

**Quotium's Windows Robot for QTest** enables performance testing of almost any application without the need for complicated scripting and '.dll' calls from the host application. Windows Robot works differently from protocol script based tools meaning that scripting with complicated 'middlewares' or protocols can be avoided.

Some applications lend themselves very well to working at protocol level such as web for example. However some really do not such as COM and proprietary protocol based applications. For these applications QTest Windows Robot becomes a very useful solution indeed.

The choice is simple – spend days or weeks reverse engineering the application under test by re writing it as complicated scripts in the testing tool or simply record a GUI (clicks, drags & keystrokes) based transaction using Windows Robot for your performance test.

The benefits are clear – WinLoad can take days or weeks off the performance testing phase or even better, can enable performance testing where previously it was either too complicated or time consuming to do.

Windows Robot works on the GUI level meaning that when WinLoad transactions are replayed they are replayed through the user interface of the application. This means that response times measured using Windows Robot are more accurate as they include the time the client application part of the system under test takes to render the raw data received at the protocol level.

How do we simulate hundreds of users all using the same client front end? By using a one or more server specification machines as injectors, many instances of the client application can be run by Windows Robot at one time. The console controls the terminal server sessions and enables the tester to take a real time view on transactions as they run. To see a video demonstration of this please see [here](#). The console makes connections either to a Terminal Server or Citrix Server using the RDP protocol.

QTest then gathers timings and server health stats from the systems under test to give an accurate report on the results of the load test. We also carefully monitor the injector machines to ensure that they remain in a healthy state and do not cause delay to the transactions we are testing for.

The diagram below explains the way the load is generated:

**Quotium**

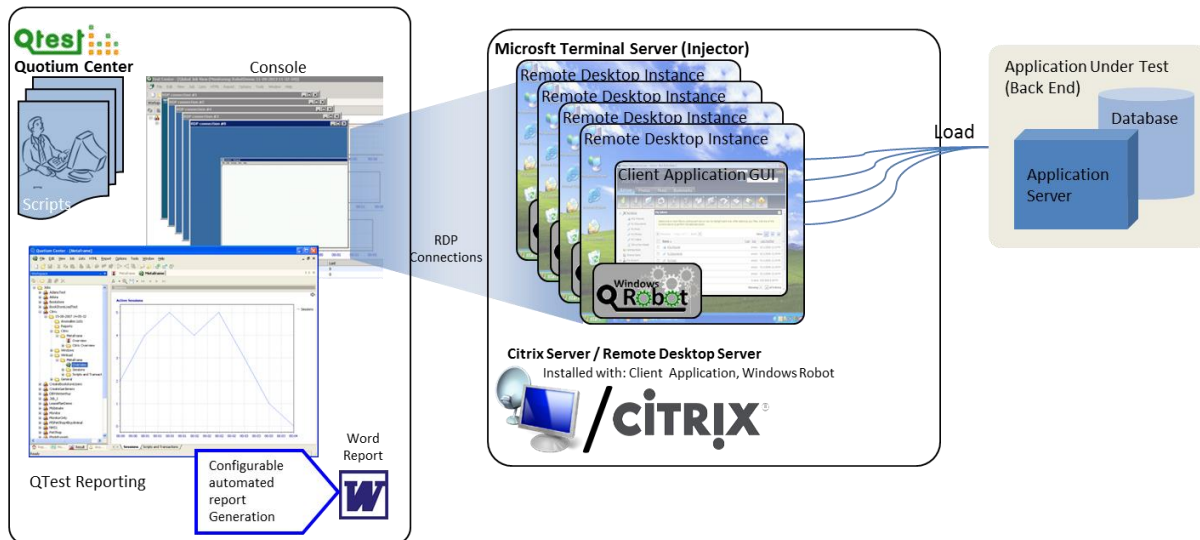
QUOTIUM TECHNOLOGIES Limited

A private limited company registered in England. Registration number: 884871468, Company Number: 5813928

Business Office: 4-5 Park Place, London SW1A 1LP

Registered office: 300 High Holborn, London WC1V 7JH

Phone: +44(0) 203 178 3683 FAX: +44(0) 207 898 9101 [www.quotium.com](http://www.quotium.com)

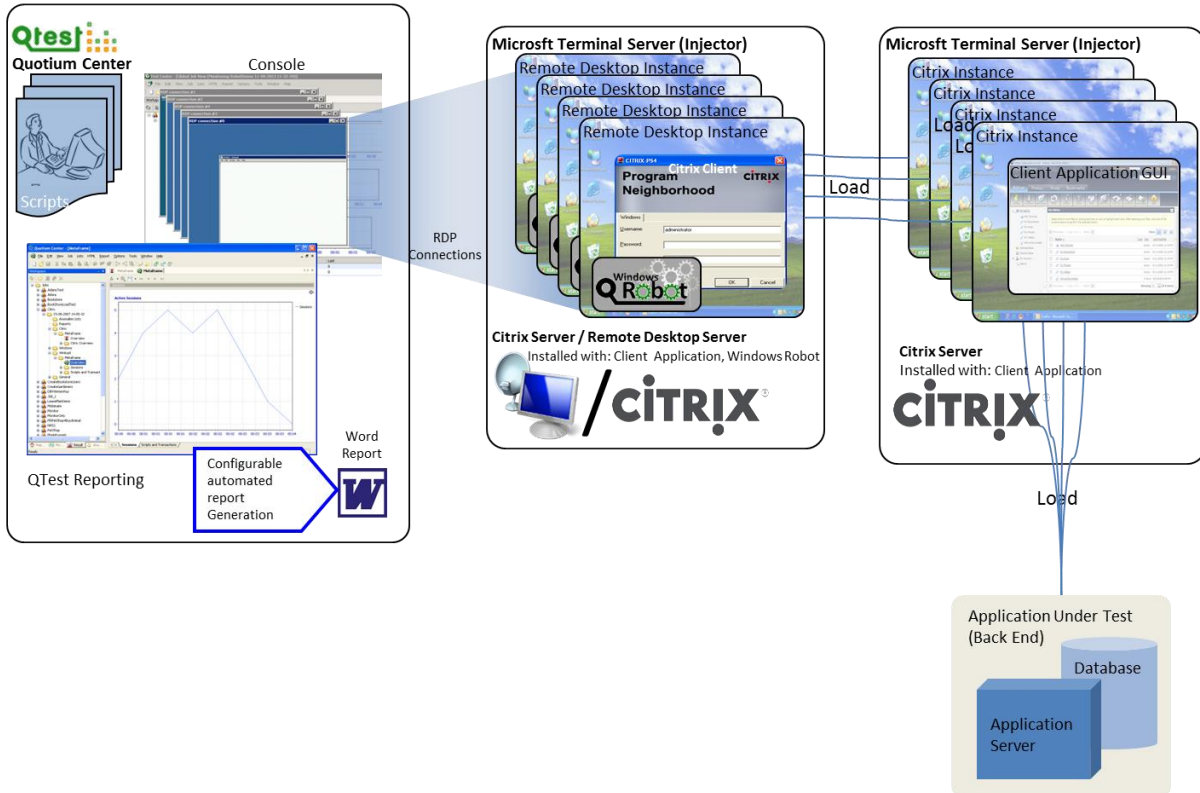


In the diagram you can see at the left Windows Robot generating load by making many Remote Desktop connections to a Remote Desktop or Citrix server where the target application is installed (middle box). On the server there is also a small agent (Windows Robot) installed. This agent directly manipulates the target application based on the instructions that it receives from Quotium Center which is in turn instructed by the scripts that are running. As a result load is generated against the back end of the system which may be either a data base or a multi-tier system.

In an alternate configuration shown as below, the model uses a Terminal Server purely as an injector machine. That machine then manipulates a Citrix (or any type of) client directly through the GUI. There are two reasons that this configuration might be chosen:

1. The Citrix server (or whatever type of technology you are using) is not subjected to the small amount of load that the Windows Robot creates through its execution.
2. It is important to know about the performance of the Citrix (or whatever technologies) protocol over the network.

This approach does add a layer of complexity and is only necessary if points 1 and 2 above are absolutely necessary. Often the only requirements are to understand how well the target application itself scales, this is because the scaling of Citrix and other server types is well understood. However the target application itself is often bespoke and little is known about its performance levels so there are questions about how well it will scale with users, transaction rates and populated databases.

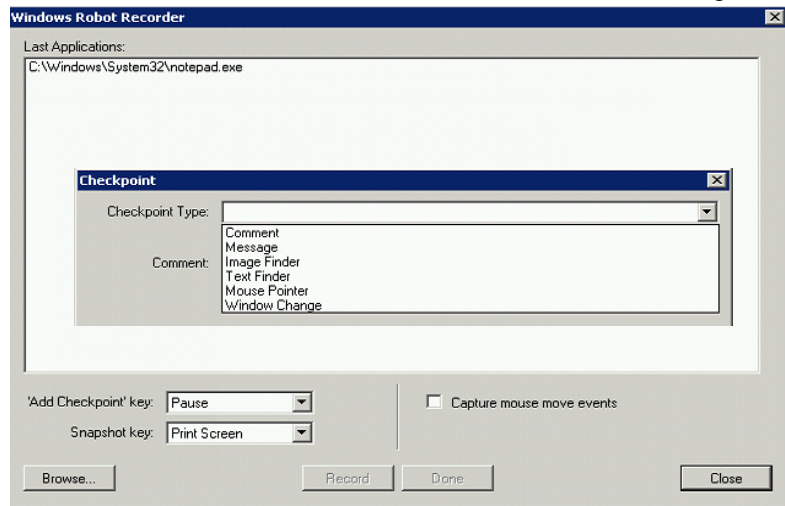


## User Journeys

User journeys or scripts are generated in the same interface as standard QTest Web scripts. WinLoad transactions are recorded by hitting the record button. QTest then makes a connection to the target server where Windows Robot captures mouse movements, mouse clicks, key strokes and extra control information that is easily entered using the pre-set keys.

When replayed the transaction is synchronised using bitmap, windows changes or text captures made during the record process, this means that the transaction will wait until it sees something on the screen that it saw during capture making for robust playback.

This is essential because when replayed, especially while under load, the application cannot be guaranteed to give the same response times. Therefore, without these synchronisation points automation like this may be trying to click on a button that has not appeared yet. The Robot also synchronises automatically on Window Events meaning that we don't always have to put in these bitmap synchronisation points. The bitmap synchronisation points are especially useful for Applications under test that do not have Window Events such as the Citrix client which is simply displayed as a large bitmap.



## Quotium

QUOTIUM TECHNOLOGIES Limited

A private limited company registered in England. Registration number: 884871468, Company Number: 5813928

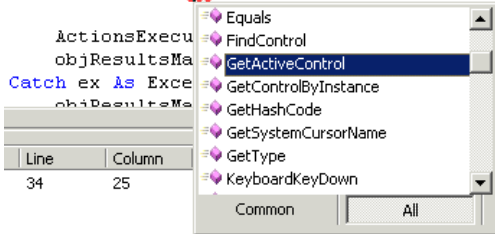
Business Office: 4-5 Park Place, London SW1A 1LP

Registered office: 300 High Holborn, London WC1V 7JH

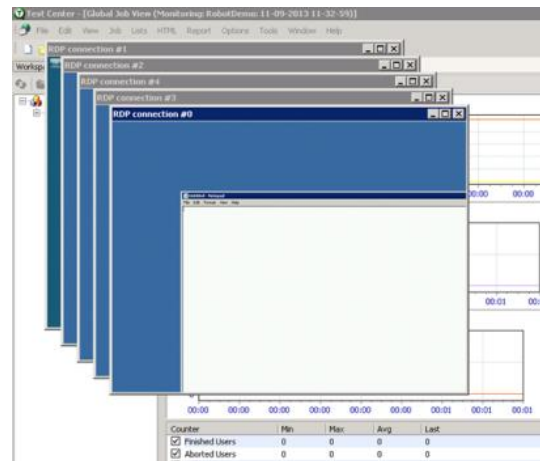
Phone: +44(0) 203 178 3683 FAX: +44(0) 207 898 9101 www.quotium.com

Try

```
Dim TargetWindow As WRWindow = objExecu
objExecuter.SetForegroundWindow(TargetW
objExecuter.TargetWindow = TargetWindow
objImageFindProcessor.TargetWindow = Ta
'Set the default KeyboardSendKey behavi
'VirtualKeyCode : Virtual Key code, Sca
objExecuter.
```



If intelligence is required in the VBA script (see left) then it can be edited using QTest Script Editor. This could be for certain attributes such as customer numbers or other field names that might need to be pulled from a data file. Once the script is ready for use in a test, the console (see below) enables full visibility of each virtual user in the test. It also allows virtual users to automatically log on with different credentials. During the replay of the test, transactions will be seen running from the console which is a view into the injector machine or target machine. This view can be suppressed to improve performance of the test rig. To see a video of the script recording and replay in Script Editor please see [here](#).



After the test Quotium Center consolidates results from back end monitoring and response times into one result set just as would be done with a Web Test in QTest. These results can be viewed either within the tool or can be used to generate a report. Please see documentation on report generation for more details about this.

# Quotium

QUOTIUM TECHNOLOGIES Limited

A private limited company registered in England. Registration number: 884871468, Company Number: 5813928

Business Office: 4-5 Park Place, London SW1A 1LP

Registered office: 300 High Holborn, London WC1V 7JH

Phone: +44(0) 203 178 3683 FAX: +44(0) 207 898 9101 [www.quotium.com](http://www.quotium.com)