

---

## QTP AUTOMATION FRAMEWORK DEVELOPMENT AND BEST CODING PRACTICES

---

**Lal Singh Dhaila**

SRM University NCR Campus

Automation Tester, Cognizant Technology Solutions 2015

---

**ABSTRACT:** Automation is one of the basic needs of the developing countries and the industries to increase the profit and decrease the cost that have been used to create a product. Among the various tool one tool is Quick Test Professional (QTP) that have been widely used to automate the application which are windows based application. QTP provides a medium to automate the application by storing the properties of the object that have been visible on the window application.

---

### INTRODUCTION

This document states the need of the use of QTP for automation development process and selection of the framework that can be used to automate the application. There are wide variety of tool that can be used to automate the application that is based on the window application. There are wide variety of tools that can be used for desktop application purpose some of them are AutoIT, Winium, TestStack.WhiteLDTP, Sikuli, UFT\QTP. QTP is widespread choice for automation as it supports automation development of desktop application that have been developed in many different languages. QTP provides a flexible development environment that when integrated with HP ALM, provides a robust environment for automation development. Along with automation of desktop application UFT provides the add-on of automating the web based application. □

QTP is enough capable of automating the scripts that have been developed using different application one of the best application is SAP system automation. To automate the application in have to buy the license for using the HP product. There are three different ways in which a customer can buy the license these are mentioned below:

1. Concurrent License
2. Seating License

Concurrent license is used for personal use and can be purchase from the HP. Using this kind of the license on will be able to use the tool only in the one system in which the permission have been assigned. Before purchasing the tool one can use the trial version which will be valid for the next 30 days and after that user have to pay for the tool to use it further? While using the trial version one will not be able to use the ALM connectivity benefit.

Seating license is used in large corporations where the requirement of use of the QTP is large. This is also known as floating license. In this type of license we use to install concurrent license server at the local workstation. With help of this concurrent license anyone in that local network can connect using the IP that have been provided by HP team.

Below is screen shot how and where to put IP to start the use of HP UFT tool.

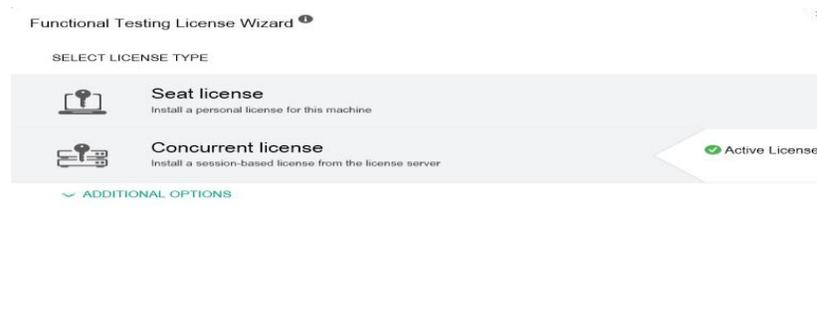


Fig. 1 ALM UFT/QTP license type



Fig. 2 HP UFT concurrent License option

In above screen shot the user have to provide the License IP that have been provided by HP and Port number that can be used to access the UFT and its features.

## QTP AUTOMATION FRAMEWORK SELECTION

QTP supports different kind of automation framework that using which one can easily automate the application that have to be to be automated. The selection of the automation framework depends on the several factors. Some of which are explained below:

1. Complexity
2. Functionality
3. Time Period
4. Run Frequency

Complexity of the application implies the extent to which application is complex.. Selection of the framework depends on the complexity in the sense is that if the complexity level of the application is high the we need to create various methods that will be used to support the functionality , creating reports and debugging purpose which will need a certain amount of time . On the other hand if the application is not complex and small we don't need to create all the supporting factors because it will take time and will not be of much use in case of low complexity test cases.

Functionality of application plays major role in deciding what kind of framework will be suitable for automating the particular application. Application with less and easy functionality doesn't require complex framework, simple framework will be enough to automate application. On the other hand application which is complex, have more complex functionality and have more frequent testing data change requirement need framework that can support all the requirement.

Time Period here specifies the duration of which the automation testing will be done in a specific project. Suppose if in some project we need automate test cases which have to run one time in a month or

have to run only one time in whole life time in this case there is no positive side to develop a complex framework to support the functionality of test cases. This will save the cost and time of the client.

## AUTOMATION VS MANUAL TESTING TIME

While doing the automation testing the manual time taken by the specific test case to get executed should be considered based on which it will be decided whether there is any benefit of automating the test cases or not.

There are two things that can happened on this scenario:

1. Automation testing time for the particular test case will be less as compared to manual testing time this information will decide whether automation is profitable in this case or not.
2. If the automation test time for particular test case is more than when test cases will run manually then automation would not be recommended. As the manual time is less which suggest that we are losing time by automating the test cases as its taking more time as compared to the automation execution time.

Below is the screen shot of manual vs automation data snippet

Testcasename	Total Time Taken Using 1 VDI(Automation) (In mins)	Total Time Taken(Mannual) (In mins)	Total time Taken to Trigger Script(In mins)	Total Time Taken to Validate Script After Dry Run(In mins)	Total Time taken to Validate Failure Script(In min)	Total Time taken to Maintain Failure Script(In min)
TEST CASE 1	15	15	1			
TEST CASE 2	25	35	1			
TEST CASE 3	18	30	1			
TEST CASE 4	15	32	1			
TEST CASE 5	53	40	1			
TEST CASE 6	39	30	1			
TEST CASE 7	9	20	1			
TEST CASE 8	18	25	1			
Total	3.2	3.78	8			
Total Time for Automation execution using 5 VDIs in parallel		0.64				
Time saved using Automation		0.58				
Average time to complete one script		23				
Remaining Automation Regression Sprint		2				

Fig 3. Automation Vs Manual log

## TYPES OF AUTOMATION TESTING TECHNIQUES

### LINEAR AUTOMATION FRAMEWORK

Linear Framework is based on creating the script which is completely based on record and run property of UFT. This framework is highly recommended when the application is very small with very easy functionality. The major challenge while selection the test case is how to pass the values in the script or how to parameterize. This doesn't seems to be an issue with the linear automation framework as it is used when the requirement of data change for the specific test case is almost to zero or very less. In linear automation the parameters to be passed is mostly hardcoded as the change in data is very less.

For every script that have created will have its own unique object repository. While doing the record and run object repository will get created automatically. Below is the screenshot of linear framework.



Fig 4. Linear Automation Framework

## FUNCTIONAL DECOMPOSITION FRAMEWORK

In the Functional Decomposition Automation Framework software testers need to initially identify the reusable code, which requires automation. Furthermore, they also need to write the reusable code in various functions, it is important to note that these functions can be called depending on project's testing requirements.

The reusable functions are identified and called when required in the script. The advantage of the functional decomposition is that less effort have to put on coding part. As most of the functionality had already been coded.

## ADVANTAGE OF FUNCTIONAL DECOMPOSITION FRAMEWORK

- ▶ Test scripts can be created comparatively faster, as the reusable functions are created only once.
- ▶ Minimal effort is required to create test cases due to the availability of reusable functions.
- ▶ If any changes/modifications are made to the application, changes need to be updated only in a single place (function).

## DISADVANTAGES OF FUNCTIONAL DECOMPOSITION FRAMEWORK

- ▶ Scripts cannot be executed with multiple data sets, since data is hard-coded.
- ▶ It takes additional time to identify the reusable scenarios.
- ▶ Programming knowledge is required to write and maintain the functional libraries/reusable functions.

With help of functional decomposition framework the different functionality of the application is decomposed into several parts which includes login part, data processing, scripts attachment, report generation and many other tasks. Then the scripts can be combined in a hierarchical fashion to build larger tests. The functional components are reusable.

Take an example of deleting email(s) from an Inbox, sent by a particular sender.



## DATA-DRIVEN FRAMEWORK

Data Driven framework deals with handling automation framework in which data is consistently changing. This technique adds up feature to the framework using which we can enhance the automation framework, by enabling it to pass the data dynamically while running the test cases instead of hardcoding the data. Although, there are several ways in which we can pass the data to our script running; one way to pass data is through

excel file .Moreover, the code created to pass the data dynamical should be should be error handled. Below are some qualities that should be available:

- 1.Code should be exiguous and should take less time to execute.
- 2.Break whole code into parts which will extract the data into different function according to the need.
- 3.Dictionary object should be used to store the data in from of key and value pair.
- 4.Capable enough to handle run time run time errors.
- 5.Should return reason in case failed to fetch the data from the excel.
- 6.Compelled to handle large data.

➤ Always remember that in QTP we can pass data through ALM. In this case if any data is passed through we have to code for that separately, or in other way , we have to create code to retrieve the data from excel and from ALM in different function.

➤ When the script for retrieving the data from excel run it will retrieve the data from both excel and ALM. Steps that can be used to retrieved the data from the ALM:

- 1.Create OTA connection to get the location of the test set or test case of which we have to retrieve the data.
- 2.Retreiving the TestCase or TestSet Location

```
strTestLocation = objQTP.Test.Location
```

```
For i = 1 to objParms.count
```

```
strTest = objParameter(objParms.item(i).Name
```

```
If err.Number = 0 Then
```

**If condition is used to check if the re is any parameter value provided through alm. If some value have been provided through ALM, the value of err.Number=0 else it will not be zero.**

**Once the flow will come inside the loop we will store the value pf the parameters for the specific test case in form of key and value pair.**

```
End IF
```

```
Next
```

Why do we need to provide value through ALM?

Sometime we need specific values for specific test and that too for a few time, in this case instead of adding one more row in excel file we can directly pass that value from ALM.

Sample code that have been used for extracting data from the excel file.

```
Private Function getexceldata
```

```
Set objCurrentTest = QCUtil.CurrentTest
```

```
strTestName = objCurrentTest.Name
```

```
strTestName = Trim(UCCase(strTestName))
```

```
Set conn = CreateObject("adodb.connection")
```

```
Set re=CreateObject("adodb.recordset")
```

```
intColumnCount = rec.Fields.Count
```

**‘In below code we will retrieve the data from excel ‘sheet that we have created and store that value in ‘dictionary object as key and value pair.**

```
For i = 2 to intColumnCount - 1
```

```
valuetext = rec.Fields.Item(i)
```

```
columnstrname = rec.Fields(i).Name
```

```
Itemkeydict = columnstrname
```

**itemvvalue = valuetext**

**Reporter.ReportEvent micInfo, Itemkeydict & ":" & itemvvalue, "" 'Checks for Env Parameter Item to add to Dic Object**

**AddKeys Itemkeydict, itemvvalue**

**Next**

**END FUNCTION**

7. Same above code can be used to retrieve environment parameter. It is suggested that , its good to created different script which will fetch the environment data and input data, the data that we are providing through sheet.
8. This mainly used in case different script concept will be used only when the application under test is complex, otherwise we can proceed with same code.
9. Same above code can be written to retrieve the actions for particular test cases from excel created in early stage.
10. Once the actions are retrieved from the action sheet , we can create the script to run those action

Below code can serve the purpose of running script:

**'Update the Env variable to avoid making another 'call to Vldaitons action in case it exists  
parame.AddKeys"IterationStatus", "Started"**

**iTotalBP = flowdat.Count - 1**

**'Total Business Processes will be 1 less than the 'total keys as the key also contains Testname**

**' Loop Through all the Business Processes defined 'in TestFlow.xls**

**For i =1 to iTotalBP**

**'Get the Current Business Process string , then 'extract the Corresponding Script Name and  
'Action Name**

**currentcd = flowdat("BP"&i)**

**currentval1 = split(currentcd, ";")**

**currentcdScript = trim(currentval1(0))**

**actioncurrent = trim(currentval1(1))**

**'Call the BP's corresponding Script and Reusable 'Action**

**LoadAndRunActionparame("RA\_QC\_PATH"&currentcdScript,actioncurrent&**

**parame(currentaction&"\_VERSION") , oneIteration**

**Next**

**'Update the Env variable to avoid making another 'call to Validaitons action in case it exists  
parame.AddKeys "IterationStatus", "Complete"**

## **I. Keyword Drive Framework**

Keyword Driven framework is used where the functionality of Application under Test (AUT) is wide. Taking instance of, an application having different functionality and some functionality from all have to be used by specific test cases. In such scenarios, instead of running all the test cases we prefer to run that specific function that is demanded by test case. I will explain in detail in later section how the functionality should be broke down in pieces.

Example,

	A	B	C	D
1	TestCase Name	Flow1	Flow2	Flow3
2	TC_01	Gmail_Login	G_EmailsReceivedToday	Gmail_Logout
3	TC_02	Gmail_Login	G_SendEmail	Gmail_Logout
4	TC_03	Facebook_Login	F_FindNotifications	Facebook_Logout

www.automationrepositrory.com



The Keywords are aligned horizontally



Since the Keyword names are self explanatory, description for the Keywords is not provided

In above example, this will be the second excel sheet that have to be maintained other than that of parameter sheet that have been used to pass the parameters to test cases. We can call this sheet as flow sheet in which the flow of a particular application will be given.

Parallel to this sheet we will develop one code that will take the flow of test cases from this sheet and store it in key and value pairs. Once the value will be stored we can develop below code.

Steps in code:

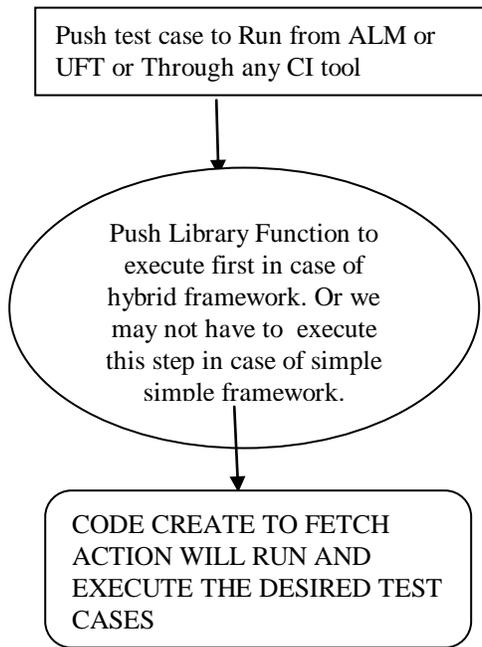
1. Check how many flow have been provided in the excel sheet.
2. Number of flow we can get by counting the number of columns that have been used by a particular test cases.
3. Once we get the count we can use that count to create for loop that will iterate the same number of time as we got value in above step.
4. Inside the for loop we will take the action one by one stored as key value pair and push the action to run.
5. We will iterate through the loop until all the actions are completed.

Below code will help in running the test cases specific action that have been mentioned in flow parameter sheet.

## II. HYBRID FRAMEWORK

Complex applications in which there are large functionalities and test cases to be automated are usually longer than compared than any other test cases, in this cases we mostly maintain different excel sheets for retaining the data to execute the particular test cases and moreover, we use some supporting library scripts in which the foundation of the test cases. This type of framework is known as hybrid automation framework. As usually it consist of data driven and keyword driven framework.

Below is the flow of Hybrid framework:



There are few points that should be taken of before developing hybrid framework such as there is no specific way of creating the hybrid work; it simply depends on the criticality of the project. Hybrid framework based on functional and data driven approach.

### HOW TO START CREATING AUTOMATION FRAMEWORK??

Selection of framework completely depends on the functionality and complexity of the application under test. Certain points are there that should be considered before creating the framework some of which are below mentioned:

1. Check the complexity of functionality of AUT.
2. Check the data requirement. If the project demands for the constant data throughout then we can hardcode the data. However, if the project requirement is changing data then it's good to use parametrize every field and use data-driven framework.
3. Again if the application is small we will not prefer using data-driven framework even if the complexity of AUT is more. The reason for is this, that coding for data driven framework will take more time and will be less value here as , because of small size and less complexity of AUT.
4. Broke whole functionality into scripts. Means, functionality that will be used most of the time will be considered as one script. In this way create as many as script possible and convert this script into reusable script. Making script reusable will help us in reusing the script for other test cases which we eliminate redundancy and will follow code ethics.
5. Once this is done we have to create plan for how to load data from the excel file, so that we can use that in our test cases.
6. The best way to do that is writing all code inside a class in such a way that all the functions declared or created will be inside the function. This will help us in calling these function from different scripts.
7. These should be separate script for every test case which will use these functions.
8. We can call these functions as functional library as they will be developed once and modified only when critical functionality will be updated.(will explain in detail later)
9. The other things that we have to do is creating a recovery scenarios if needed. Recovery scenarios will help in executing the test cases even if some error occurs.
10. OTA (Object Test Architecture) framework can be developed separately that will help in handling the

object run time property change. There are several ways in which we can change the property of the object dynamically during run time which will update the property to specific object to the current during execution time and will cease failing of test cases.

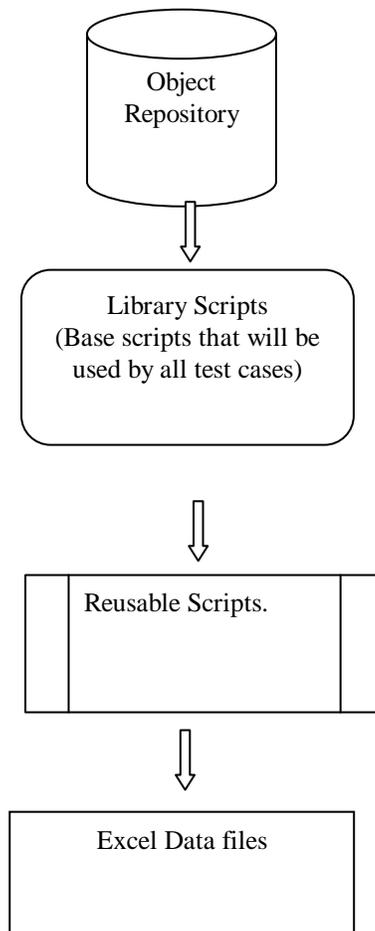
11. HTML reporting can be done which can be used to retrace the reason for failure of test cases.
12. Starting point of all test cases should be same.
13. It helps to create test cases easily and automating test cases in most feasible way.

There are certain other things that should be noted before developing the framework.

On such thing is creation of shared object repository.

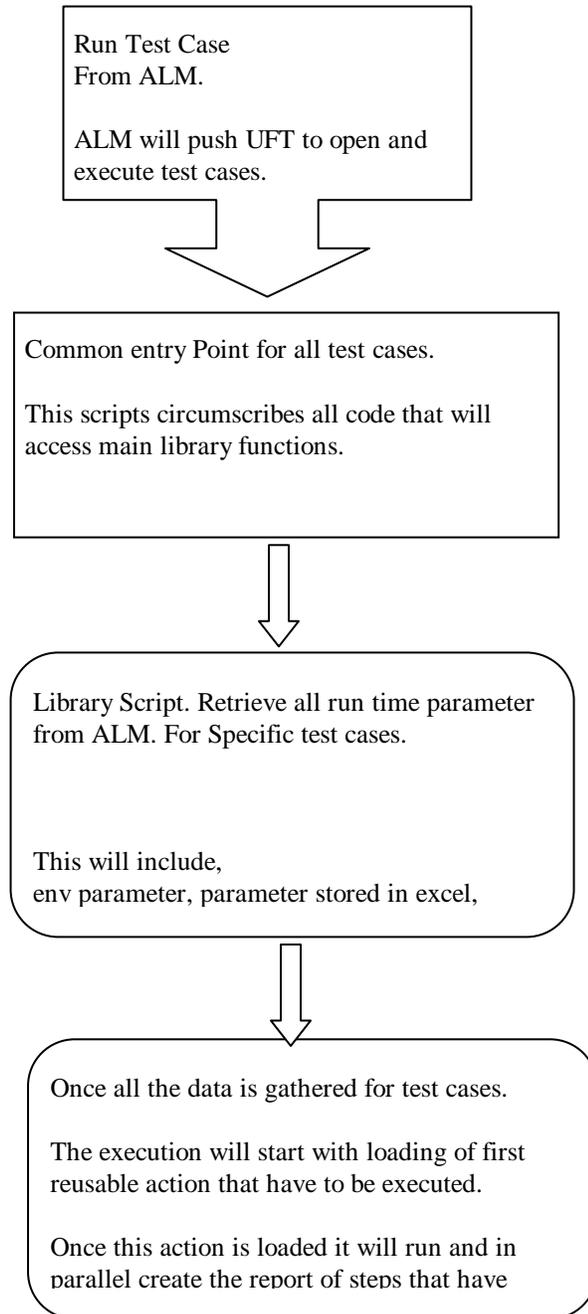
Shared object repository provides the way to maintain the object's in single repository this repository then can be modified and reused by other scripts in a feasible way.

This in addition will make the execution fast, as we don't have to load different object repositories for test cases.



How the execution will happen.

Before running test cases we should update our parameter and other sheets so we can retrieve data for particular test case from there.



## REFERENCES

1. <https://www.tutorialspoint.com/qtp/>
2. QuickTest Professional Unplugged Book by Tarun Lalwani.
3. Designing and Implementing Test Automation Frameworks with QTPBook by Ashish Bhargava.
4. Advanced UFT 12 for Test Engineers Cookbook Book by Jonathon Lee Wright and Meir Bar-Tal.
5. G. Franceschini and V. Geffert, "An In-Place Sorting with  $O(n \log n)$  Comparisons and  $O(n)$  Moves", Proceedings of 44th Annual IEEE Symposium on Foundations of Computer Science, (2003), pp. 242-250.
6. O. O. Moses, "Improving the performance of bubble sort using a modified diminishing increment sorting", Scientific Research and Essay, vol. 4, no. 8, (2009), pp. 740-744.

7. Pandey A., Bansal K.K. (2014): "Performance Evaluation of TORA Protocol Using Random Waypoint Mobility Model" *International Journal of Education and Science Research Review* Vol.1(2)
8. Tiwari S.P., Kumar S., Bansal K.K. (2014): "A Survey of Metaheuristic Algorithms for Travelling Salesman Problem" *International Journal Of Engineering Research & Management Technology* Vol.1(5)
9. Yadav R.K., Bansal K.K. (2012) : "Analysis of Sliding Window Protocol for Connected Node" *International Journal Of Soft Computing and Engineering(IJSCE)* Vol.2(5) pp.292-294
10. D. Knuth, "The Art of Computer programming Sorting and Searching", 2nd edition, Addison-Wesley, vol. 3, (1998).
11. Thorup M., "Randomized Sorting in  $O(n \log \log n)$  Time and Linear Space Using Addition, Shift, and Bit Wise Boolean Operations," *Computer Journal of Algorithms*, vol. 42, no. 2, pp. 205-230, 2002.
12. Aho A., Hopcroft J., and Ullman J., *The Design and Analysis of Computer Algorithms*, Addison Wesley, 1974.
13. Astrachanm O., *Bubble Sort: An Archaeological Algorithmic Analysis*, Duk University, 2003.
14. Deitel H. and Deitel P., *C++ How to Program*, Prentice Hall, 2001.
15. [http://en.wikipedia.org/wiki/Software\\_testing](http://en.wikipedia.org/wiki/Software_testing)
16. [http://www.tutorialspoint.com/qtp/qtp\\_test\\_automation\\_process.htm](http://www.tutorialspoint.com/qtp/qtp_test_automation_process.htm)
17. <http://www.onestoptesting.com/win-runner/>
18. <http://www.careerride.com/Testing-QTP-vs-WINRUNNER.aspx>
19. <http://sqa.fyicenter.com/FAQ/QTP-QuickTest>
20. Research article "COMPARISION OF SOFTWARE TESTING TOOLS QTP AND SELENIUM USRD" *International journal of Computer Science and Information Technology*, Vol. 3No. 6 June 2013.
21. Innovative approaches of automated tools in software testing and current technology as compared to manual *Global journal of enterprise of information system*, jan 2009.
22. [Http://en.wikipedia.org/wiki/manual-testing](http://en.wikipedia.org/wiki/manual-testing).
23. Automation testing--[www.guru99.com/automation-testing.html](http://www.guru99.com/automation-testing.html).
24. Sahil Batra and Dr. Rahul Rishi "IMPROVING QUALITY USING TESTING STRATEGIES" *journal of Global Research in Computer Science*, Vol.2, No.6, June 2011.
25. Software Test Automation-[http://en.wikipedia.org/wiki/Test\\_automation](http://en.wikipedia.org/wiki/Test_automation).