Selenium VS QTP

* **The blow is a detail comparison with QTP and Selenium testing tool, QTP is a popular business process testing tool, selenium is another popular open-source tool.**

|  |  |  |
| --- | --- | --- |
| Quick Test Professional | Selenium | Who is the best Choice? |
| QTP is expensive | Selenium is open source and free | Selenium |
| QTP works for both Web and Windows application testing | Selenium natively works only for Web based applications, but you can integrate with Auto It or Robot to do some the other type testing. | QTP |
| QTP add ins supports many additional application type,WebServices,WPF etc. | No need to add ins, for the difficult application like Ajax, Flex, or Flash testing, need to integrate with some other library. And open source community are developing furthermore add ins for Selenium | Both |
| Support VBScript as programming language for developing test scripts | Support many languages, like Java, Python, Ruby, C# etc. As selenium is developed by Java ,so using java language is the best choice | Selenium |
| Not much coding knowledge required for creating simple scenario | Programming knowledge is must | QTP for non-programmer |
| Distributed testing in QTP is impossible | Distributed testing can be done via selenium Grid or via Jenkins | Selenium |
| Test cases can be written and maintained via ALM | Test cases can be written and maintained via CI tools ,like Jenkins or Cruise Control | Both |
| QTP supports following browsers(IE ,Firefox, Chrome) and the browser’s version is sensitive | Selenium support almost all the popular browsers, like IE, Firefox, Chrome, Opera, Safari, Android etc.  Every browser’s version is well for it | Selenium |
| Record and playback in QTP can be done via QTP IDE itself | Selenium record and playback can be done via another tool, Selenium IDE(it’s Firefox’s add in)  And the record code is the same as written code. | QTP |
| Object Repository is the heart of QTP and store almost all the recognized object ,shared Object Repository can also make multiple people to work on same object same time ,but it’s better we not use the object Repository ,it’s hard to maintain .we use the descriptive program . | It uses the Page Object Pattern to organize every page easily, just need the xpath. It’s very easy to maintain the testing page. | Selenium |
| Development of testing script can be done via QTP IDE itself | Develop code can be done via Eclipse, Net bean or any other popular java IDE; | Depends on you |
| Study says QTP knowledge among automation testing community is getting less | Study says Selenium is getting huge and many Employers are suggesting Selenium as their automation testing tool.  has tons of free information online | Selenium |
| No 3rd Party automation test frameworks available. That’s why there are many different complex framework for QTP | There are lots of free agile automation framework available ,and hence development of code can be done much faster with best practices | Selenium |
| With the ALM integrated, so you can review the QTP test result from there clearly. | Test result via JUnit or TestNG. No built-in Report with Selenium.  Selenium itself cannot generate the test report ,but you can integrate it with another open-source tool to manage the test result (Test Link), | Both |
| Using Recovery Scenarios to catch the error, more recovery scenarios and slower test execution. | Add a listener into the Junit or testNG. | Both |
| Debugging code with built-it editor, but reface the test methods across nullity files is too bad. | With the Eclipse, I think it’s much easy to maintain. | Selenium |
| Object Spy with QTP itself and recognize is not good ,it’s difficult to get the object locators | Firebug and Firebug Path, it can recognize using CSS selector or Xpath selector across the nullity browsers, and it can get the object locator very easy. | Selenium |
| It required less effort to create a script ,as it has a very user friendly script environment | User needs to have good amount of java skill and more coding effort is required to implement the function | Depends on you |
| QTP only support windows very well. | Selenium supports more number of OS like windows ,Linux or Mac | Selenium |
|  |  |  |
|  |  |  |

* **The choice for these two to take**

**Selenium is an excellent choice if...**

* you have developers responsible for writing and maintaining the tests
* you need to have these tests as part of a larger infrastructure
* you are pretty sure you do not need to do much testing outside of the browser
* you think you might want to test other browsers and other platforms besides Windows
* you want something free

**QTP might be a good choice if...**

* you have less code-savvy people writing and maintaining tests (though I am not sure if Selenium IDE is hard to learn for non-coders)
* if you need significant testing outside of the browser

I have implemented both tools in the past, the decision is always based on many factors including: project, application, expertise, etc.   
  
if you asked me I would say automation in general has a great future, regardless of the tool. Tools can be learned within a week, I would focus on the proper design of automation frameworks first

* **the knowledge need for Selenium testing**

**Java +TestNG+Ant+Jenkins**

1, Java language background, with java base experience;

2, TestNG or Junit Framework, very simple review at these links:

Learn guideline: <http://testng.org/doc/documentation-main.html>

Intergraded with Selenium: <http://testng.org/doc/selenium.html>

3,Ant Task ,actually no need to learn deeply ,just know a little knowledge for this .as all the Ant script had been done before we build the test framework;

4, Jenkins, this’s used for executing the Selenium, so you can know it very quickly in about 8 hours;

|  |  |
| --- | --- |
| **Background we need for Selenium** | **Time needs** |
| Java | Basic java program ,and it’s enough if there are one people know it well in one team |
| TestNG | 3 days almost |
| Ant | It can be learned very quickly ,about 1 day |
| Jenkins | 1 day is enough for a tester |
| Xpath | Basic knowledge :1 day ,practice :2 or 3 days |

PS:

Separate every test, so if the test is failed we can go ahead to run the other tests quickly.

The framework needs to more ability to identify the error where it’s.

If the error met, the test better to stop and we manually identify the error why??? And then we can begin to run the other tests quickly.

So that’s mean every test we need to use one job to do it in the Hudson project;

Or another good choice is that we run the tests from eclipse in the execution host;

You can refer the blow link for a reference: <http://executeautomation.com/blog/quick-test-professional-vs-selenium-2/>