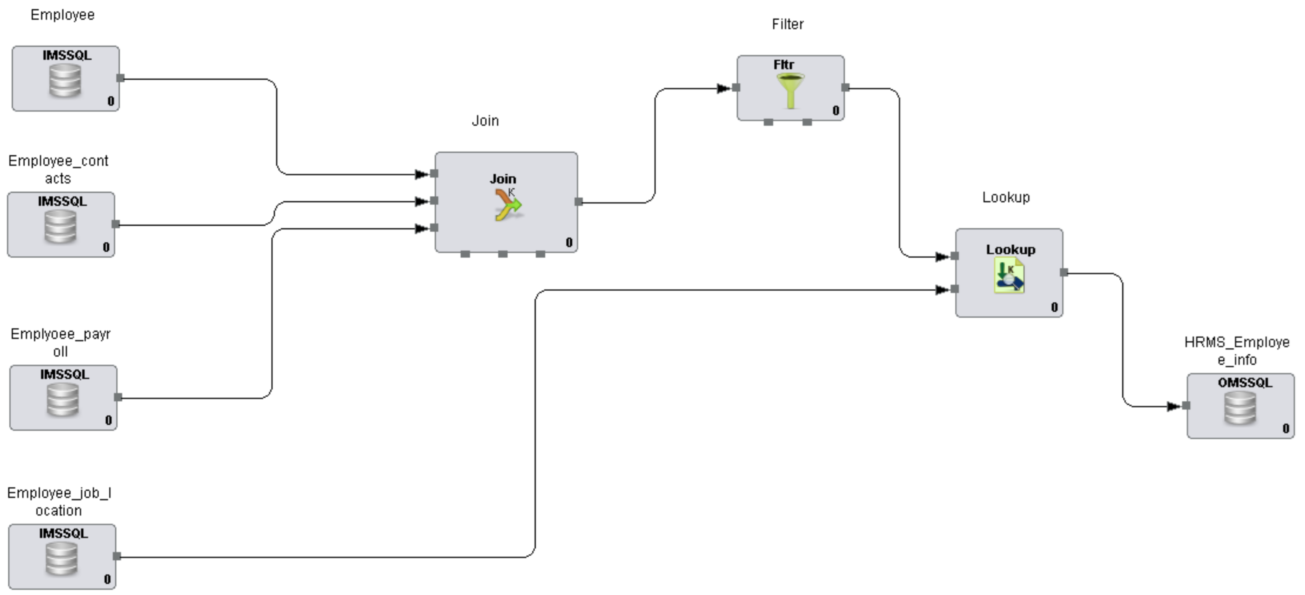
Here we are –

- Applying filter on basis of *Employee_Payroll.Dept_No* data
- Applying some transformations on filtered *Employee* data
- Taking its join with *Employee_Contact* and *Employee_Payroll* table
- Taking its Lookup with *Employee_Job_Location* table
- Output of the ETL jobs is – *HRMS_Employee_Info*

This ETL job is further tested using QualiDI.

To start with the testing, it is needed to know the ETL job under test. Following is the diagrammatic representation of our example:

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.



Following are the data models for the source and target table(s). To replicate the use case, tables with below structures can be created in any database supported by QualiDI.

Source table: Employee

Source Name	Field Name	Field Data Type
Employee	ID	NUMBER(38)
Employee	EMPLOYEEID	VARCHAR(25)
Employee	BIRTH_DATE	DATE
Employee	FIRST_NAME	VARCHAR(25)
Employee	LAST_NAME	VARCHAR(25)
Employee	JOB_LOCATION_ID	VARCHAR(10)
Employee	GENDER	VARCHAR(10)
Employee	HIRE_DATE	DATE

Source table: Employee_Contact

Source Name	Field Name	Field Data Type
Employee_Contact	ID	NUMBER(38)
Employee_Contact	CONTACT_ID	VARCHAR(25)
Employee_Contact	EMPLOYEEID	VARCHAR(25)
Employee_Contact	ADDRESSLINE1	VARCHAR(50)

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

Employee_Contact	ADDRESSLINE2	VARCHAR(50)
Employee_Contact	CITY	VARCHAR(25)
Employee_Contact	STATEPROVINCEID	VARCHAR(25)
Employee_Contact	POSTALCODE	NUMBER(38)
Employee_Contact	EMAILADDRESS	VARCHAR(25)

Source table: Employee_payroll

Source Name	Field Name	Field Data Type
Employee_payroll	ID	NUMBER(38)
Employee_payroll	EMPLOYEEID	VARCHAR2(25)
Employee_payroll	BASIC_SALARY	NUMBER(38)
Employee_payroll	INCENTIVE	NUMBER(38)
Employee_payroll	DESIGNATION	VARCHAR2(25)
Employee_payroll	DEPT_NO	NUMBER(38)

Source table: Employee_Job_Location

Source Name	Field Name	Field Data Type
Employee_Job_Location	JOB_LOCATION_ID	VARCHAR2(10)
Employee_Job_Location	JOB_LOCATION	VARCHAR2(50)

Source table: DEPARTMENT

Source Name	Field Name	Field Data Type
DEPARTMENT	DEPT_NO	VARCHAR2(10)
DEPARTMENT	DEPT_NAME	VARCHAR2(50)

Target table: HRMS_Employee_Info

Source Name	Field Name	Field Data Type
HRMS_Employee_Info	EMPLOYEEID	VARCHAR2(25)
HRMS_Employee_Info	FIRST_NAME	VARCHAR2(25)
HRMS_Employee_Info	LAST_NAME	VARCHAR2(25)
HRMS_Employee_Info	ADDRESSLINE1	VARCHAR2(50)

HRMS_Employee_Info	ADDRESSLINE2	VARCHAR2(50)
HRMS_Employee_Info	EMAILADDRESS	VARCHAR2(25)
HRMS_Employee_Info	FULL_NAME	VARCHAR2(51)
HRMS_Employee_Info	FULL_ADDRESS	VARCHAR2(127)
HRMS_Employee_Info	STD_CODE	VARCHAR2(4)
HRMS_Employee_Info	JOINING_YEAR	NUMBER
HRMS_Employee_Info	SALUTATION	VARCHAR2(4)
HRMS_Employee_Info	TOTAL_SALARY	NUMBER
HRMS_Employee_Info	DESIGNATION	VARCHAR2(25)
HRMS_Employee_Info	COUNTRY	CHAR(24)
HRMS_Employee_Info	JOB_LOCATION	VARCHAR2(50)
HRMS_Employee_Info	TAX_EXEMPTION	VARCHAR2(3)

The mapping information for this ETL job is in an excel sheet:

HRMS_Employee_Info_mapping_sheet.x

Please refer ***Chapter: QualiDI Mapping Sheet*** for detailed information related Mapping Sheet.

The format of the mapping document is completely compatible with the QualiDI.

If you want to replicate the example and use the exact input data, you can simply import the following data into your database server(s):

Employee.csv Employee_Contact.csv Employee_Job_Location.csv DEPARTMENT.csv Employee_payroll.csv

Output data:

HRMS_Employee_Info.csv

All the required files for this example are at below path

http://<machine_ip:8080>/SampleProjectFiles/Employee_Usecase.zip

E.g. *http://MyDomain:8080/SampleProjectFiles/Employee_Usecase.zip*

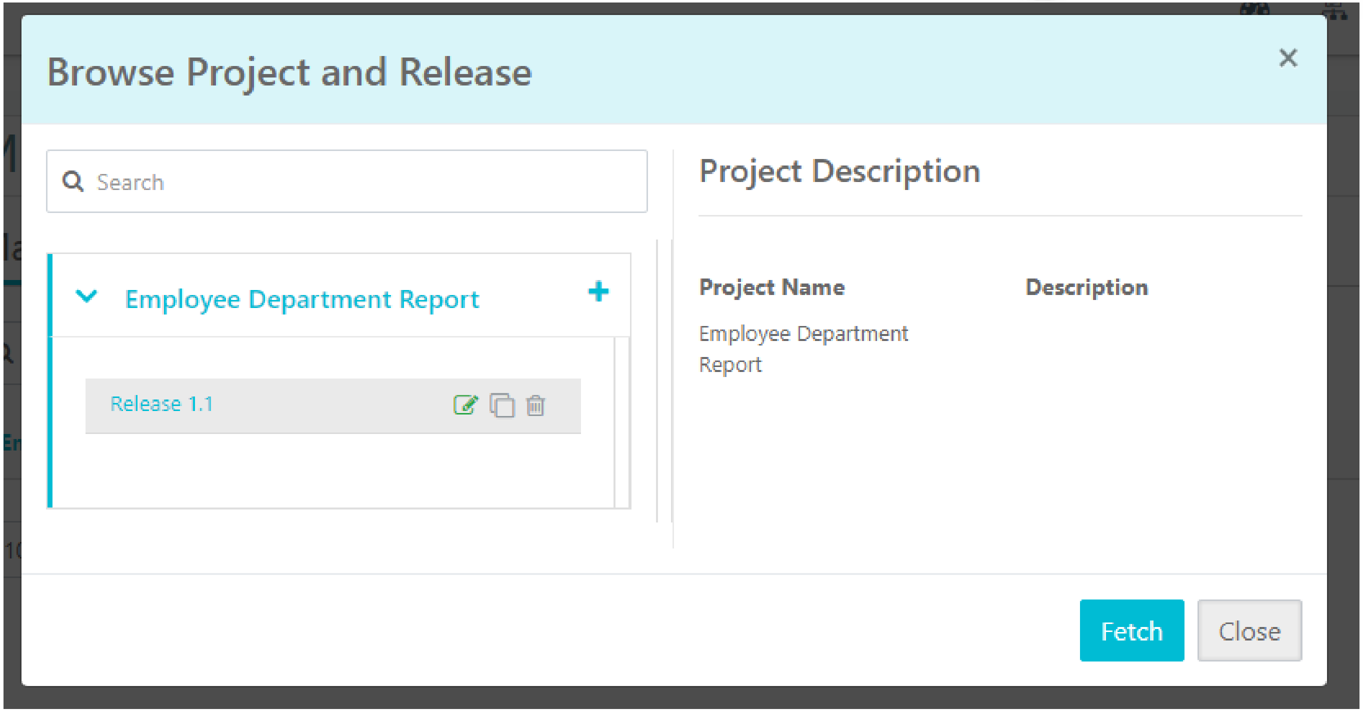
Alternatively, you can simply generate the test data using our **Test Data Management** tool.

This use case will be referred further in the process of learning QualiDI for ETL Test Automation.

5.2. Setting up Test Project

Here, we will create a base for our ETL test suite:

1. Browse the QualiDI application using a web browser (google chrome recommended).
2. Login to the QualiDI application using valid username and password
3. Select the Project created from admin suite and create Release under selected Project
4. Select the Release created in Step 3
5. Click on '**Fetch**'



Project Name: Employee Department Report

Description:

By following above steps, we are done with basic project setup needed.

5.3. Creating the Database Connections

As per the example in this document now we need to create database connections for source and target.

Prerequisites:

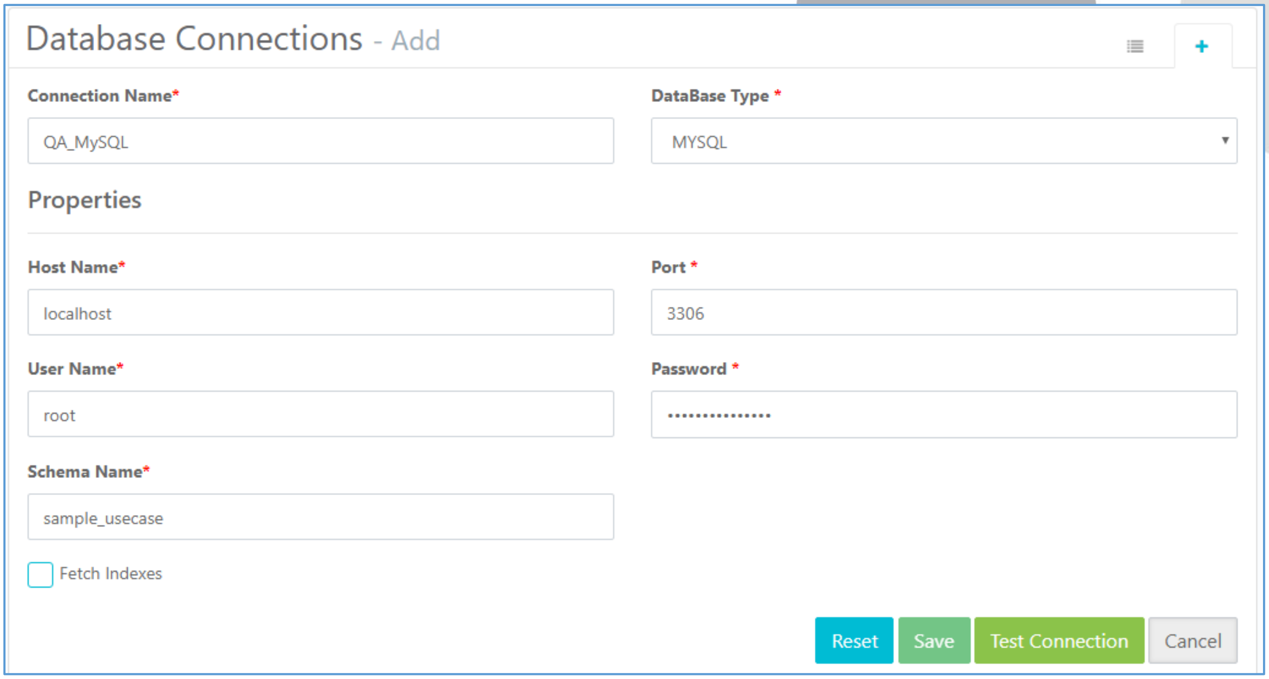
- You have access to Source databases, in our case, it will be Microsoft SQL Server and MySQL database

To create new desired database connections:

1. Click on **Database Connections** at the left-hand module navigation slider (If you do see navigation slider, click on 3 horizontal bar icon at top left corner of the screen)
2. Click on **Add New Connection** button on the top right corner of the screen
3. Enter **Connection Name** and select **Database Type** from the dropdown options. For our example select **MYSQL** for our example.

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

4. Now the properties for the database connection will be listed based on the value selected from **Database Type** dropdown.
5. In this example, enter the **Host Name**, **Port**, **Username**, **Password** and **Schema Name** in the respective text boxes.



Database Connections - Add

Connection Name*: QA_MySQL

DataBase Type *: MYSQL

Properties

Host Name*: localhost

Port *: 3306

User Name*: root

Password *: •••••••••••••••

Schema Name*: sample_usecase

☐ Fetch Indexes

6. As you fill all the mandatory fields for any connection, **Test Connection** button will be enabled. Go ahead and click the same to test the connection with the database.
7. Once the connection with the database is a success, the **Save** button will be enabled. Now to save our database connection hit the **Save** button.

Now for our example, we can repeat all the above steps and create a new connection for our Source/target Microsoft SQL Server Database. Remember, that the connection properties values mentioned in step 5 & 6 will be different this time and very easy to understand.

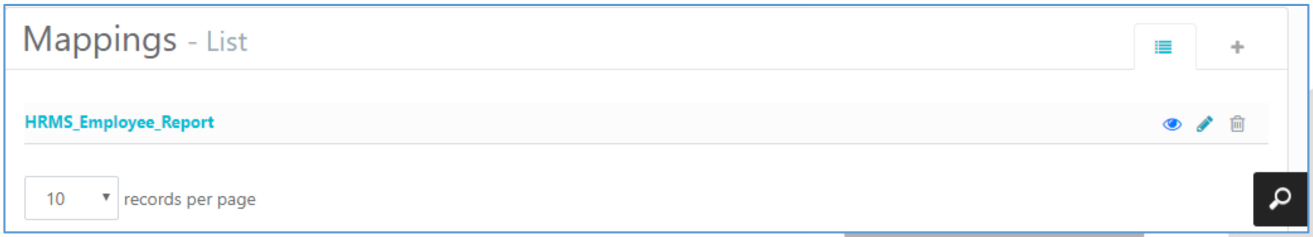
5.4. Creating Mapping Sheet for ETL Test-Suite

To create all our mappings under one umbrella we now will create a new mapping sheet.

To create a new mapping sheet:

1. Click on **Mapping Sheet** at the left-hand module navigation slider (If you do see navigation slider, click on 3 horizontal bar icon at top left corner of the screen)
2. Click on **Add Mapping Sheet** button on the top right corner of the screen
3. Enter **Title**, **Description** and hit **Save** button in the dialog box.

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.



Now, we have a new Mapping Sheet that we will use to define/add the mappings further.

5.5. Defining Mappings under Mapping Sheet

To create mapping we can follow one of the two approaches available. You can even use both the approaches

1. Adding mappings using QualiDI web-interface.
2. Bulk import using Excel Sheet.

Mappings using QualiDI Web-interface

You can create mapping rules using this interface for both source and target database

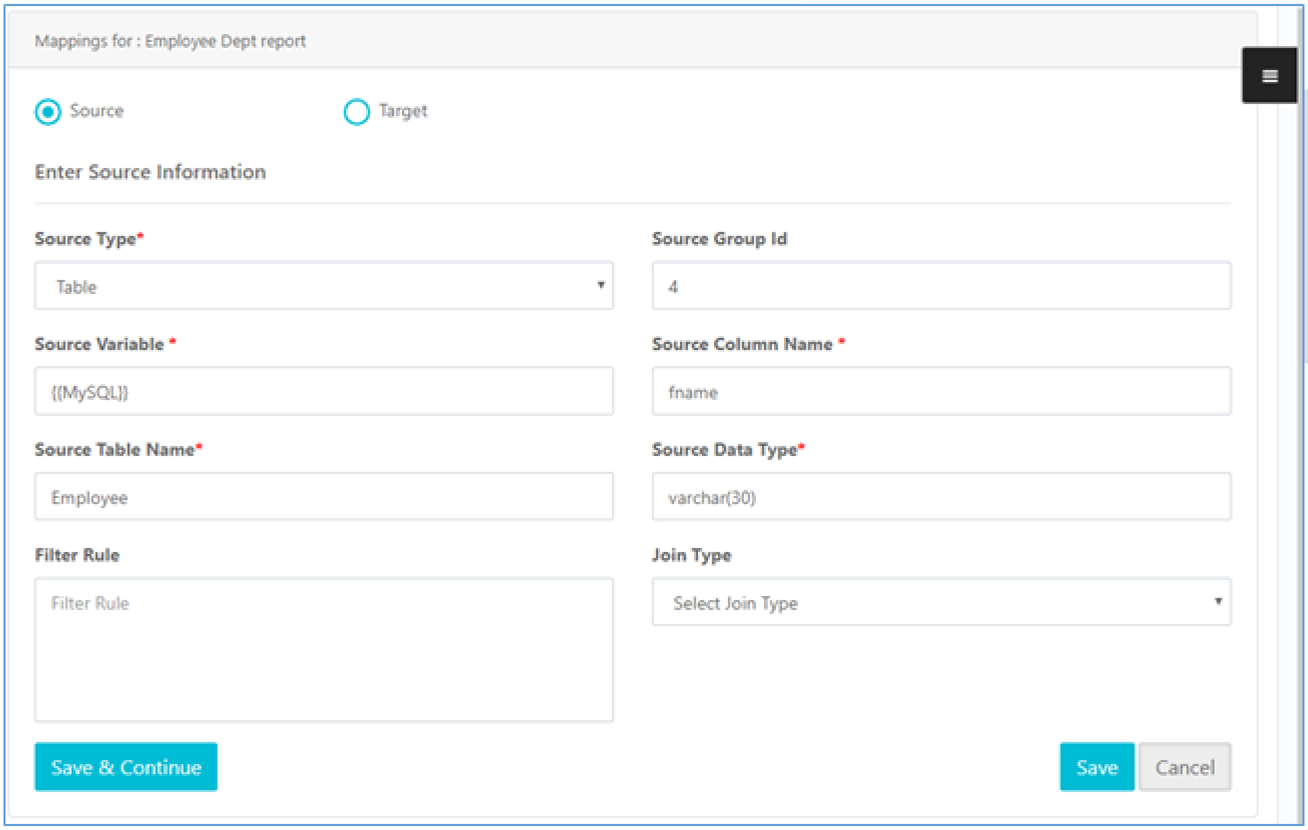
To create mapping rules using web-interface:

1. Click on **Mapping Sheet** at the left-hand module navigation slider (If you do see navigation slider, click on 3 horizontal bar icon at top left corner of the screen)
2. Click on **Edit** mapping sheet from the **Mapping List** that we created in steps in *Creating Mapping Sheet for ETL Test-Suite*
3. Click the on right sub-menu ≡ and click **Add** button at the right top corner of the screen and select **Add Mappings** option.
4. Select the mapping type radio button, i.e. **Source** or **Target**. In our example let us create a new mapping rule for Source, hence select **Source** radio button
5. Enter the **Source Schema**, **Source Table Name**, **Source Column Name** and **Source Data Type** information.
6. Optionally you can also enter **Filter Rules** to filter the data. You can also mention join information using **Join Type** dropdown if needed.
7. Now if you want to continue adding more mappings, check the **Save & Continue…** check box and click **Save** button. Alternatively, just click **Save** button.

If user selects **Target** at step 4, few more information will be required i.e. **Target column Nullability**, **Expression Types** and **Target Expression**

Now newly added mappings will be listed in the grid view of the page.

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.



Mappings for : Employee Dept report

● Source ○ Target

Enter Source Information

Source Type*: Table

Source Group Id: 4

Source Variable *: {{MySQL}}

Source Column Name *: fname

Source Table Name*: Employee

Source Data Type*: varchar(30)

Filter Rule: Filter Rule

Join Type: Select Join Type

Bulk Import of Mappings Using Excel Sheet

We also have the option to define all the mapping related information in one Excel Sheet and import it directly into our QualiDI project. For our example, we will use bulk import by using *ETL mapping sheet*

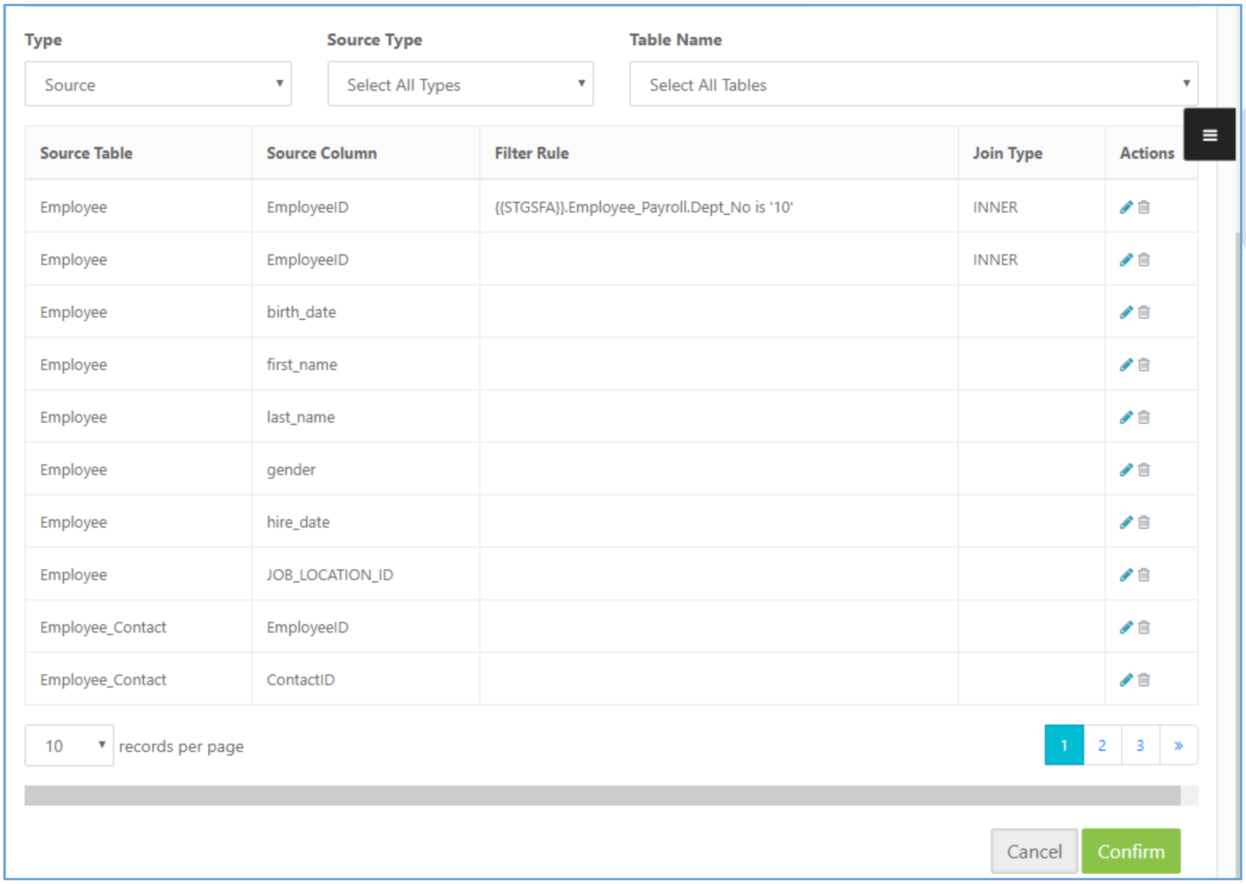
If you see the **Expressions** section at **Target Transformation**, notice that the transformation rules are written in English like language.

To import the Mappings using Excel Sheet:

1. Click on **Mapping Sheet** at the left-hand module navigation slider (If you do see navigation slider, click on 3 horizontal bar icon at top left corner of the screen)
2. Click on **Edit** mapping sheet from the **Mapping List** that we created in steps in *Creating Mapping Sheet for ETL Test-Suite*
3. Click on right sub-menu ≡ to expand and click **Import Mappings**.
4. Browse the excel work book (Excel named as Employee_Address_SQL_MYSQL.xlsx for our example). (Please make sure that your mapping information is in the first sheet.)
5. Now on the **Extracted Data File** dialogue box, select the mapping sheet name from **Mapping List** dropdown, if not already at step 1.
   a. On this dialogue box, you can apply filter according to **Type** (Source/Target mapping) and **Table Names** by selecting the values from respective drop downs.

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

b. Here, you can also delete or modify the mappings by clicking icons next to each mapping.



Type	Source Type	Table Name
Source	Select All Types	Select All Tables

Source Table	Source Column	Filter Rule	Join Type	Actions
Employee	EmployeeID	{{STGSFA}}.Employee_Payroll.Dept_No is '10'	INNER	✎ 🗑
Employee	EmployeeID		INNER	✎ 🗑
Employee	birth_date			✎ 🗑
Employee	first_name			✎ 🗑
Employee	last_name			✎ 🗑
Employee	gender			✎ 🗑
Employee	hire_date			✎ 🗑
Employee	JOB_LOCATION_ID			✎ 🗑
Employee_Contact	EmployeeID			✎ 🗑
Employee_Contact	ContactID			✎ 🗑

10 records per page

6. Click on **Confirm** button at the bottom right of the dialogue box.

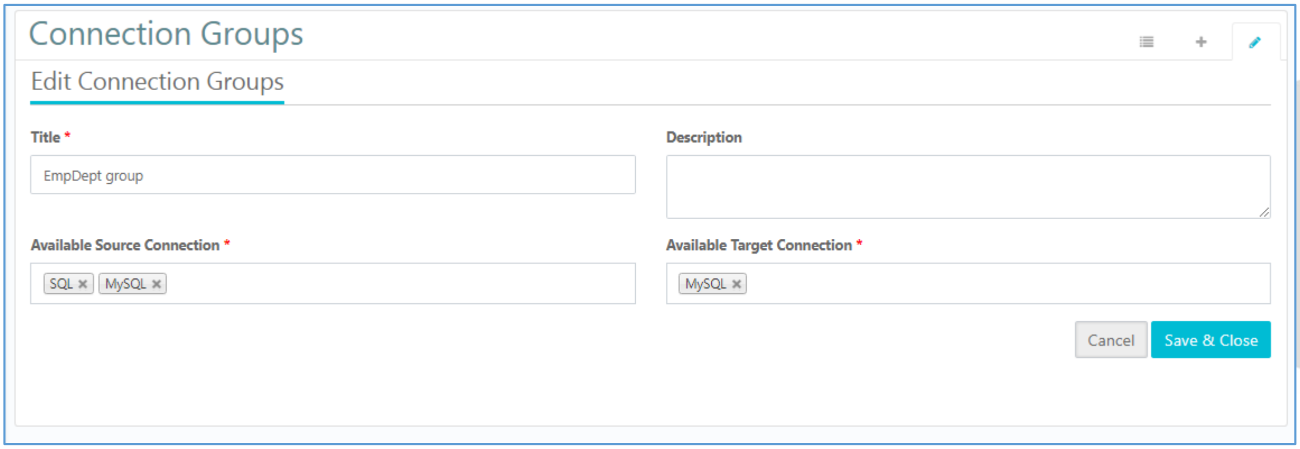
   Now newly added mappings will be listed in the grid view of the page.

5.6. Grouping Source and Target Databases

To bundle and identify the source and target databases from our data base connections we create groups. To create groups:

1. Click on **Connection Group** at the left-hand module navigation slider (If you do see navigation slider, click on 3 horizontal bar icon at top left corner of the screen)
2. Click on **Add** button on the top right corner of the screen
3. Enter **Title** and **Description** in the dialog box.
4. Hit **Save & Close** button.

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.



Connection Groups

Edit Connection Groups

Title * EmpDept group

Description

Available Source Connection * SQL ×, MySQL ×

Available Target Connection * MySQL ×

5.7. Creating Configuration File

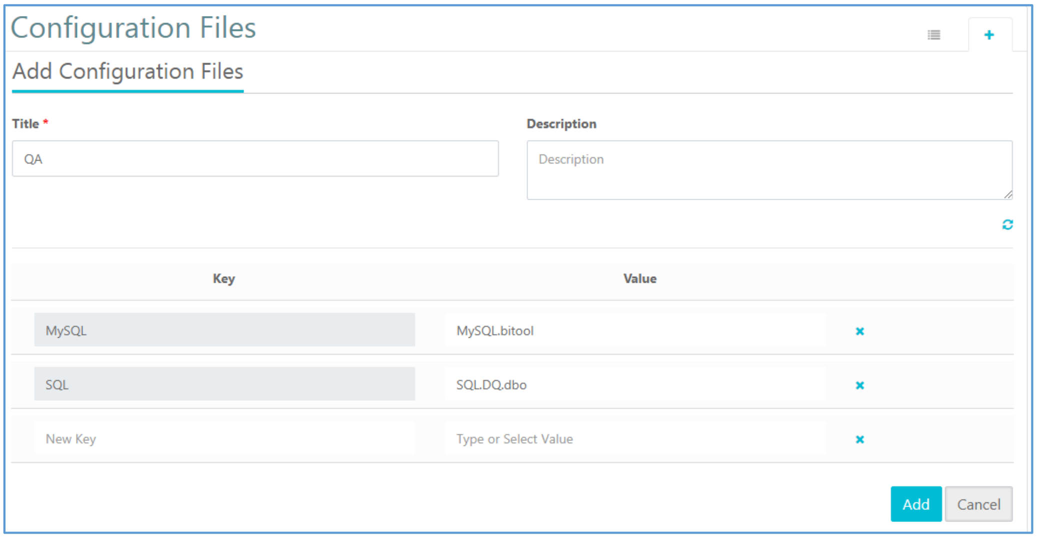
To allow the same set of test cases to be executed on different environments, we create a configuration files.

Configuration file is intelligent to identify the used variables in mappings and make them readily available for you. It also suggests the possible value by referring to available database connections, however, you can add or remove the KEY, VALUES manually as well.

To create new Configuration File-

1. Click on **Configuration File** at the left-hand module navigation slider (If you do see navigation slider, click on 3 horizontal bar icon at top left corner of the screen)
2. Click on **Add** button.

   Give a values corresponding to the Variables which are already present. (Values can be given using auto select or manually typing) These values will be replaced in test cases automatically on selecting configuration file for executing.
3. Click on **Add** at right bottom.



Configuration Files

Add Configuration Files

Title * QA

Description: Description

Key	Value
MySQL	MySQL.bitool
SQL	SQL.DQ.dbo
New Key	Type or Select Value

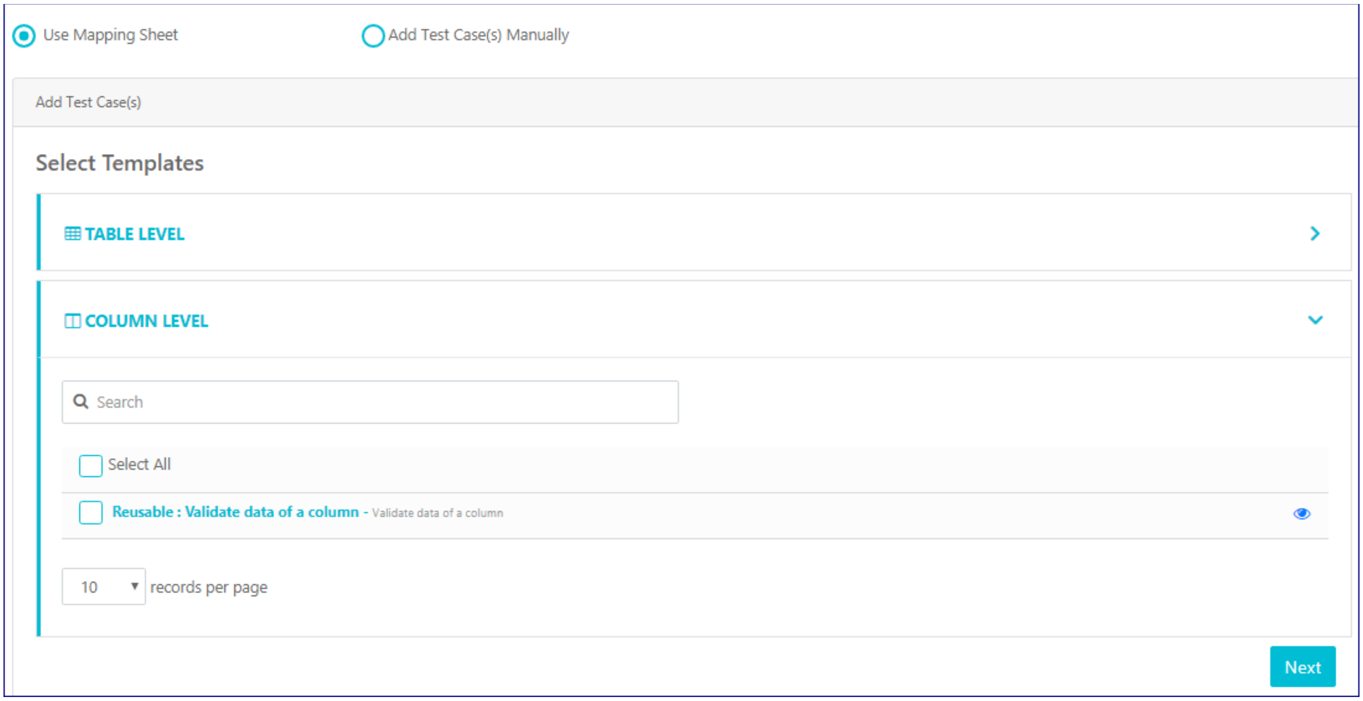
5.8. Creating Test Suite

This is the stage where you can create and execute the Test Cases. System provides two different mechanisms to add test case(s) to your test suite(s).

1. Using Mapping sheet
2. Using source & target test queries directly

To create new Test Suite and add test cases via Mapping Sheet:

1. Click on **Test Suite** at the left-hand module navigation slider (If you do see navigation slider, click on 3 horizontal bar icon at top left corner of the screen)
2. Select Add **New Test Suite**.
3. For straight pulls select the query templates as per desire from the list (at column or table level) by clicking (you can select more than one at a time just by clicking). For our Example, select **Column Level** tests templates, i.e. select **Column Value Null Check** and **Duplicate Records.** If you want to go ahead and select more.



● Use Mapping Sheet ○ Add Test Case(s) Manually

Add Test Case(s)

Select Templates

TABLE LEVEL

COLUMN LEVEL

Search

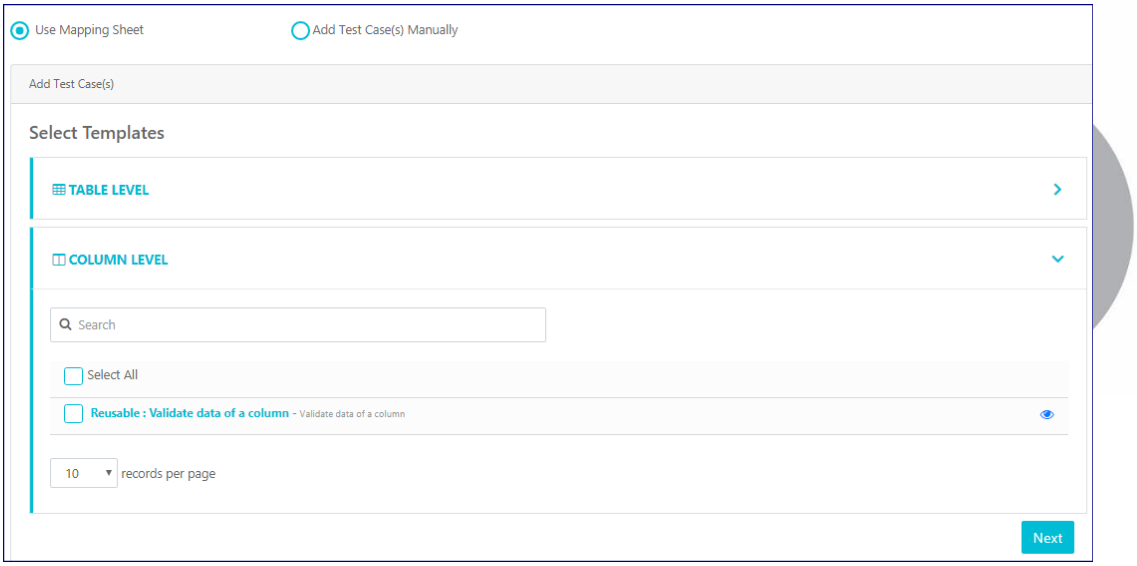
□ Select All

□ Reusable : Validate data of a column - Validate data of a column

10 records per page

Next

4. Hit **Next** button at the bottom right of the screen.



● Use Mapping Sheet ○ Add Test Case(s) Manually

Add Test Case(s)

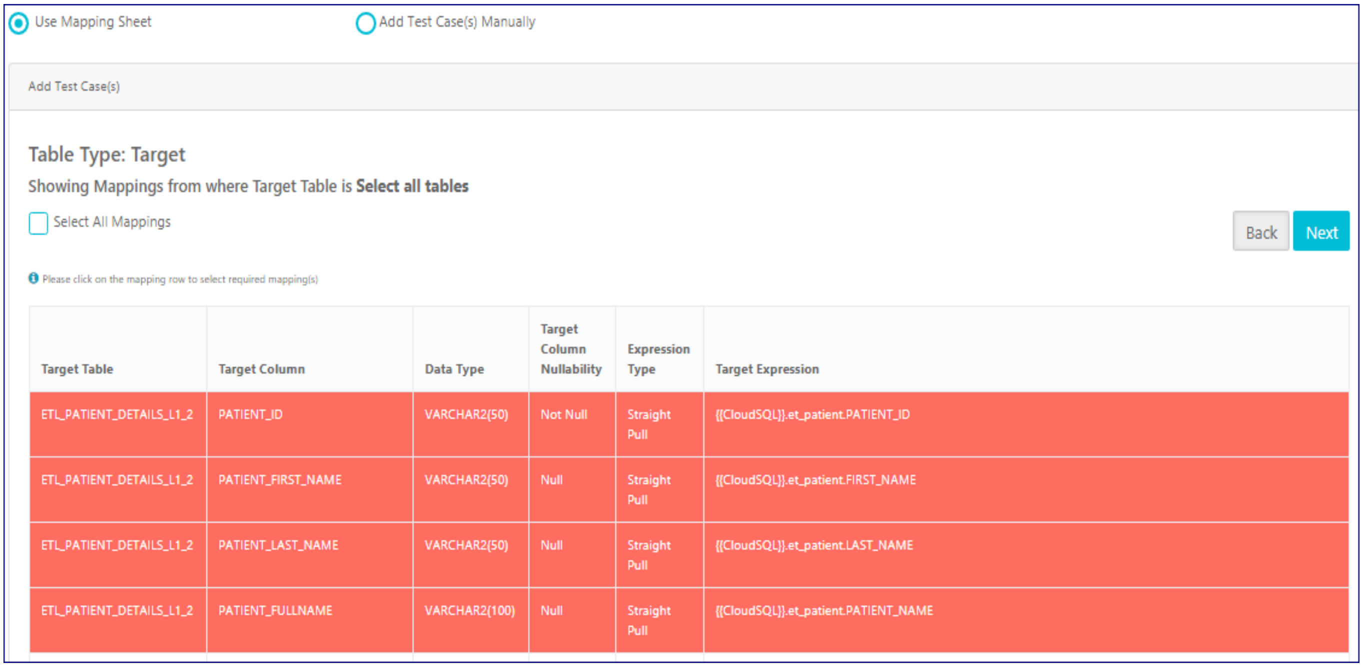
Select Templates

□ Select All

□ Reusable : Validate data of a column - Validate data of a column

10 records per page

5. On the **Rules** dialogue box select **Mapping Sheet** and **Target Table** from the dropdown. For our example, choose **Select All Table** from Target Table dropdown.
6. You can select **Other Options** as per your desire and hit the **Next** button. For our example, let us select both the options.
7. Now you can select Mappings that you want to generate test case (query). From the grid view, you can select more than one mapping at a time. Now hit **Next** at top or bottom right corner. Let select the 4 shown in the screen grab below:



● Use Mapping Sheet ○ Add Test Case(s) Manually

Add Test Case(s)

Table Type: Target

Showing Mappings from where Target Table is **Select all tables**

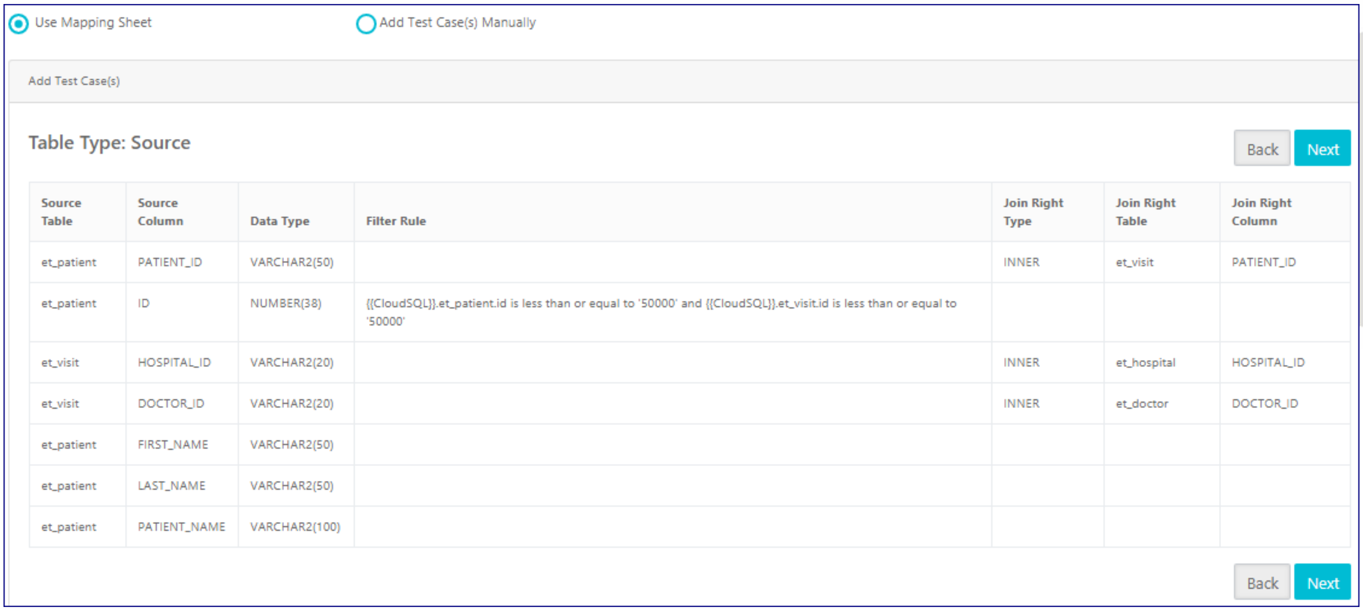
□ Select All Mappings

Please click on the mapping row to select required mapping(s)

Target Table	Target Column	Data Type	Target Column Nullability	Expression Type	Target Expression
ETL_PATIENT_DETAILS_L1_2	PATIENT_ID	VARCHAR2(50)	Not Null	Straight Pull	{{CloudSQL}}.et_patient.PATIENT_ID
ETL_PATIENT_DETAILS_L1_2	PATIENT_FIRST_NAME	VARCHAR2(50)	Null	Straight Pull	{{CloudSQL}}.et_patient.FIRST_NAME
ETL_PATIENT_DETAILS_L1_2	PATIENT_LAST_NAME	VARCHAR2(50)	Null	Straight Pull	{{CloudSQL}}.et_patient.LAST_NAME
ETL_PATIENT_DETAILS_L1_2	PATIENT_FULLNAME	VARCHAR2(100)	Null	Straight Pull	{{CloudSQL}}.et_patient.PATIENT_NAME

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

8. On the following screen, you can see all the corresponding source mappings.



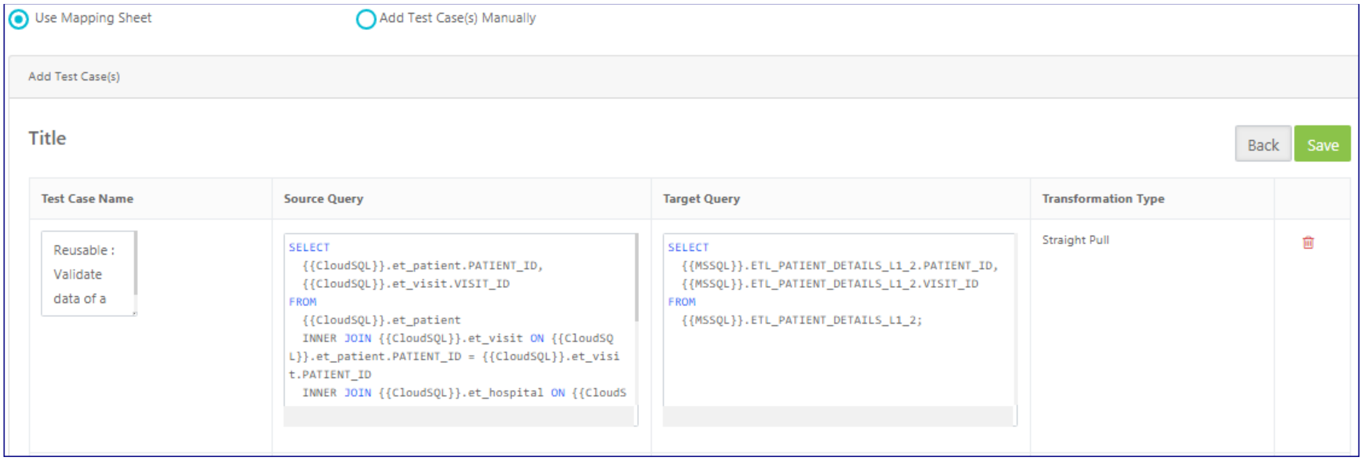
● Use Mapping Sheet ○ Add Test Case(s) Manually

Add Test Case(s)

Table Type: Source

Source Table	Source Column	Data Type	Filter Rule	Join Right Type	Join Right Table	Join Right Column
et_patient	PATIENT_ID	VARCHAR2(50)		INNER	et_visit	PATIENT_ID
et_patient	ID	NUMBER(38)	{{CloudSQL}}.et_patient.id is less than or equal to '50000' and {{CloudSQL}}.et_visit.id is less than or equal to '50000'			
et_visit	HOSPITAL_ID	VARCHAR2(20)		INNER	et_hospital	HOSPITAL_ID
et_visit	DOCTOR_ID	VARCHAR2(20)		INNER	et_doctor	DOCTOR_ID
et_patient	FIRST_NAME	VARCHAR2(50)				
et_patient	LAST_NAME	VARCHAR2(50)				
et_patient	PATIENT_NAME	VARCHAR2(100)				

9. Hit **Next** to generate queries. Here you can edit the **Source** or **Target** Query if you desire so.
10. Now hit **Next** at top or bottom right corner.
11. User will see the Test Suite which contains actual data validation SQL queries (shown below)
12. Upon verifying your tests, hit **Save** button at top or bottom right corner.



● Use Mapping Sheet ○ Add Test Case(s) Manually

Add Test Case(s)

Title

Test Case Name	Source Query	Target Query	Transformation Type
Reusable : Validate data of a			Straight Pull

```
SELECT
  {{CloudSQL}}.et_patient.PATIENT_ID,
  {{CloudSQL}}.et_visit.VISIT_ID
FROM
  {{CloudSQL}}.et_patient
  INNER JOIN {{CloudSQL}}.et_visit ON {{CloudSQ
L}}.et_patient.PATIENT_ID = {{CloudSQL}}.et_visi
t.PATIENT_ID
  INNER JOIN {{CloudSQL}}.et_hospital ON {{CloudS
```

```
SELECT
  {{MSSQL}}.ETL_PATIENT_DETAILS_L1_2.PATIENT_ID,
  {{MSSQL}}.ETL_PATIENT_DETAILS_L1_2.VISIT_ID
FROM
  {{MSSQL}}.ETL_PATIENT_DETAILS_L1_2;
```

This will create the test suite for you.

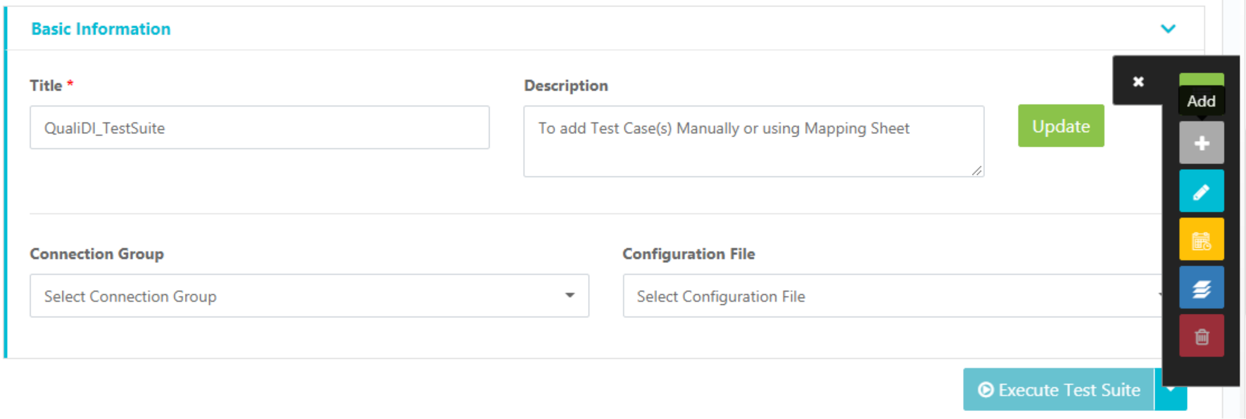
To create new Test Suite and add test cases via Mapping Sheet:

1. Click on **Test Suite** at the left-hand module navigation slider (If you do see navigation slider, click on 3 horizontal bar icon at top left corner of the screen)
2. Select Add **New Test Suite**.

   OR

   Open an existing Test Suite
3. From the floating menu on the right hand side of the page, click on **Add** icon (+)

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.



Basic Information

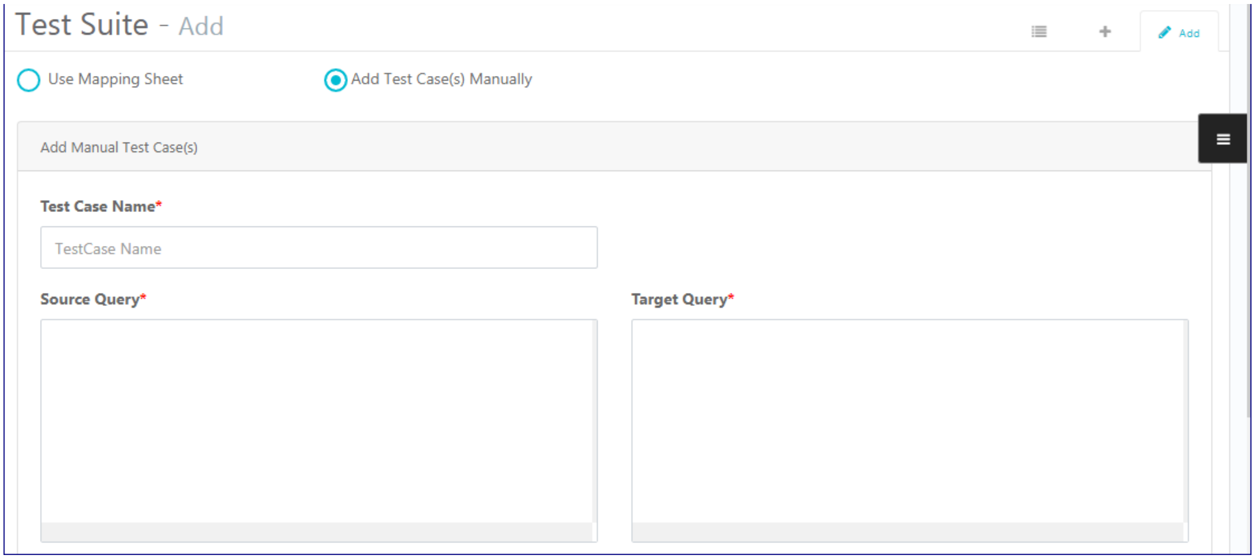
Title * QualiDI_TestSuite

Description: To add Test Case(s) Manually or using Mapping Sheet

Connection Group: Select Connection Group

Configuration File: Select Configuration File

4. **Test Suite – Add** page opens where two radio buttons are present
   a. Use Mapping Sheet
   b. Add Test Case(s) Manually
5. Select **Add Test Case(s) Manually** radio button



Test Suite - Add

○ Use Mapping Sheet ● Add Test Case(s) Manually

Add Manual Test Case(s)

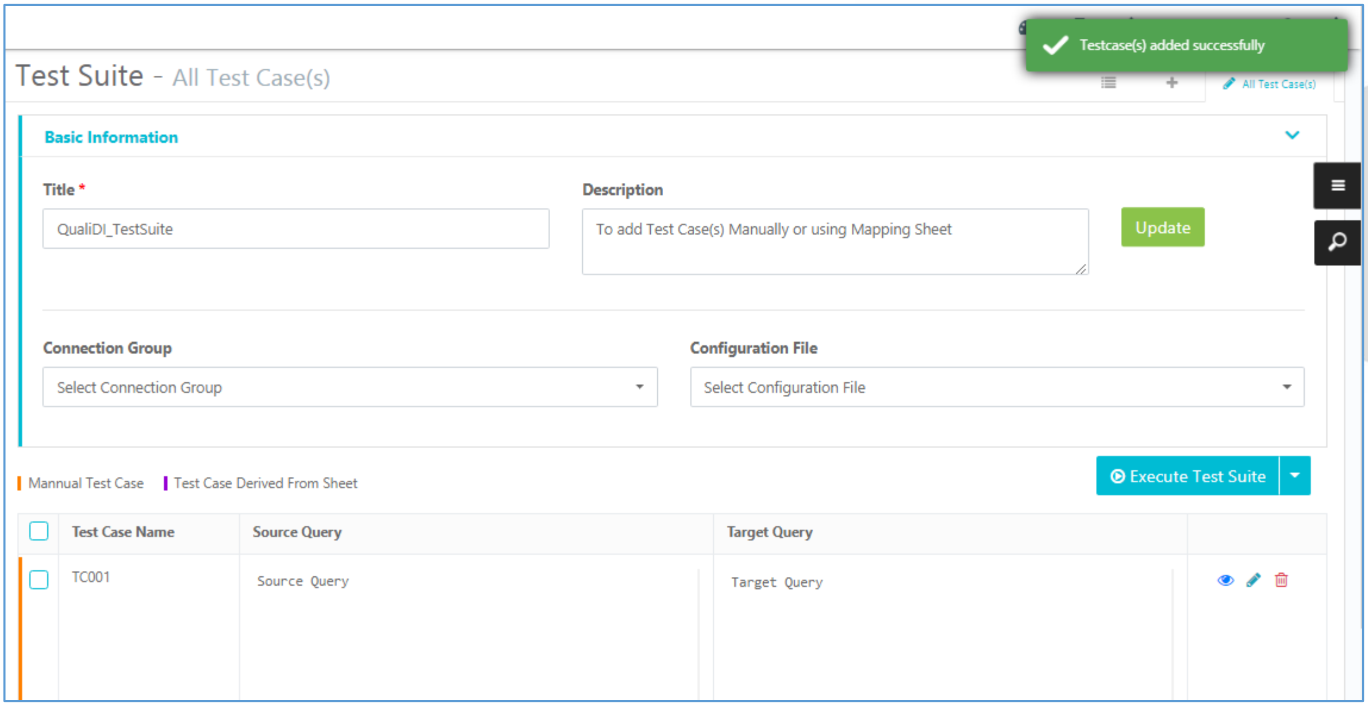
Test Case Name* TestCase Name

Source Query*

Target Query*

6. Enter valid data in **Test Case Name**, **Source Query** and **Target Query** mandatory fields
7. If you want to want to add one Test Case, click on **Save** button
8. If you want to want to add more than one Test Case, click on **Save & Add More** button
9. After adding Test Case(s) successfully, **Test Suite - All Test Case(s)** page opens where you can see respective Test Case(s) enlisted

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.



Testcase(s) added successfully

Test Suite - All Test Case(s)

Basic Information

Title * QualiDI_TestSuite

Description: To add Test Case(s) Manually or using Mapping Sheet

Connection Group: Select Connection Group

Configuration File: Select Configuration File

□	Test Case Name	Source Query	Target Query
□	TC001	Source Query	Target Query

This is how you can add your Test Case(s) Manually or using Mapping Sheet.

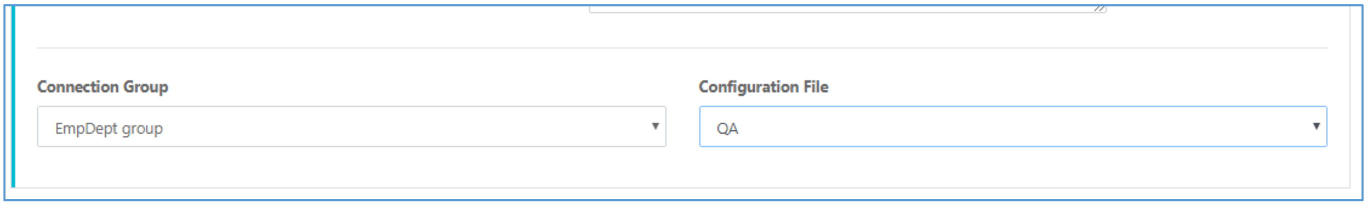
You can also make your manual test cases reusable for different environments by simply using variables in double curly braces to replace connection names. Now define values to these connection names in *Configuration File* and select the same at the time of execution

5.9. Execution of Test Suite/Cases

After creating the Test Suite, we are ready to execute the same.

To execute the test cases:

1. If you have completed the steps at *Creating Test Suite*, you can directly select the test cases by clicking on the columns and hitting the **Execute** Button at top right.
   OR
1. If you have navigated to any other screen, click on **Test Suite->Edit** Select the **Connection group** and **Configuration file** from dropdown.



Connection Group: EmpDept group

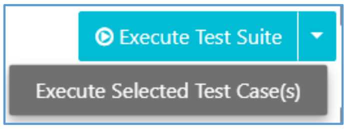
Configuration File: QA

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

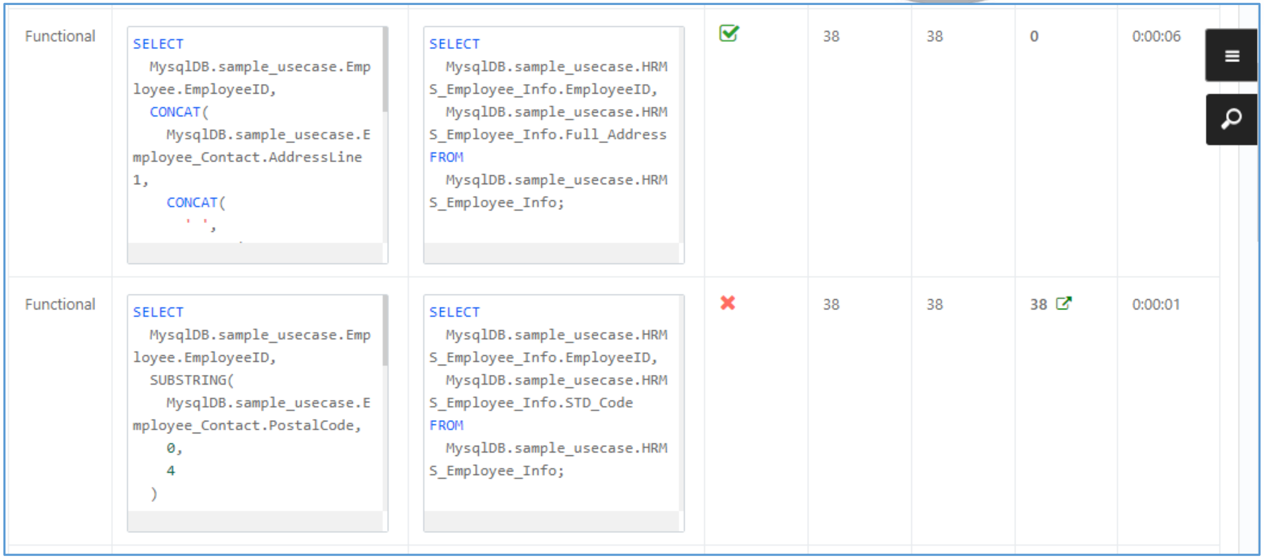
2. Simply hit **Execute Test Suite** button.

   OR

   If you want to execute few test cases, select the required Test Cases just by clicking the rows, click the down arrow next to **Execute Test Suite** button and click **Execute Selected Test Case(s)** option

3. Queries will get executed and mismatch report will be shown.

Functional	SELECT
	  MysqlDB.sample_usecase.Employee.EmployeeID,
	  CONCAT(
	    MysqlDB.sample_usecase.Employee_Contact.AddressLine1,
	    CONCAT(
	      ' ',
		SELECT
		  MysqlDB.sample_usecase.HRMS_Employee_Info.EmployeeID,
		  MysqlDB.sample_usecase.HRMS_Employee_Info.Full_Address
		FROM
		  MysqlDB.sample_usecase.HRMS_Employee_Info;
			✓	38	38	0	0:00:06

Functional	SELECT
	  MysqlDB.sample_usecase.Employee.EmployeeID,
	  SUBSTRING(
	    MysqlDB.sample_usecase.Employee_Contact.PostalCode,
	    0,
	    4
	  )
		SELECT
		  MysqlDB.sample_usecase.HRMS_Employee_Info.EmployeeID,
		  MysqlDB.sample_usecase.HRMS_Employee_Info.STD_Code
		FROM
		  MysqlDB.sample_usecase.HRMS_Employee_Info;
			✗	38	38	38	0:00:01

This will execute the Test Cases; you can also schedule them using **Schedule** button for execution based on event or time.

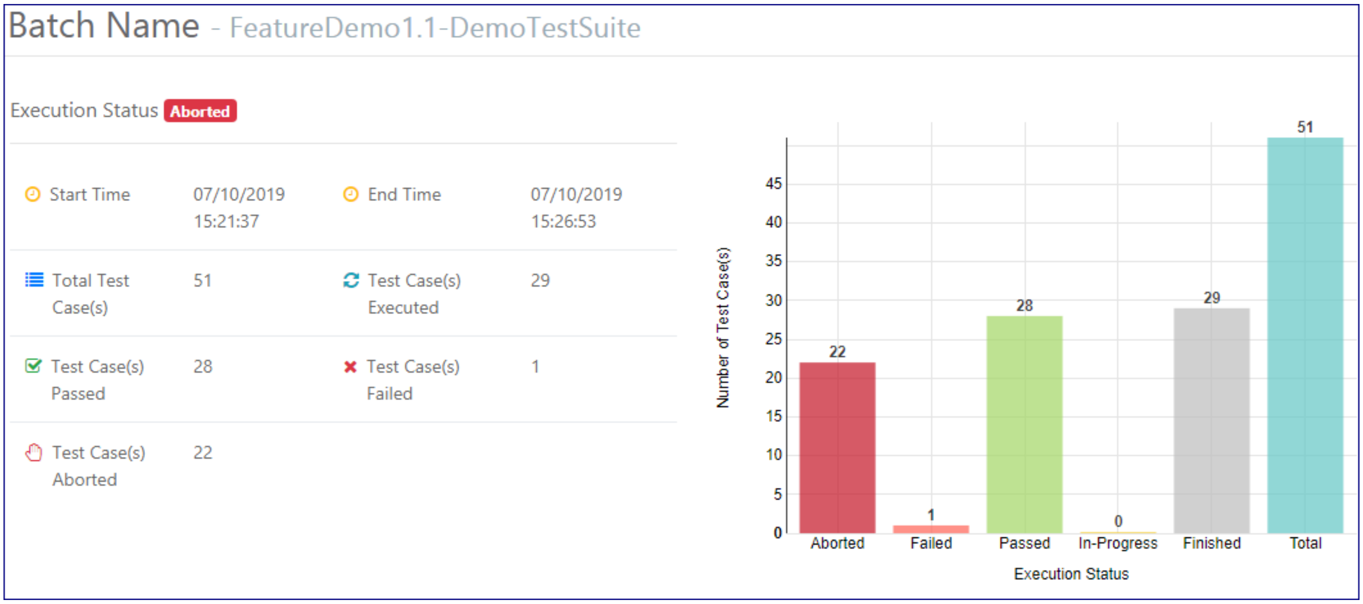
This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

5.10. Reports

Once the test suite is executed, it shows the graphical report for the result of execution along with execution report for all the test queries.

To access the report:

1. Navigate to **Execution History** in the left had drawer
2. Select the one of the desired **Batch History** | **Schedule History** option.
3. Select the desired item to view the execution details



Batch Name - FeatureDemo1.1-DemoTestSuite

Execution Status Aborted

Start Time	07/10/2019 15:21:37	End Time	07/10/2019 15:26:53
Total Test Case(s)	51	Test Case(s) Executed	29
Test Case(s) Passed	28	Test Case(s) Failed	1
Test Case(s) Aborted	22		

CHAPTER 6. ADVANCED FEATURES

6.1. Workspace

Workspace is a personal space where user can make necessary changes in steps and checks in only the final changes to make it available for other team/project members.

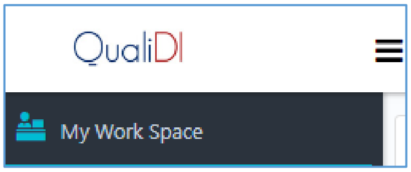
The workspace is created when user selects to work on a project for the first time ever. This workspace is user specific.

To navigate to Workspace-

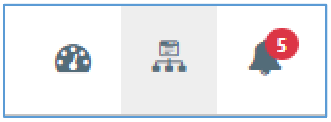
1. Login to QualiDI.
2. If user is logging in for the first time, then after login it will show the pop up for workspace.
3. If user is not logging in for the first time, then pop up for workspace will not appear. Workspace will have the last used project as active.

There are 2 ways to access workspace-

a. Click on **My Work Space** at the left-hand module navigation slider (If you do see navigation slider, click on QualiDI logo at top left corner of the screen)



b. Click on Highlighted Icon at the top right corner



6.2. Central Repository and Item Management

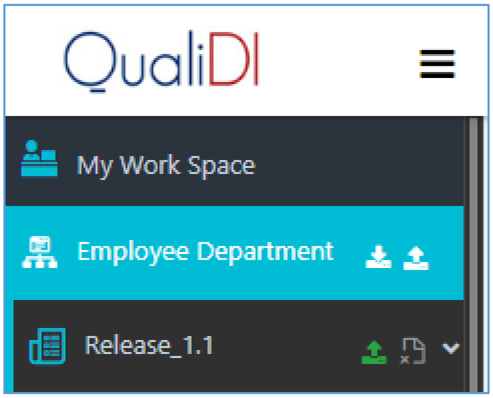
Central repository is the collection where all users check-in their artifacts to make it available to entire team or project members.

Check in – Check out life cycle:

When user Checks-In items, it goes into central repository and it will be accessible to other authorized users by Checking-Out in their respective workspaces. The Check-In function is used to send items to a central repository while Check-Out function used for fetching the items to own workspace from central repository. Discard feature helps user to undo the changes from workspace i.e. revert back to the central repository version.

However, **Save** persists the changes in workspace even if items are not checked-in and project/release is switched.

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.



6.3. Version Control

QualiDI records the changes made to items those can be published to central repository.

QualiDI provides a feature to compare two different versions of an artifact. The differences between two versions are highlighted to identify the changes.

Version comparison is supported for modules-

1. Mappings
2. Test suite
3. Configuration files

Rollback versions:

QualiDI supports Rollback feature for versioned artifacts. If user wants to work on any older version of an artifact, then he/she can Rollback to any version from artifact version page.

6.4. Project and Release Level check-in History

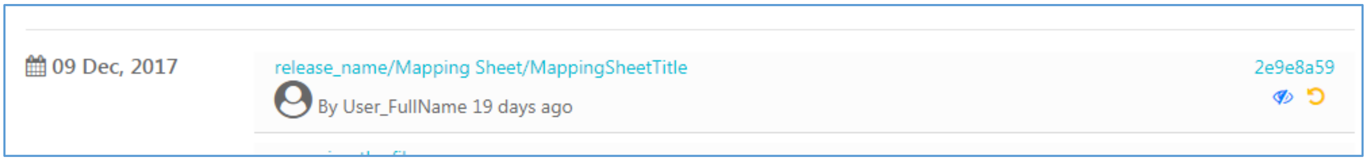
Project and Release level history is supported for modules-

1. Mappings
2. Test suite
3. Configuration files

Once user checks in his/her changes to central repository, the changes get reflected in check-in history as well.

Changes made at Release level will be shown in the **Release Artifact History.**

Changes made at Project level will be shown in the **Project Artifact History**.


09 Dec, 2017

release_name/Mapping Sheet/MappingSheetTitle

By User_FullName 19 days ago

2e9e8a59

6.5. Scheduling Test Suite

Scheduling allows user to execute a test suite at exact time or based on any event. This is used for unattended execution of test suites.

QualiDI supports two different ways of scheduling.

Time based scheduling

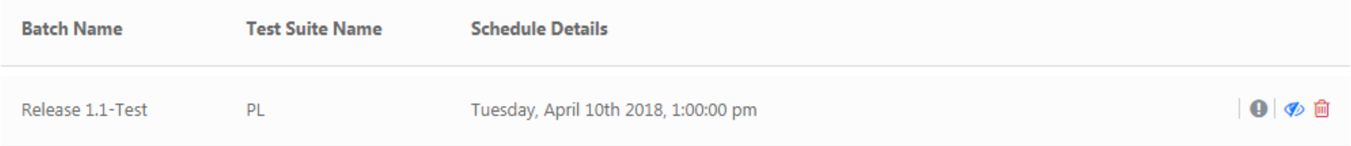
This allows users to schedule the execution of test suite at a given time. Following are the steps for the same:

Pre- requisite:

1. You have a test suite created already.

Steps:

1. Navigate to test suite listing page by clicking **Test Suite** at left drawer navigation under your Release.
2. Click on **Edit** icon next to the suite you want to schedule
3. Select the desired **Connection Group** and **Configuration File** under **Basic Information** accordion.
4. Click on **Schedule** icon from the list of action icons at top right corner of screen
5. Enter the desired **Title** and select the **Time Based** radio button
6. Click on **DateTime** text box and set the desired date and time of execution.
7. Click **Save & Close** button
8. The newly scheduled test suite will appear under **Execution History → Schedule history** option respectively in the left drawer navigation.
9. You can view, delete or see the execution status of items here



Batch Name	Test Suite Name	Schedule Details	
Release 1.1-Test	PL	Tuesday, April 10th 2018, 1:00:00 pm	❗ 👁 🗑

The batch name is prefixed with the release name in History Section, so that, you can identify which release this execution belong to.

Event based Scheduling

This allows users to schedule the execution of test suite to be executed at any event. A URL is generated for the trigger by the system. To trigger the execution simply hit the URL in your browser or via your external application. Following are the steps for the same:

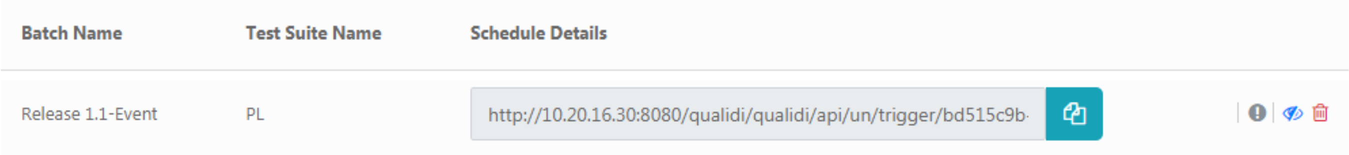
Pre- requisite:

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

1. You have a test suite created already.

Steps:

1. Navigate to test suite listing page by clicking **Test Suite** at left drawer navigation under your Release.
2. Click on **Edit** icon next to the suite you want to schedule
3. Select the desired **Connection Group** and **Configuration File** under **Basic Information** accordion.
4. Click on **Schedule** icon from the list of action icons at top right corner of screen
5. Enter the desired **Title** and select the **Event Based** radio button
6. Click **Save & Close** button
7. The newly scheduled test suite will appear under **Execution History → Schedule history** option respectively in the left drawer navigation.
8. You can view, delete or see the execution status of items here



Batch Name	Test Suite Name	Schedule Details
Release 1.1-Event	PL	http://10.20.16.30:8080/qualidi/qualidi/api/un/trigger/bd515c9b·

9. You can copy this URL and hit it to trigger the execution as many time as required.

This document is proprietary to Bitwise.

The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

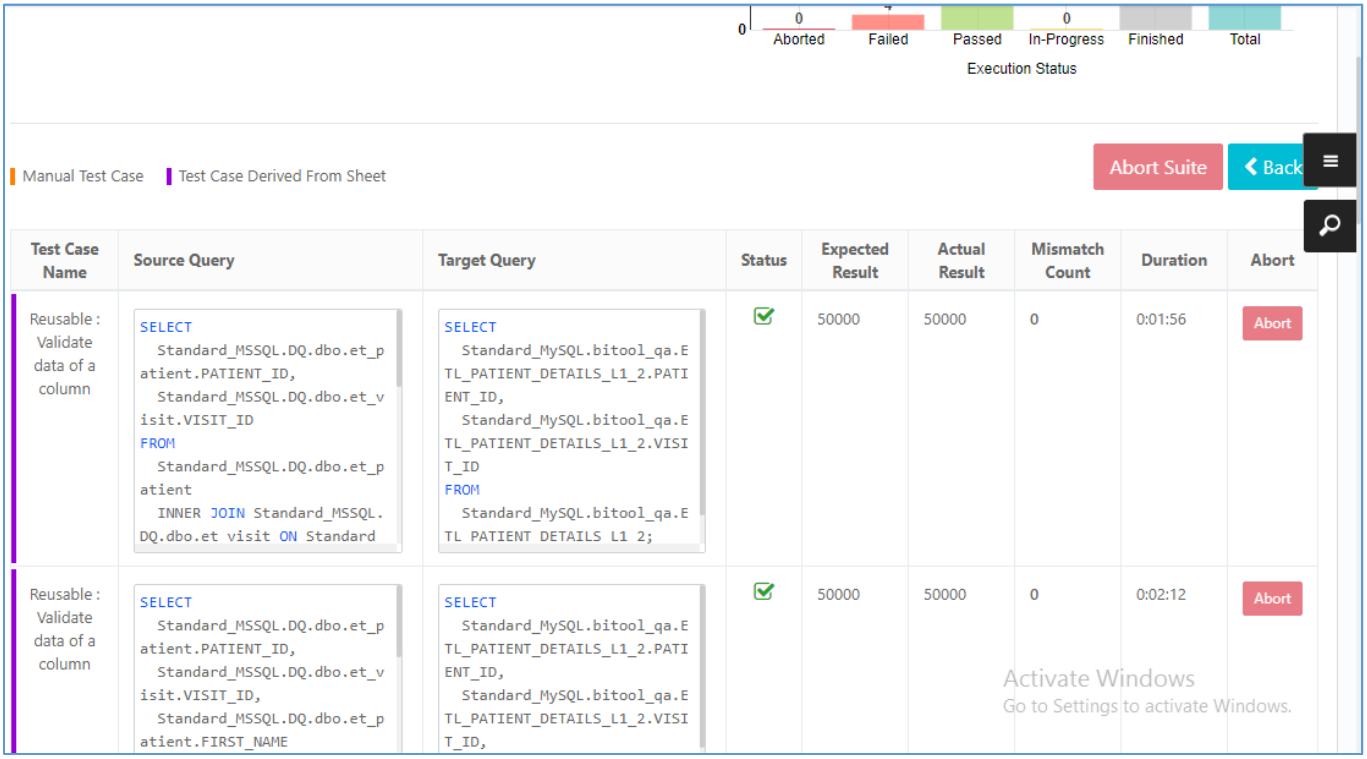
6.6. Abort a Test Suite or Test Case(s)

Abort functionality allows you to stop any ongoing Test Case execution or a batch file which is queued or scheduled for execution.

On following pages, abort functionality is available:

1. Batch Execution Details
2. Batch History – List

Batch Execution Details page:



Test Case Name	Source Query	Target Query	Status	Expected Result	Actual Result	Mismatch Count	Duration	Abort

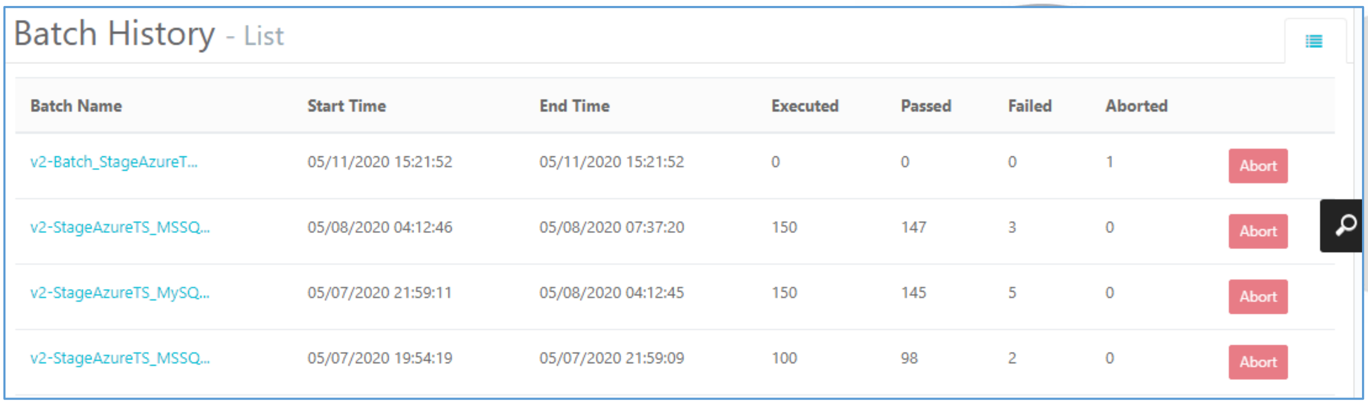
Reusable : Validate data of a column	SELECT
	Standard_MSSQL.DQ.dbo.et_patient.PATIENT_ID,
	Standard_MSSQL.DQ.dbo.et_visit.VISIT_ID
	FROM
	Standard_MSSQL.DQ.dbo.et_patient
	INNER JOIN Standard_MSSQL.DQ.dbo.et visit ON Standard
		SELECT
		Standard_MySQL.bitool_qa.ETL_PATIENT_DETAILS_L1_2.PATIENT_ID,
		Standard_MySQL.bitool_qa.ETL_PATIENT_DETAILS_L1_2.VISIT_ID
		FROM
		Standard_MySQL.bitool_qa.ETL PATIENT DETAILS L1 2;
			☑	50000	50000	0	0:01:56	Abort

Reusable : Validate data of a column	SELECT Standard_MSSQL.DQ.dbo.et_patient.PATIENT_ID, Standard_MSSQL.DQ.dbo.et_visit.VISIT_ID, Standard_MSSQL.DQ.dbo.et_patient.FIRST_NAME	SELECT Standard_MySQL.bitool_qa.ETL_PATIENT_DETAILS_L1_2.PATIENT_ID, Standard_MySQL.bitool_qa.ETL_PATIENT_DETAILS_L1_2.VISIT_ID,	☑	50000	50000	0	0:02:12	Abort

Activate Windows
Go to Settings to activate Windows.

1. When execution is in progress and click on **Abort Suite** button Abort Suite, whole Test Suite is aborted with appropriate message
2. When execution is in progress and click on **Abort** button Abort, that particular Test Case is aborted with appropriate message and Status icon ⚠
3. Once abort action is successful or execution is completed, **Abort Suite** or **Abort** button(s) is disabled accordingly

Batch History – List page:



Batch History - List

Batch Name	Start Time	End Time	Executed	Passed	Failed	Aborted	
v2-Batch_StageAzureT...	05/11/2020 15:21:52	05/11/2020 15:21:52	0	0	0	1	Abort
v2-StageAzureTS_MSSQ...	05/08/2020 04:12:46	05/08/2020 07:37:20	150	147	3	0	Abort
v2-StageAzureTS_MySQ...	05/07/2020 21:59:11	05/08/2020 04:12:45	150	145	5	0	Abort
v2-StageAzureTS_MSSQ...	05/07/2020 19:54:19	05/07/2020 21:59:09	100	98	2	0	Abort

1. When execution is queued or is in-progress, click on **Abort** button Abort, to stop execution
2. Once abort action is successful, **Abort** button is disabled

6.7. Logging Bugs Automatically

QualiDI supports JIRA for automatically logging bugs on the conditions specified by the users.

Pre- requisite:

1. You have JIRA installed.
2. QualiDI server is able to make API calls to your JIRA

Configure Project Settings: Following is the one-time setup at project level to enable auto bug logging

Steps:

1. Navigate to **Project Settings** at left drawer navigation.
2. Click the right sub menu ≡ option to expand it.
3. Click on **Bug Tracker** icon to open one time configuration screen.
4. Select **JIRA** from **Defect Management Tool** drop down.
5. Select your respective JIRA version from the **Version** drop down.
6. Enter the **Domain** on which your JIRA is up and running. E.g. Myjiradomain:20001
7. Enter the JIRA project key where you want to log bugs in **Project Key** text box.
8. Enter the **User Name** and **Password** of one valid JIRA account using which bugs will be logged.
9. Click on **Save Info** button to validate and save the information entered so far.
10. Click on **Get Fields** button. This will fetch all fields of Bug issue type from JIRA for the respective project.
11. Select the checkboxes against the fields that you would like to populate while QualiDI automatically logs the bugs. JIRA configured mandatory fields will be preselected and disabled.

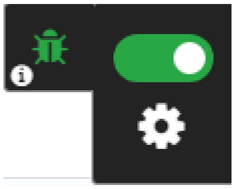
This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

12. Click **Save** button on the **Configure Fields** overlay.
13. Populate the Bug Fields as per your desire using the *system variables* and text.
14. Click on **Save** button. And you are all set with the configuration part.

Enable Auto Bug Logging: Every time you want to automatically log the issues, you need to switch the bug logging at the time of Test Suite execution.

Steps:

1. Navigate to test suite listing page by clicking **Test Suite** at left drawer navigation under your Release.
2. Click on **Edit** icon next to the suite you want to schedule
3. Select the desired **Connection Group** and **Configuration File** under **Basic Information** accordion.
4. Click on **Auto bug Logging** sub menu in the right hand side.
5. Toggle the switch to auto log bug in this execution instance. This will turn the bug icon to green, notifying that the bug logging is ON.



6. You can choose, when to log bugs by selecting options in settings available on gear icon click. Gear icon is visible when Auto Bug Logging is switched ON.

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

6.8. Know About Variables to Populate Bug Fields

To populate the dynamic content in the bugs that will be logged by QualiDI, user can configure fields with the variables that are available in QualiDI

You can use following variable to populate the bug fields in you desired format:

1. **Test_Suite_Name**: Populates the test suite's name
2. **Test_Suite_ID**: Populates the test suite's ID
3. **Test_Case_Name**: Populates the test case's name
4. **Test_Case_ID**: Populates the test case's ID
5. **Test_Suite_Execution_Result**: Populates the test suite's execution results
6. **Test_Case_Source_Query**: Populates the test case's source query
7. **Test_Case_Target_Query**: Populates the test case's target query
8. **Test_Case_Status**: Populates the test case's status
9. **Test_Case_Expected_Result:** Populates the test case's expected result
10. **Test_Case_Actual_Result**: Populates the test case's actual result
11. **Test_Case_Mismatch_Count**: Populates the test case's mismatch count
12. **Test_Case_Mismatch_Report**: Fetch and populates entire mismatch table as in the UI
13. **Test_Case_Duration**: Populates the test case's time execution duration
14. **Test_Suite_Execution_Report_Link**: Will have entire batch history detail page content
15. **Test_Suite_Start_Time**: Populates the test suite's start time
16. **Test_Suite_End_Time:** Populates the test suite's end time
17. **Test_Suite_Total_Rules**: Populates the total number of test case queued for execution
18. **Test_Suite_Rules_Passed**: Populates the total number of test cases passed in test suite
19. **Test_Suite_Rules_Failed**: Populates the total number of test cases failed in test suite
20. **Test_Suite_Rules_Processed**: Populates the total number of test cases executed in test suite

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.

CHAPTER 7. WHAT'S NEXT

You have learned how QualiDI helps you to automate ETL testing and collect statistics and information about it in order to test any ETL job according to defined set goals.

You have seen:

- How to use the ETL Test Automation perspective of the QualiDI to test the ETL job using test automation
- How the test reports show the success or failed test cases and how to trace your test cases against each mapping.

Once you succeed with the simple procedures outlined in *Creating and Executing ETL Test Cases*, you can start digging deeper.

To learn more about Bitwise products and solutions, visit www.bitwiseglobal.com

This document is proprietary to Bitwise.
The recipient should ensure that this document is not deconstructed, reproduced or circulated without the prior approval of Bitwise Solutions Pvt. Ltd.