**6 Popular Test Automation Frameworks for UFT (QTP)**

 [Home](https://www.evoketechnologies.com/blog) / 6 Popular Test Automation Frameworks for UFT (QTP)

30 April 2015[Naveen Varadaraju](https://www.evoketechnologies.com/blog/author/nvaradaraju/)[4 Comments](https://www.evoketechnologies.com/blog/popular-test-automation-frameworks-uft/#comments)

**Post navigation**

In the software testing domain, automation frameworks are considered to be of critical importance, especially when you are involved in automation testing. In this blog, we would be exploring the different types of [automation frameworks](http://www.evoketechnologies.com/blog/test-automation-framework-design/). So, what exactly is an automation framework. In technical terms, an automation framework is a set of guidelines, which comprises of coding standards, structure of app code and app data folders, test results storage, accessing external resources and shared object repositories etc. Now, let’s take a look at some important features/requirements of a test automation framework.

**Standard Automation Framework Requirements**

* Code must be re-usable, comprehensible and maintainable.
* Data Parameterization must be a part of the automation framework.
* Test data should be kept distinct from the code and the code should be reusable for different sets of input data.
* Test results should be logged into a separate log file and should be fairly easy to understand.
* Appropriate error handling process must be included as part of the automation framework. Additionally, usage of recovery scenarios must be kept at minimal, as recovery scenarios consume additional processing time and other resources.
* The automation framework must be written in such a way that if one test script fails, [Unified Functional Testing](http://www8.hp.com/in/en/software-solutions/unified-functional-automated-testing/) (formerly QTP) should not stop the execution process. Ideally, the framework should report the errors and move on to execute the next test case.



**Types of UFT (QTP) Frameworks**

Basically, there are six standard automation frameworks that are commonly followed in the software testing domain, especially while performing automation testing. In the long run, it would be highly beneficial for software testers to have a broad understanding of these automation frameworks. So, let’s explore these frameworks.



**Linear Automation Framework**

In the Linear Automation Framework, software testers do not create functions and moreover all steps are written in a sequential order.

Here’s an example to demonstrate Linear Framework, let’s assume we are automating UFT’s (QTP) Flight Application; in this instance, we would be recording the steps as illustrated below.

*Login -> Book a Flight Ticket -> Insert Order -> Exit*



**Advantages of Linear Framework**

* Automation expertise is not necessary, as we are just using ‘*Record*‘ and ‘*Playback*‘ feature.
* Test scripts can be easily written and it will consume minimal time to write these scripts.
* Understanding the flow is comparatively easy as the scripts are written in a linear manner.
* Extensive planning is not required as built-in functionality is utilized.

**Disadvantages of Linear Framework**

* Lacks reusability.
* Maintenance of Linear Framework is considered to be challenging.
* All input data is hard-coded, hence test cases cannot be run with multiple data sets.
* If the application is modified at any given point, a lot of rework is required, as the same type of code is replicated at several instances.

**Functional Decomposition / Modular 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.

Let’s consider an example of the Functional Decomposition Framework.

*Login -> Book a Ticket -> Insert Order -> Verify the Flight Number and Departure Time -> Exit*
*Login -> Book a Ticket -> Insert Order -> Verify the Price and Total Amount to be paid -> Exit*



**Advantages 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.

**Business Process Testing Framework**

In the Business Process Testing (BPT) Automation Framework the test cases can be divided into multiple scenarios and each scenario can be scripted by utilizing a Business Process Component (Available as part of HP’s Quality Center). When all components of a particular test case are ready, these can be linked to the test cases and can be executed directly from [HP’s Quality Center Tool](http://www8.hp.com/in/en/software-solutions/quality-center-quality-management/) (Quality Center Enterprise).



**Advantages of Business Process Testing Framework**

* Extremely easy for Business users to create test scripts utilizing Business process components.
* The Business Process Testing Automation Framework offers a lot of built-in features, such as Component Linking and Data Parameterization. This saves a lot of time for business users, which would have otherwise been spent on preparing reusable functions.

**Disadvantages of Business Process Testing Framework**

* HP’s Quality Center (Quality Center Enterprise) Tool is required to perform Business Process Testing; Licensing cost should also be taken into consideration, as it can impact the project budget.

**Data-Driven Framework**

In the Data Driven Automation Framework, the Input Data/Input Parameters are not hard-coded in the test scripts. Instead, these are stored and passed from external files/resources such as Microsoft Excel Spreadsheets, Microsoft Access Tables, SQL Tables and XML files etc. The test scripts connect to the external data source and read the data.



**Advantages of Data-Driven Framework**

* A single test can be executed with multiple data sets.
* Hard-coding can be avoided.
* Faster execution helps in saving time.

**Disadvantages of Data-Driven Framework**

* Strong technical expertise is required to identify external data sources, and create functions that connect to these sources seamlessly.

**Keyword Driven Framework**

In the Keyword Driven Automation Framework, we can create multiple keywords that allow testers to associate a unique action or function for each of these keywords. It helps software testers to create an Action or Function Library that holds the logic to read keywords and call the associated action.

Correspondingly, a shared object repository is used to map objects to all required actions. Generally speaking, in the Keyword Driven Framework, test scenarios are written in a Microsoft Excel spreadsheet. The Driver Action will read the scenario and perform test execution.



**Advantages of Keyword Driven Framework**

* After setting up the framework, writing test scripts is relatively easy.
* Programming skills are not essential, since a Microsoft Excel spreadsheet is generally used to write various test scenarios.

**Disadvantages of Keyword Driven Framework**

* A significant amount of time is consumed to set up this framework, including setting up Functional Libraries and Shared Object Repositories.
* It also takes a fair amount of time to understand the Keyword Driven Automation Framework. Furthermore, it is a bit complex to make any changes to this framework.

**Hybrid Framework**

The Hybrid Automation Framework is created by combining distinct features of two or more frameworks mentioned above.

**Advantages of Hybrid Framework**

* The Hybrid Framework is a blend of the best features of various Automation Frameworks.
* It is highly robust and flexible, if implemented effectively.

**Disadvantages of Hybrid Framework**

* Strong technical expertise is required to design and maintain the Hybrid framework.

**Conclusion**

To summarize, Automation Frameworks play a key role in software testing. Software testing teams must carefully review all available frameworks before making a decision on any framework. The above guidance would certainly help the teams to identify and apply the right automation framework for their projects.