**Infosys**

**QUALITY SYSTEM DOCUMENTATION**

**References**

Hybrid Automation Framework and

Execution Guidelines

***September, 2014***

**INFOSYS TECHNOLOGIES LIMITED,**

**Chandigarh, SEZ**

COPYRIGHT NOTICE

This Quality System Documentation is the property of Infosys Technologies Limited. All ideas and information contained within these documents are the intellectual property rights of Infosys Technologies Limited. These documents are not for general distribution and are meant for use only for the person they are specifically issued to. These documents shall not be loaned to anyone, within or outside Infosys, including its customers. Copying or unauthorized distribution of these documents, in any form or means including electronic, mechanical, photocopying or otherwise is illegal.

Infosys Technologies Limited

Hosur Road

Electronic City, 3rd Cross

Bangalore 560 100

India.

Telephone: (91) (80) 2852 0261

Fax: (91) (80) 28520362

Website: http://www.infosys.com

e-mail: tools\_deployment@infosys.com

**REVISION LIST**

| **Ver.Rev** | **Date** | **Author** | **Description** |
| --- | --- | --- | --- |
| 1.0 | 19-Sep-2013 | Kanwarpreet Singh Khurana(605782) | This document explains requirement and use of Automated batch execution utility. |

**TABLE OF CONTENTS**

[1 Overview](#_Toc248917259) 5

1.1 About this Guide**.........................................................................................................................................5**

[2 Automation Framework Flow](#_Toc248917260) 5

[2.1 Automation Test Approach](#_Toc248917261) 5

[3 Introduction to Framework](#_Toc248917268) 6

[3.1 About the Framework](#_Toc248917269) 6

3.2 Framework Architecture **…………………………………………………………………………………………………………………………7**

3.3 Components in Hybrid Framework **…………………………………………………………………………………………………………….8**

3.4 Silent Features of Hybrid Framework**………………………………………………………………………………………………………….9**

3.5 Directory Structure**………………………………………………………………………………………………………………………………….10**

3.6 Framework WorkFlow**………………………………………………………………………………………………………………………………13**

**4 Creating Scripts……………………………………………………………………………………………………………………………………………..13**

4.1 Prepare/Maintain Function Libraries**…………………………………………………………………………………………………………13**

4.2 Create/Update Object Repository**……………………………………………………………………………………………………………..14**

4.3 Test Script Preparation**……………………………………………………………………………………………………………………………..14**

4.4 Data Sheet Updation**…………………………………………………………………………………………………………………………………16**

# 5 Pre-requisites for Test Execution…………………………………………………………………………………………………………………….16

# 6 Test Execution ……………………………………………………………………………………………………………………………………………….17

6.1 Execution Approach**………………………………………………………………………………………………………………………………..17**

6.2 How to UPDATE EXECUTION FILE**…………………………………………………………………………………………………………….17**

6.3 Execute the Regression Suite**………………………………………………………………………………………………………………….17**

**7 Over All Execution flow………………………………………………………………………………………………………………………………….19**

**8 Benefits………………………………………………………………………………………………………………………………………………………….20**

# Overview

## About This Guide

The objective of this document is to describe how to use Hybrid Test Automation Framework. It provides step-by-step instructions to setup and use the framework to create, maintain and execute scripts.

# 2 Automation Framework Flow

Automation accelerates the testing efficiency. Because of large usage of Regression Test Cases, Automation is identified as the key factor to reduce the manual intervention thereby reducing the testing effort. Automation Suite Run for every release after individual project testing and is upgraded post the completion of the release testing.

**2.1 Automation Test Approach**

This section gives a brief description on the Test Approach for online banking automation.

Figure 1 Automation Test Flow

1. **Automation Feasibility Analysis**: Once System or Project testing is completed, Automation feasibility analysis needs to be done for next release for new added/updated test scenarios.
2. **Develop/Update Function Libraries**: existing Function Libraries i.e. Generic or Application specific are developed / updated.
3. **Update Object Repository**: Shared Object Repository with new Objects for Application Screens are updated
4. **Create New Test Scripts**: New Test Scripts created for the subsequent release.
5. **Request for Test Data**: Test data requirements are identified and Mined
6. **Execute Regression Suite**: Execute Regression Suite after System Testing.
7. **Update/Review Test Results**: Review screenshots and Test Execution report created
8. **Maintain QTP Scripts**: Regression test scripts updated to execute in future regression.

# Introduction to Framework

## About the Framework

The Hybrid Automation Framework is a technology-agnostic automation framework that supports test automation of any application that is built using a technology supported by HP QuickTest Professional. The Framework is fully integrated with HP Quality Center for Test Cataloguing and Test Execution.

We can easily generate our test scripts using Function libraries and Shared Object Repository. The Framework supports the following technologies, and can easily be extended to support others.

* Web
* Web/Java
* Web/.NET
* Mainframe

## Framework Architecture

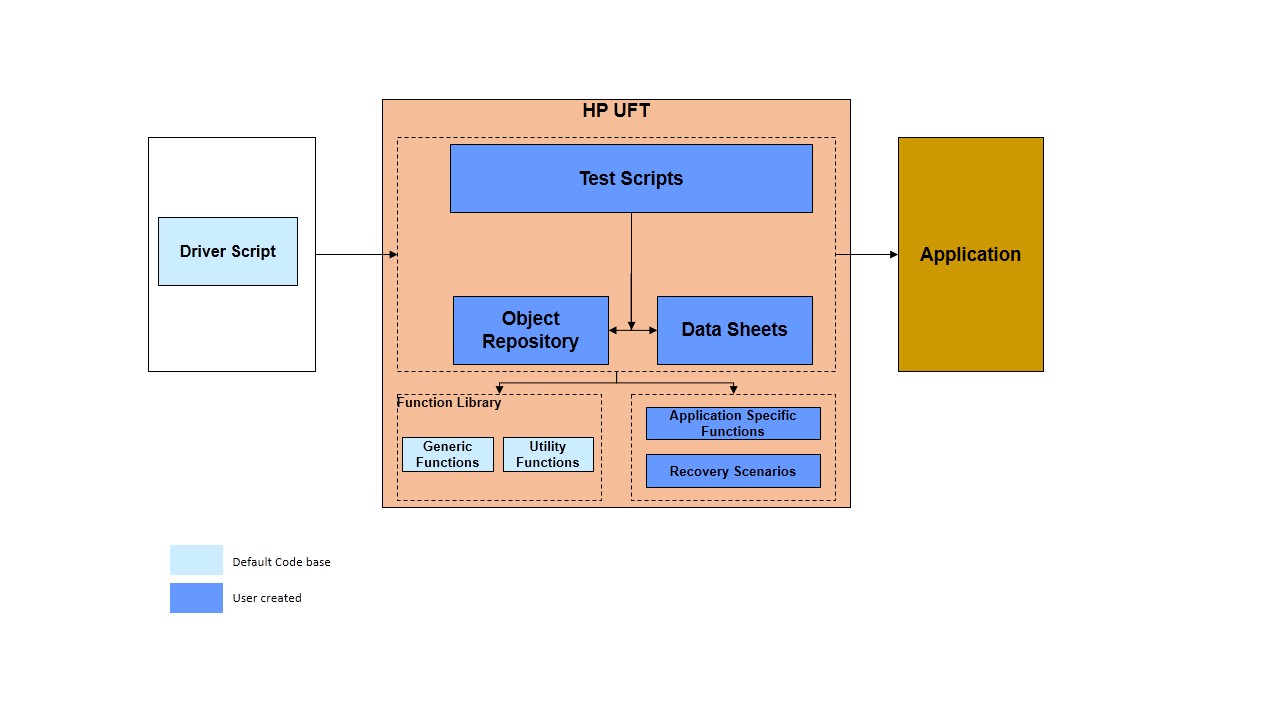
This Framework is designed to provide high reusability of scripts developed using the Framework. 

Figure 2 Framework Architecture

***3.3 Components Used to create Hybrid Framework***

Below is the list of all the components that will be used to create the hybrid framework. A short description accompanies each component.

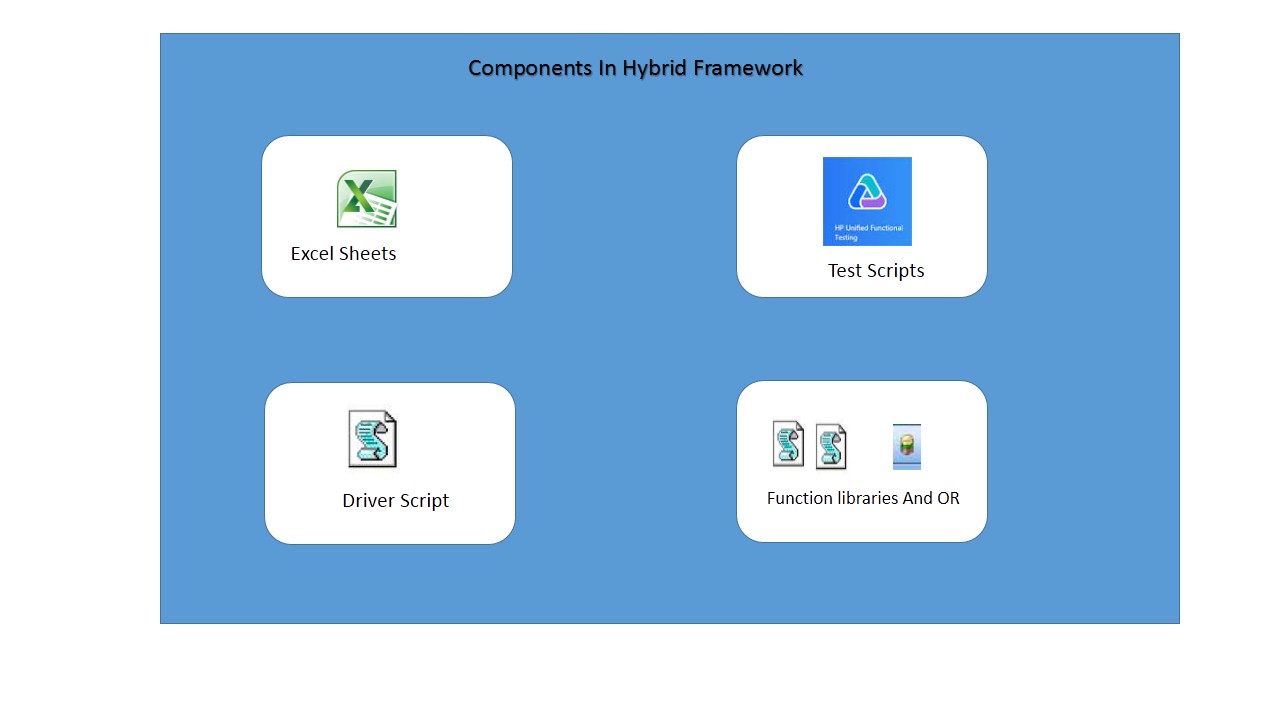


Figure 3 Components in Hybrid Framework

Hybrid Framework has the following components:

1. **Driver Script** : The Driver Script will act as the starting point for running the test scripts. How it would work is like this – There will be an excel sheet which will contain the list of all the test cases.

If you want to run the test scripts, you will need to double click to Driver Script. So when you click on the button,driver script will open QTP and load the first test case to it. It will then instruct QTP to execute the test script.

The same procedure will be followed for all the test cases that are available in the excel sheet. The driver script will use QTP AOM (automation object model) concepts to interact with QTP.

**2) Excel Sheets:** Excel sheets will be used extensively by the hybrid framework that we would create. Excel sheets will be primarily used to store the following information -

* **a) Test Case List:** As mentioned above in point 1, a list of all the test scripts will be maintained in excel sheet. This will be used for batch execution.
* **b) Test Data:** All the test data would come from excel sheets. No test data will be hard-coded anywhere in the test scripts.
* **c) Test Results:** Excel sheets will be used to provide a summarized result of batch execution. Here, the excel sheet will simply contain the list of test cases that were executed together with the information about whether the test scripts passed or failed.

3) **Test Scripts**: These are the actual QTP test scripts. For each scenario that you would automate, you would need to create a separate QTP test script.

**4) Function Library:** This hybrid automation framework will use two separate function libraries. The first function library will store all the generic functions (application independent functions). Some examples for this can be functions to click on a link, to enter value in a text field etc.

The second function library will contain application specific functions. Functions such as login, logout etc will be a part of this second function library.

**5) Object Repository:** All the object definitions will be stored in the object repository. There will be minimal use of descriptive programming concepts.

### Salient features of the Hybrid Framework

Following are some of the important features that you will see in this hybrid framework.

**1)** This framework will be **optimized for batch execution**. This is what most good frameworks are all about. With this framework, you will be able to easily select the test cases that you want to execute in batch run.

**2)** To execute your test scripts, you would not be required to open QTP. This would be taken care of entirely by the driver script (excel macro).

**3)** This framework will provide the test execution results at two different levels – summarized results and detailed results.

**4)** The test execution results from multiple runs **would not be overwritten**. Each test execution result will be appended by date-time stamp so that it can be referenced whenever required.

### 3.5 Directory Structure

Automation suite primarily contains the test cases in batch sheet. The Hybrid Automation Framework follows the Folder Structure as given below:

<**Root Folder**>

**Data Files**

**Object Repository**

**Functions**

**Driver Script**

**QTP\_ResultFiles**

**Results**

**ScreenShots**

**Test Scripts**

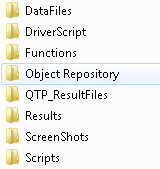


Figure 4 Automation framework folder structure

**Data Sheets**

* As hybrid approach is followed, the test data for automation scripts has been parameterized. This would allow data independency of the scripts and easier maintainability.
* The test data is stored externally in the form of excel sheets and the script will read the data from excel sheet during execution.
* The excel sheet format is designed to fetch the test data automatically from Data Base using the dynamic Queries and either populate in the spread sheet or can be effectively managed in the memory during runtime.

Driver Script

Automation Batch Driver script allows end user to run automated QTP scripts through an excel file.

* It is designed in such a way that end user needs to input the test cases name and their location one time only. Then tool will provide below options and advantages to the user
* Update the Excel sheet for column “ExecutionRequired” with Y or N for the specific test case which needs to be executed
* Update the result folder location in column “ResultDir” where test results needs be captured. If this field left blank, tool will automatically create a results folder in test scripts directory

**Functions**

* Contains Generic (Re-usable) Functions that can be used across particular application.
* Contains Application Specific Functions that can be used for repeated scenarios.
* Common Scripts separates much of the programming work from the actual testing steps so that the test steps can be developed earlier and can often be maintained with only minor updates, even when the application or testing needs undergo changes significantly.

Guidelines for writing a New function:

* Always the function should be unique and reusable set line of code.
* Specific Parameters should be provided with Function definition.
* List of All Functions is listed in the attached Excel.



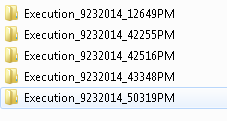
**Object Repository**

* Object Repository is the collection of test objects in the AUT.
* It will be associated to the test script manually during Design Stage.

Name of the object repository used in Demo is ‘ObjectRepository.tsr’

**QTP\_ResultFiles**

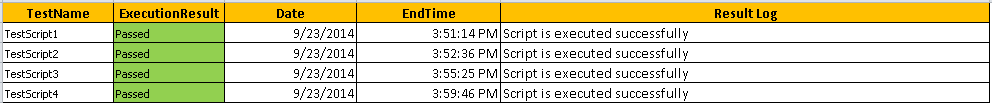
QTP Test Results (.xml) files would be saved in this folder with every New Execution run. It is easy to maintain and view for future reference to validate QTP results. A new folder will be created for every execution with name of the current date and time Combination.



**Results**

In Results folder Excel file is placed with the list of all Test cases of Automated Suite and will be updated with status after execution is completed.

Test case status will be shown as given below



**ScreenShots**

User can place all the Step execution screenshots in nthis folder. User can use these screenshots for future reference. We are using two funtions to capture screenshots:

* **StepScreenShot** : This function is used  to capture the screenshot for every Step during run time
* **ScreenShots** : This function is used  to capture the screenshot during run time with status Fail or Pass

**Test Scripts**

* Contains the automated steps to be carried out for execution of a test case.
* The test script will be in the form of a QTP test that gives the call to common functions in the library folder under respective application as well as to generic functions / web functions stored under the utilities folder.
* It is saved as a QTP script in the test plan in QC and will get executed from the test lab after creating the test suite.

Following are the guidelines we need to follow while writing the test cases.

* Use specific functions for each and every Action with specific parameters i.e. Browser Name, Page Name, Control Name should be the exact names which displays in Object Repository

## Framework Work Flow

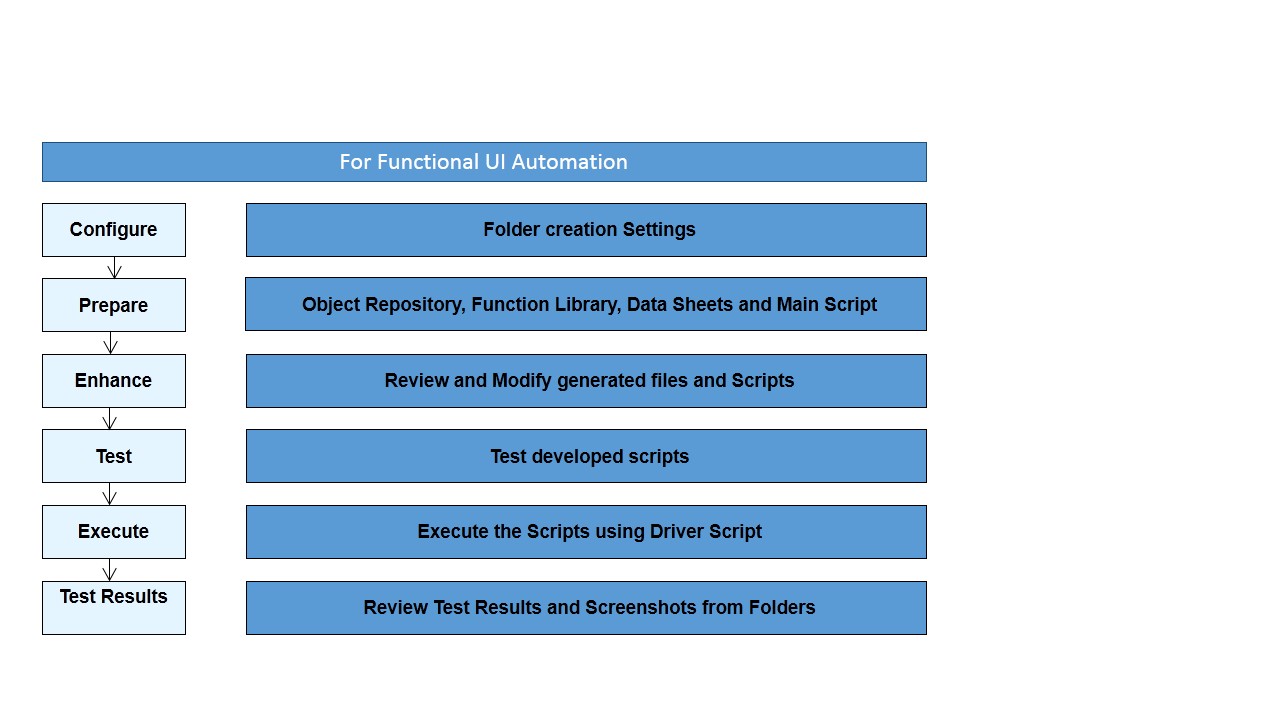


Figure 5 Framework Work Flow

***4. Creating Scripts***

After creating Folder Structure for Automation suite, the next step is Test Script Preparation. We need to prepare Test Scripts or converting all manual steps into Automation test script.

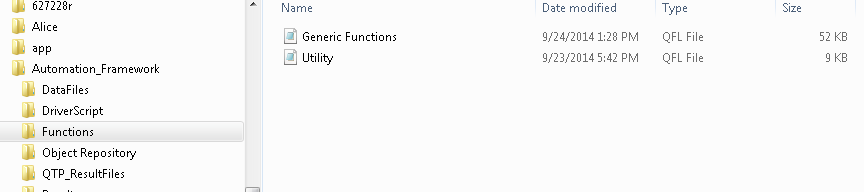
**4.1 Prepare/Maintain Function Libraries:**

Create/Update generic Functions in the Function library placed in the below path :

“**D:\Automation\_Framework\Functions**”

* **Generic Functions**: Contains Generic (Re-usable) Functions that can be used across particular application.
* **Utility Function** : Contains Functions that can be used at the time of execution such as loading OR or function libraries if not associated before execution.
* **Application Specific** : Contains Application Specific Functions that can be used for repeated scenarios.

All Function Libraries must be placed at the path below shown:



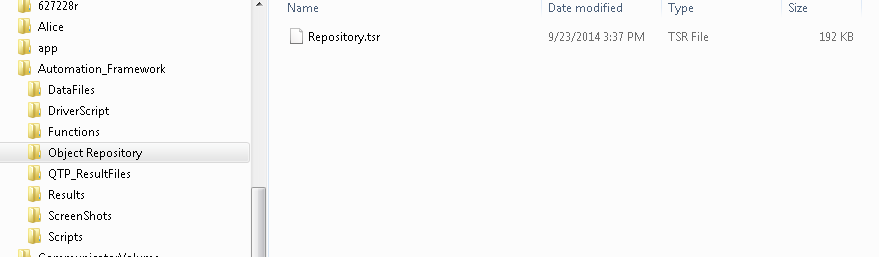
All Function Libraries must be placed under this Folder

**4.2 Create/Update Object Repository:**

We need to make sure that the Object Repository is placed in the required location (as mentioned in the fig). The default location of the Object Repository is within the Object Repository Folder in the Folder Structure.

Location of Object Repository:

**D:\Automation\_Framework\Object Repository**



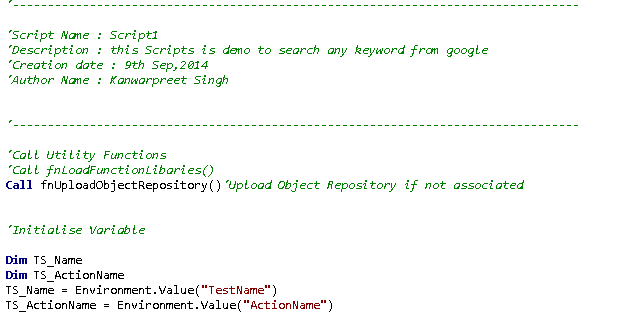
Folder “Object Repository” is created in the directory Structure

Shared Object Repository must be placed under this Folder

**4.3 Test Script Preparation:**

**Steps to Create A New Script:**

1. Provide Name, Description , Creation date and Author at the top of the script as best practices to maintain the script records.



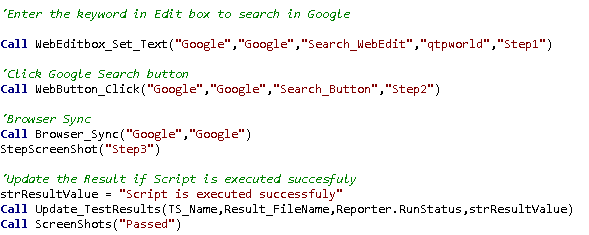
Get the Values from Env. Variables

Initialize Variables

Call Utility Functions

Script Decription

1. Call Utility Functions to upload Function libraries or Object Repositories at run time if no Shared Object Repository is not associated with Test Script.
2. Initialize variables such as TS\_Name and TS\_ActionName in the main script for Test Case name and Test Case Action name.
3. Call Generic Functions from the Function library (Generic\_Functions.vbs) with all specific parameters need to be passed.
4. We don’t need to write the line of code to do any specific Action related to Objects such as Click , Set Input, Select Value etc. We need to call Functions with parameters.



Call Function to take screenshots of the current screen

Pass Arguments in the Function Call

**4.4 Data Sheet Updation:**

After preparing New Test Script, add the name into the sheet “” placed in data Sheets Folder.

**ScriptExecution**: This sheet will contain all the test scripts and refer the same after execution to view test results “**Passed**” or “**Failed**”.

**Script\_Results**: Contains the test scripts name of Regression Suite and updated with Column column “**ExecutionRequired**” with **Y** or **N** for the specific test case which needs to be executed

**5. Pre-requisites for Test Execution**

* User should have basic knowledge of QTP.
* User should have basic knowledge of Excel.

# Test Execution

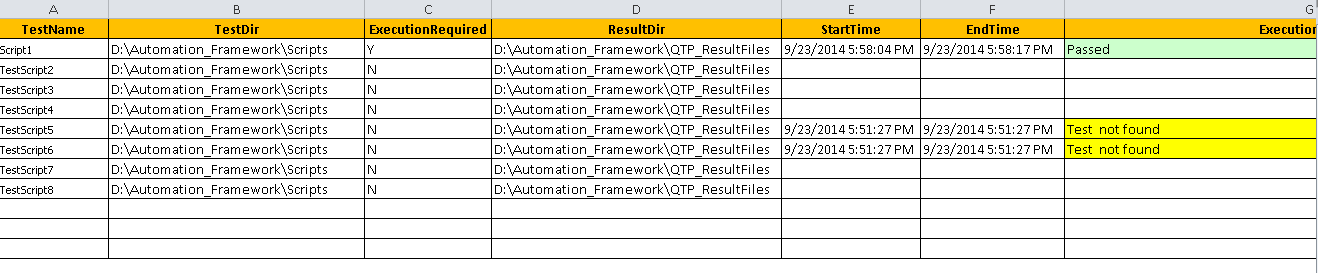
The steps for running the tests in the automation suite are as given below:

**6.1 Execution Approach:**

* Driver Script is created through VBScript, which will execute the desired scripts listed in given Excel in a batch.
* Selection of test cases for execution is parameterized in Excel sheet and in very simple format.
* It solved the problem to get consolidated results and execution status as both are provided in a single excel file.
* QTP execution results will be consolidated at one place and user can track the results from a single location to check if there is any failure in any particular test.
* Users can change the location of Excel and VB code as per their preference.
* Tool is using the EXCEL File to contain all information which is easy to maintain and edit.
* User just needs to double click/run on “AutomatedBatchExecution.vbs” file after updating excel.

**6.2 How to UPDATE EXECUTION FILE**

* Open **ExecutionFile.xls**



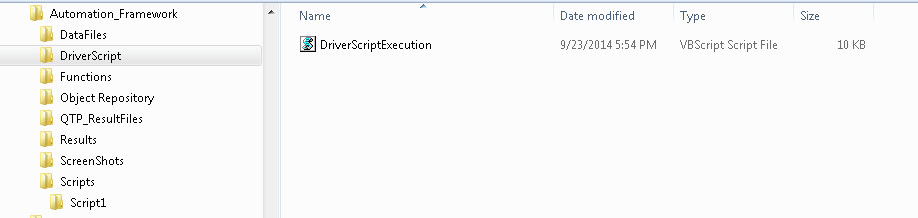
* Under “**TestName**” column enter the automation test scripts name.
* Under “**TestDir**” column enter the location of Automation Test Scripts mentioned in “**TestName**” column.
* Update “ExecutionRequired” column with either “**Y**” or “**N**”. “**Y**” for the scripts which needs to be executed and “**N**” for rest of the scripts.
* Update “**ResultDir**” column with location where user want to store execution results, if left blank this tool will save the results at the path updated in “**TestDir**” column.
* Click Save and then close the file.
* **Make sure the ExecutionFile.xls is closed before starting the driver script**.
  1. **Execute the Regression Suite**

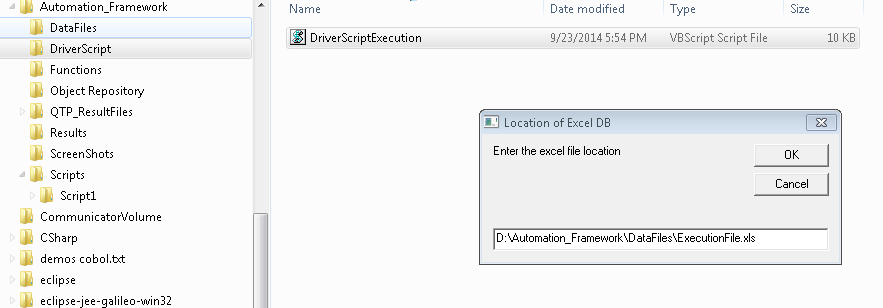
Steps to Execute Test Scripts:

**Step1**: Open the **ScriptExecution.xls** placed at path “D:\Automation\_Framework\DataFiles”.

And mark “**ExecutionRequired**” column as “**Y**” for all test scrips which needs to be executed and close the sheet after saving the changes.

**Step 2**: Double click on “**DriverScriptExecution.vbs**”. It will ask to enter execution file .location which is updated above. Enter the complete path with name.



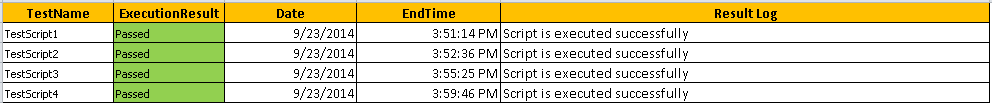


**Step 3**: Click on OK button on pop-up and wait for process completion..

**Step 4**: After execution, the execution file will look like as below,

**Step 5:** If the test result is failed for any particular test case, we can see the results from the sheet ”**Script\_Results.xls**”

**Step 6:** We can view the results with all information i.e. Test Results and Test Descitopn in the sheet. The Test Result file will look like as below:



**7. Over All Execution flow:**

Double click on “**AutomatedBatchExecution.vbs**”, it automatically launches the QTP and starts the execution.

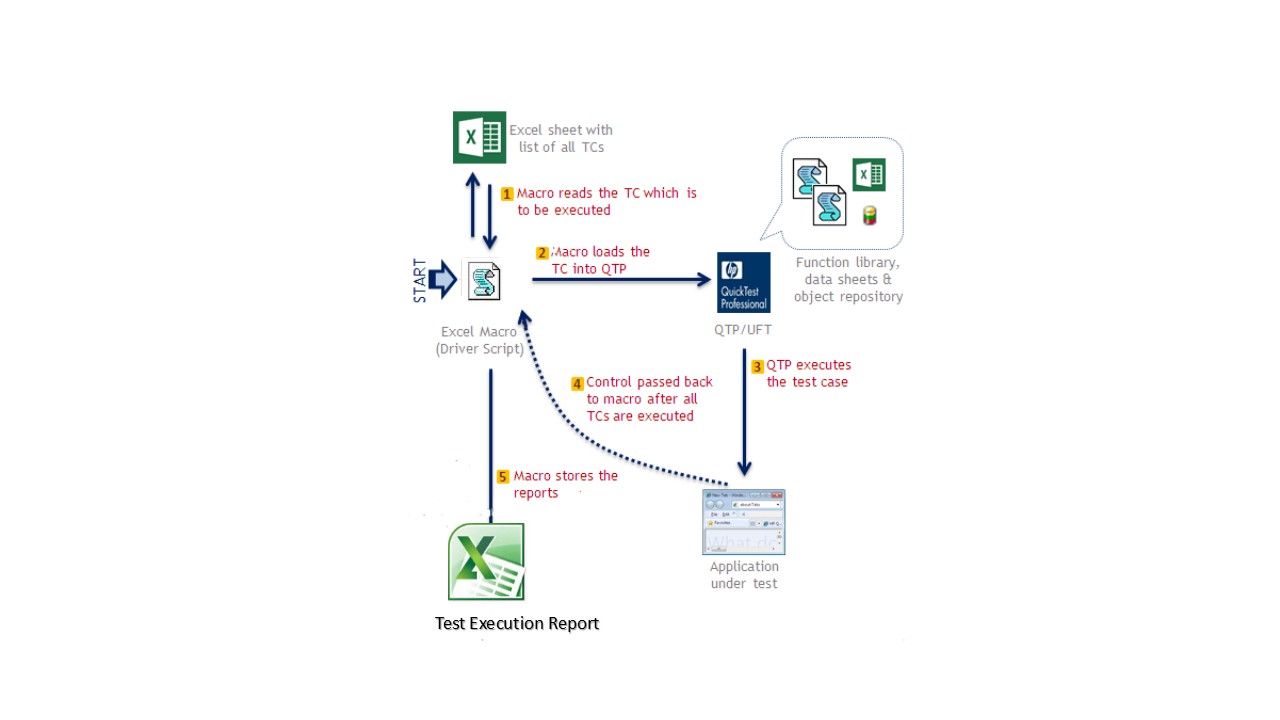


Figure 6 Overall Execution Flow

**Summary View:**

The overall flow is:

1. Automated Test Script is executed from Driver Script.
2. Test Scripts invokes QTP tool to run Automated Scripts.
3. Automated tool QTP is available on desktop from which test is executed.
4. Each test script will utilize the framework and shared resources available in Directory Structure..
5. Test scripts are executed and the results are stored in Excel Sheets in Data Sheets Folder.

**8. Benefits:**

* New applications can be included in the current automation suite with minimal modification to the current Automated Library Scripts
* 100% reusability with minimal maintenance for QTP code.
* Reduced effort for the new applications being added after framework is stabilized as most of the library scripts will be reusable
* Common Scripts can be used to reduce the programming effort thereby making maintenance easy
* Dynamic test data population
* Major cut-down in effort for manual intervention and cost due to automation.
* Reduced turnaround time for testing, facilitating more frequent releases.
* Processes followed reduced rework time due to minor changes
* Usage of Quality Centre gives the flexibility of creating multiple test environment for different applications