



### Traveling Performance Diagnostics Laptop using LoadRunner GUI Vusers

#### The Challenge

Our customer, an international provider of 24/7 on-line business-to-business data and commercial interchange, wanted to equip a laptop with a software toolkit that they could take to customer sites to troubleshoot and diagnose performance problems. The laptop needed to have a way to reproduce application activity from the customer site while monitoring latency through the customer's network and activity on their own application infrastructure.

#### The Solution

We installed on a laptop these components: LoadRunner, QuickTest Pro, a GUI v-user license, and a QTP script prepared with a previously reported troublesome application workflow. We inserted transaction markers into the QTP script to time the response of key actions which would be reported to the Controller. After the test, we correlated these QTP timings with server resource metrics.

We set up a single-user scenario using the QTP script, configured the LoadRunner monitors to monitor Unix resources on their apps and database servers, and the Network Delay monitor to measure latency of network hops. As a next step, we plan to add the Oracle client libraries and configure a dsn so that we can monitor Oracle DBMS internal activity as well, since a key suspect causing slow performance were un-optimized SQL queries.

#### Lessons Learned

1. Using the combination of LoadRunner, QTP and a GUI vuser – which is inexpensive – is an effective way of pulling together application timings, server metrics, bandwidth throughput, and network latency into a single Analysis session to visually correlate all these metrics. It eliminates the otherwise manual steps of exporting to Excel, creating pivot tables, and graphing – skills that not all testers may have.
2. This is an effective solution to equip a field tester/diagnostician to evaluate customer reported performance problems that may have either site-specific network latency issues or load related infrastructure bottlenecks. It's an objective approach that presents a professional, concrete, 'going the extra mile' response to an irate customer.
3. If the ROI does not justify purchasing a Controller license specifically for a laptop, another option is to drive the scenario from a centralized Controller. The stationary controller can either be remotely accessed over Remote Desktop from the customer site, or launching the scenario can be coordinated by phone with another tester at the Controller site.

### Technical Issues and Solutions

**Issue:** Getting QTP script to report timings of key actions.

**Practical Solution:**

Insert Start and End transactions in this format:

```
Services.starttransaction "<name>"
```



<action steps>  
Services.endtransaction "<name>"

Example:

```

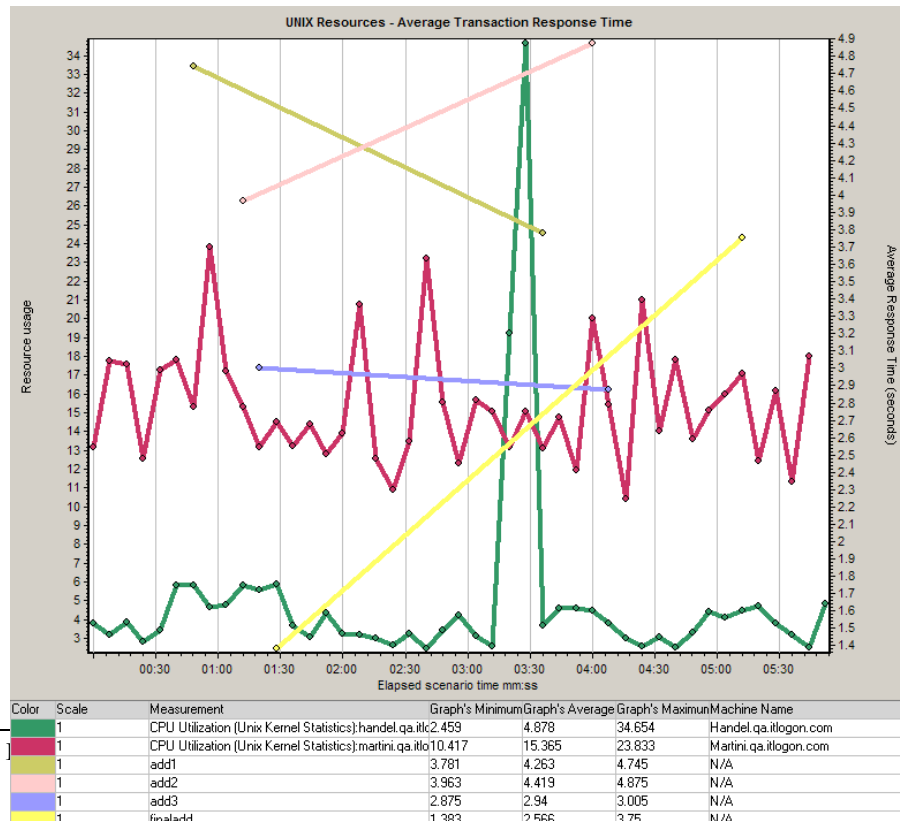
Action1
7: Call IW_Login_Admin()
8: Call Set_TextField("searchField",0, "412909873")
9: Call Click_Image_byImageFileName("search_header.gif",0)
10: Call Click_Image_byImageFileName("admin_inactive.gif", 0)
11: Call Click_Link("Event Notification", 0)
12: Call Click_Link("Configure Event Notifications", 0)
13: Call Click_Button("Add Document/Archive Notification", 0)
14:
15: 'Clicking the add notification button right away to verify error messages are generated.'
16: Services.starttransaction "add1"
17: Call Click_Button("Add Notification", 0)
18: Call Verify_WebElement_ExistsByOuterhtml("<FONT class=InputFieldLabel>Name is required.</FONT>", 0)
19: Call Verify_WebElement_ExistsByOuterhtml("<FONT class=InputFieldLabel>At least one notification event must be selected.</FONT>", 0)
20: Call Verify_WebElement_ExistsByOuterhtml("<FONT class=InputFieldLabel>Email Address 1 is required.</FONT>", 0)
21: Services.endtransaction "add1"
22:
23: 'Adding a name to get rid of one error, then verifying the other two still exists'
24: Call Set_TextField("name", 0, "AutoEmailTest")
25: Call Click_Button("Add Notification", 0)
26: Call Verify_WebElement_ExistsByOuterhtml("<FONT class=InputFieldLabel>At least one notification event must be selected.</FONT>", 0)
27: Call Verify_WebElement_ExistsByOuterhtml("<FONT class=InputFieldLabel>Email Address 1 is required.</FONT>", 0)
  
```

**Issue:** Correlating response times and metrics in LoadRunner Analysis.

**Practical Solution:**

The Controller automatically exports monitored server resources and script transaction times into a local Access file which Analysis drives from to generate graphs.

Sample graph:





**Issue:** Monitoring application infrastructure through firewalls from the customer site.

**Practical Solution:**

We discussed implementing Monitor-over-firewall (MOFW) on a dedicated monitoring server collocated with the application infrastructure, with the MI Listener installed on the laptop. We did not implement this, but MOFW is designed to monitor resources that are separated from the LoadRunner Controller by one or more firewalls, as long as SSL traffic over port 443 is enabled in the firewall. As an alternative resource usage monitored locally via sar/perfmon scripts could be imported into LoadRunner after the test. [Note: Exercising either of these options was outside the scope of this POC, but we've successfully implemented them in other projects].

**Issue:** Staying cool with the customer "under fire".

**Practical Solution:**

Going on-site with a customer to diagnose a performance problem often means that the situation has been escalated beyond 'Support' and customer frustration is running high. Scheduling an on-site session requires both business and technical support, thus disrupting people's jobs – which raises temperatures even further. Recognizing that not all QA testers have this "under fire" experience, we identified a gap: teaching basic consulting skills to the diagnostic team. These skills would include use of the diagnostic laptop tools, basic diagnostic skills, and consulting skills that embrace the concept "never let the customer see you sweat".

This could be the subject of a whole separate *Practical Solution!*

-----

**About Mentora** [www.mentora.com](http://www.mentora.com)

Mentora is a pioneer in the new generation of managed application hosting providers, combining support of Linux and the newest technologies, system performance testing and a complete menu of managed services for networks, servers, operating systems, databases, and storage, with our exclusive, personalized **named-engineer-on-call** and **Application Availability SLA**. Headquartered in Atlanta, Mentora has offices in Boston and Washington DC, and delivers services nationwide.