

# Is Your Database Stressed? Load Testing for Peak Performance

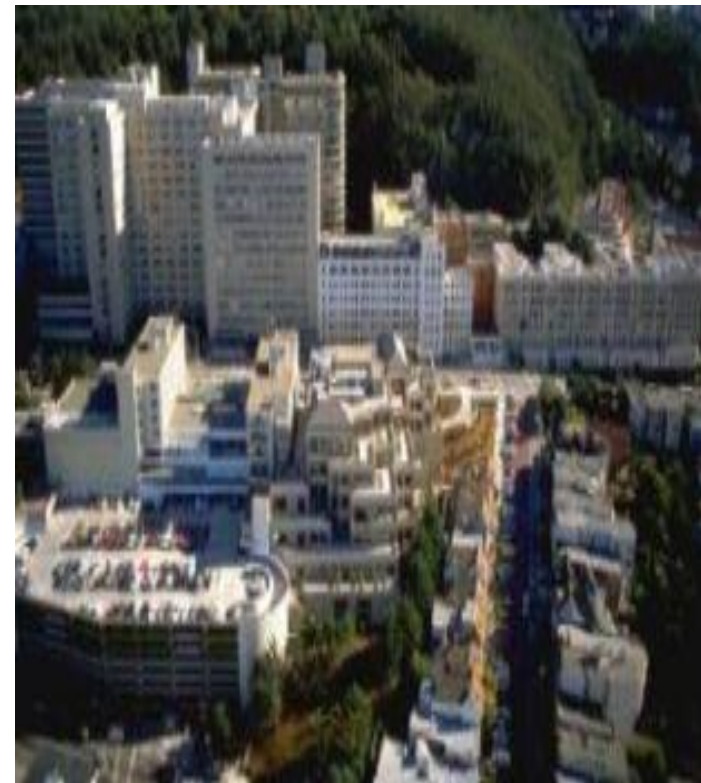
Noelle A. Stimely  
Senior Performance Test Engineer  
University of California, San Francisco  
[noelle.stimely@ucsf.edu](mailto:noelle.stimely@ucsf.edu)

# Who am I?

- ▶ Senior Oracle Database Administrator for over 13 years
- ▶ Senior Performance Test Engineer
- ▶ Past Oracle Instructor
- ▶ Past Lead Database Administrator
- ▶ Oracle Certified Professional (OCP) – 8, 8i, 9i, 10g
- ▶ Hewlett Packard AIS – LoadRunner v11 Certified

# University of California, San Francisco

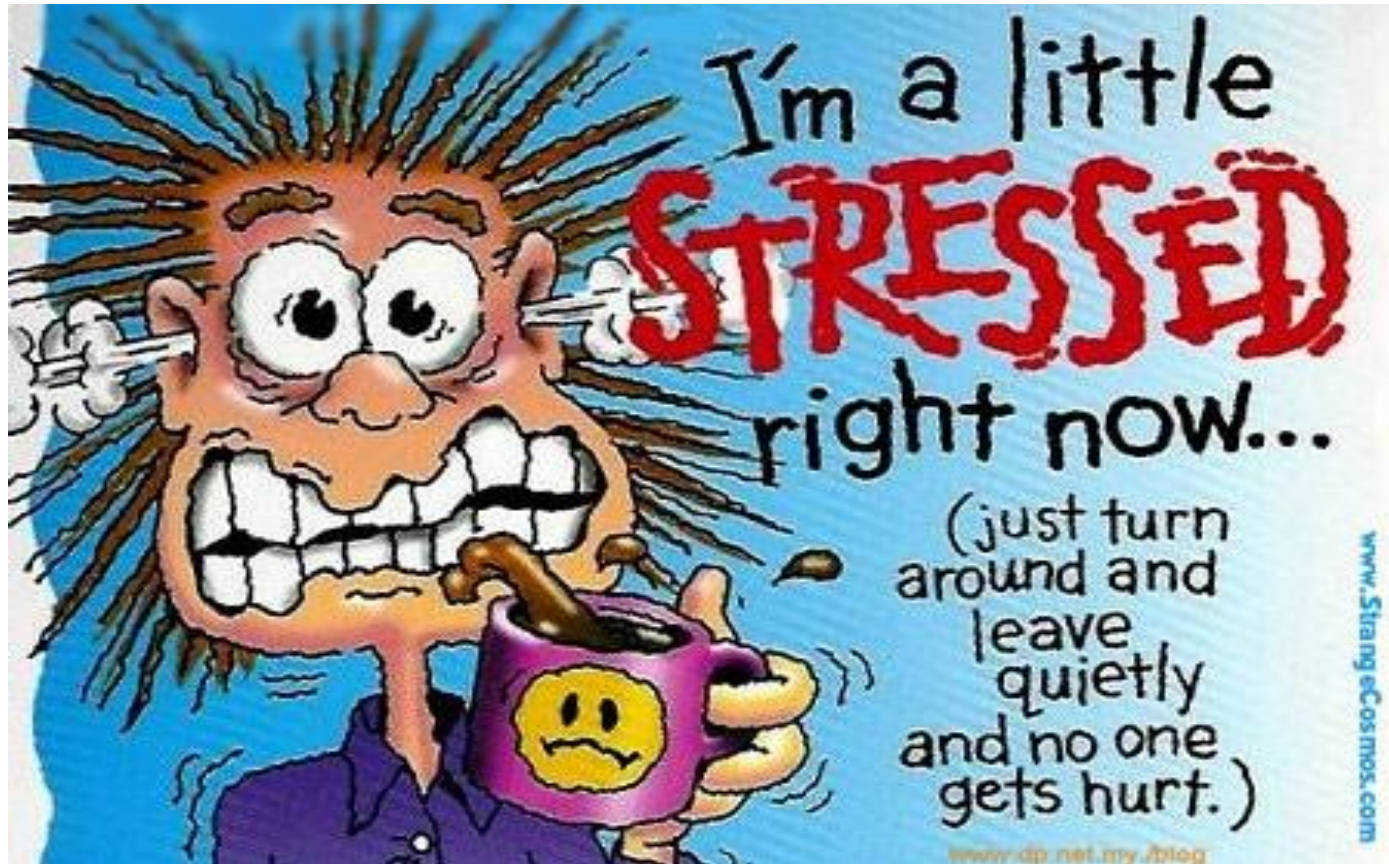
- ▶ University of California, San Francisco is a leading university dedicated to defining health worldwide through advanced biomedical research, graduate level education in the life sciences and health professions, and excellence in patient care
  - 3000+ Students
  - 20,000+ faculty, staff and post docs
  - Annual budget about \$2.4 billion
  - San Francisco's second largest employer





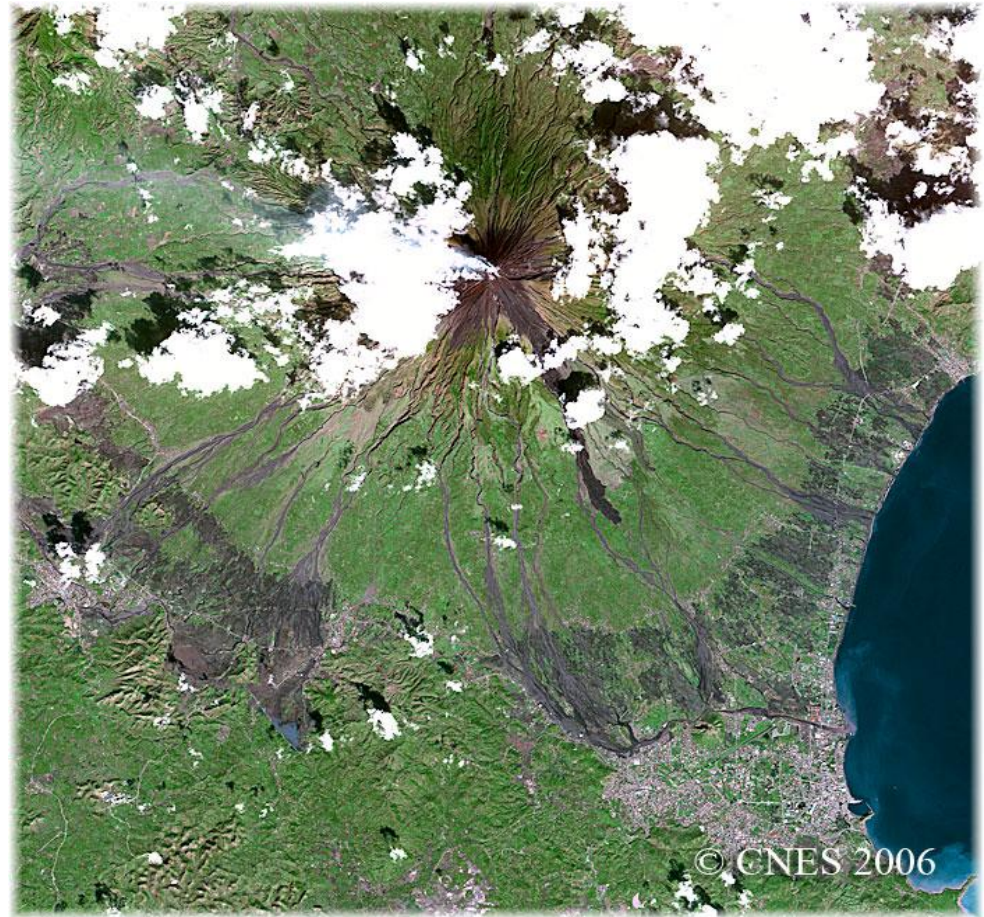
**This is not going to be a presentation on database tuning,  
but on load testing.**

Stressed out due to performance issues?



# OVERVIEW

- ▶ What is load testing?
- ▶ Types of load tests
- ▶ Purpose of load testing
- ▶ Developing the load test
- ▶ Load test design
- ▶ Load testing tools
- ▶ The test itself
- ▶ Analyze and report
- ▶ Conclusions
- ▶ DEMO



# What *is* Load Testing?



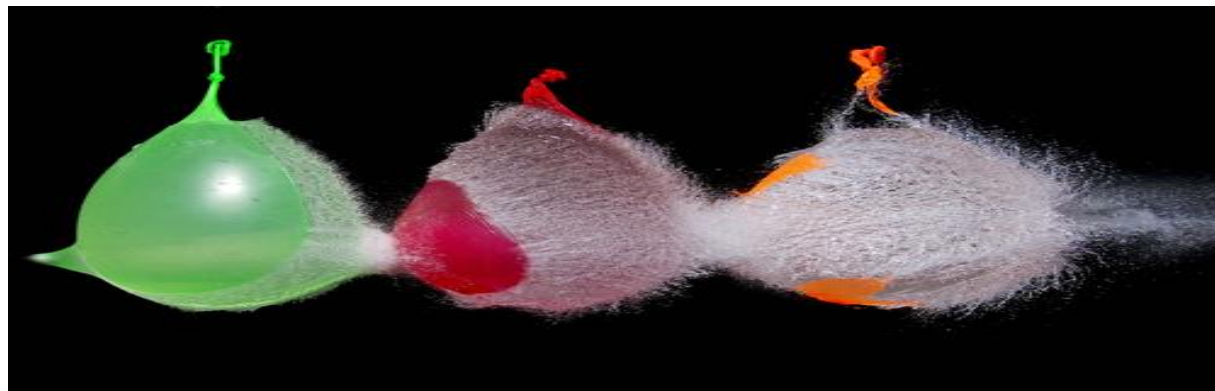
# Types of Load Tests

- ▶ **Transaction response testing**
  - Order entry must complete within 8 seconds
  - AP query must return results within 5 seconds
  - PDF attachment must upload within 5 seconds
- ▶ **End user experience testing**
  - Run full-scale web load test while a subset of users logs into system to conduct normal work activity
  - Have a subset of end users log into the system in the middle of load test to gauge performance
- ▶ **Stress test**
  - Multiple users logging into the system at one time (100 users log in at one time)
  - Users logging into the system very rapidly (e.g. 1 user every second)
  - Extended concurrency times (25 users remain logged into system running heavy transactions for an extended period)



# *Why* load test?

- ▶ After a software/hardware upgrade?
- ▶ Does the application perform acceptably with a minimum amount of users on the system?
- ▶ Does the application respond quickly with additional concurrent users?
- ▶ Are there hardware bottlenecks in the system?
- ▶ Can the application handle a growing number of users/data?
- ▶ What is the maximum amount of load the system can handle?

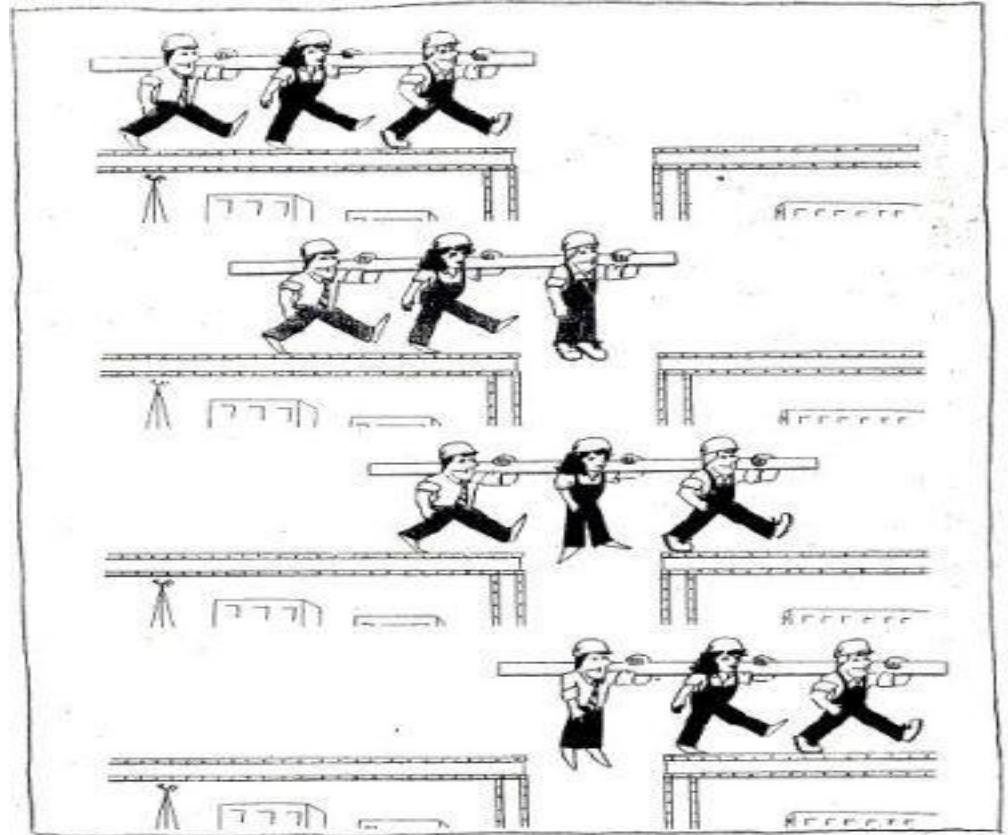


# Developing the Load Test



# Load Test Team

- ▶ Application owner(s)
- ▶ Database administrator(s)
- ▶ Web server administrator(s)
- ▶ Network administrator(s)
- ▶ System administrator(s)
- ▶ Developer(s)
- ▶ Project manager(s)



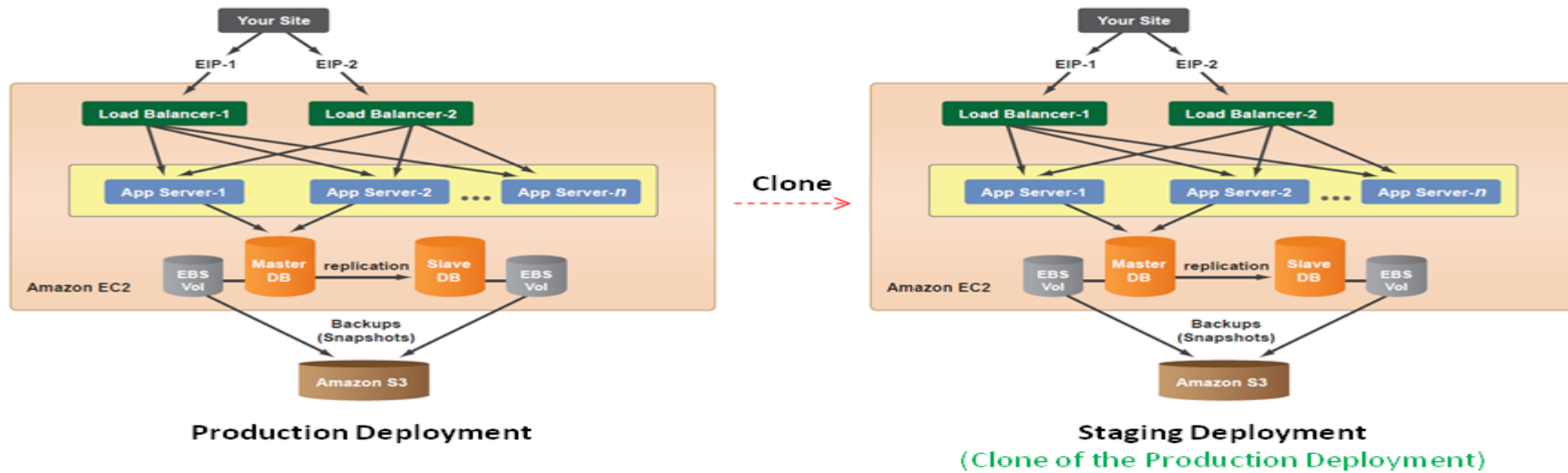
# Load Test Criteria Identification



# Load Test Criteria Identification (cont'd)

## ▶ Test environment

- Close to production as possible (ideal)
- How to adjust for inconsistencies between test and production
- Building environment
- Testing the test environment



# Load Test Criteria Identification (cont'd)

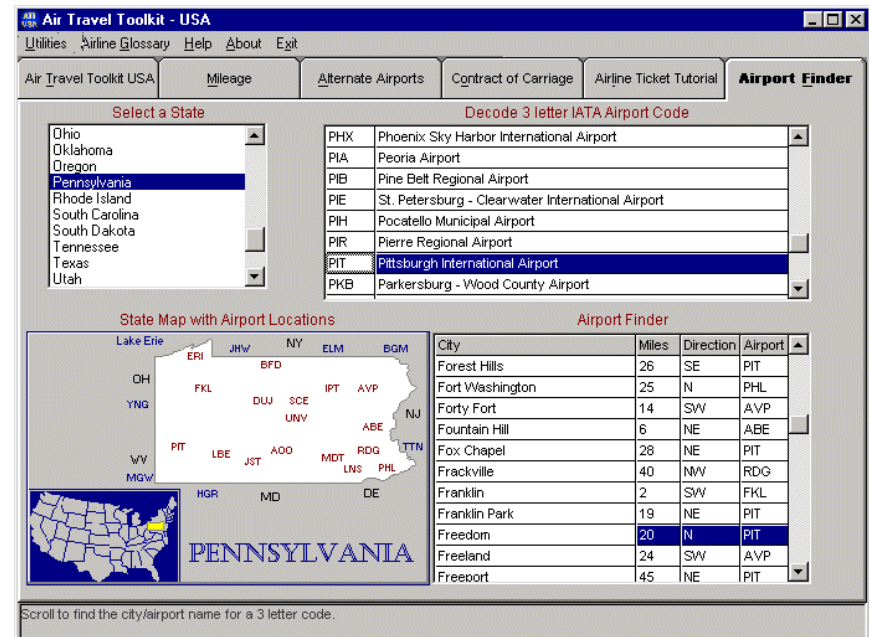
- ▶ Performance acceptance criteria
  - Service Level Agreements (SLAs)
  - End user expectations
  - Business requirements
  - Workload – key scenarios representing reality
  - Performance optimization objectives
  - Scalability
  - Range of accepted workload conditions
  - Resource utilization objectives



# Load Test Criteria Identification (cont'd)

## ▶ Application Scenarios

- What aspect of application going to test?
  - Breakdown into separate scenarios or tests?
  - Actions to be performed?
- Example – Book a flight:
  - Log into application
  - Search airline flights
  - Book flight
  - Pay for flight with credit card
  - Logout of application
- Scenario mix
- Screenshots of scenario



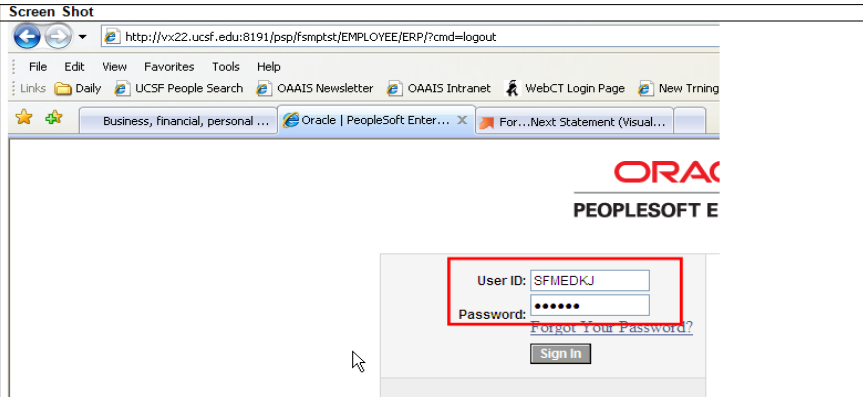

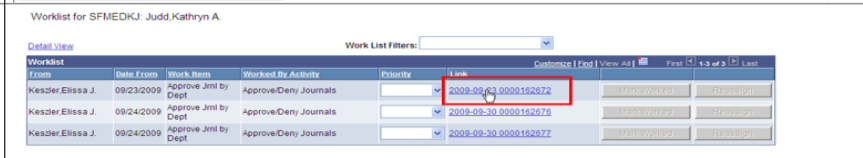
State Map with Airport Locations

| City            | Miles | Direction | Airport |
|-----------------|-------|-----------|---------|
| Forest Hills    | 26    | SE        | PIT     |
| Fort Washington | 25    | N         | PHL     |
| Forty Fort      | 14    | SW        | AVP     |
| Fountain Hill   | 6     | NE        | ABE     |
| Fox Chapel      | 28    | NE        | PIT     |
| Frackville      | 40    | NW        | RDG     |
| Franklin        | 2     | SW        | FKL     |
| Franklin Park   | 19    | NE        | PIT     |
| Freedom         | 20    | N         | PIT     |
| Freeland        | 24    | SW        | AVP     |
| Freeport        | 45    | NE        | PIT     |

Scroll to find the city/airport name for a 3 letter code.

# Load Test Criteria Identification (cont'd)

- ▶ Application Scenarios (cont'd)
  - Screenshot example:

| <b>Step</b><br>1) Login as SFMEDKJ/PASSWORD | <b>Screen Shot</b><br>  |                      |                       |           |                       |             |          |             |          |                   |            |                      |                       |  |                       |             |          |                   |            |                      |                       |  |                       |             |          |                   |            |                      |                       |  |                       |             |          |
|---|---|----------------------|-----------------------|-----------|-----------------------|-------------|----------|-------------|----------|-------------------|------------|----------------------|-----------------------|--|-----------------------|-------------|----------|-------------------|------------|----------------------|-----------------------|--|-----------------------|-------------|----------|-------------------|------------|----------------------|-----------------------|--|-----------------------|-------------|----------|
| <b>Step</b><br>2) Click 'Worklist'          | <b>Screen Shot</b><br>   |                      |                       |           |                       |             |          |             |          |                   |            |                      |                       |  |                       |             |          |                   |            |                      |                       |  |                       |             |          |                   |            |                      |                       |  |                       |             |          |
| <b>Step</b><br>3) Select first link.        | <b>Worklist for SFMEDKJ, Judd, Kathryn A.</b><br> <table border="1"><thead><tr><th>From</th><th>Date From</th><th>Work Item</th><th>Worked By/Activity</th><th>Priority</th><th>Link</th><th>Mark Worked</th><th>Reassign</th></tr></thead><tbody><tr><td>Keszler,Elissa J.</td><td>09/23/2009</td><td>Approve Jmli by Dept</td><td>Approve/Deny Journals</td><td></td><td>2009-09-30-0000162872</td><td>Mark Worked</td><td>Reassign</td></tr><tr><td>Keszler,Elissa J.</td><td>09/24/2009</td><td>Approve Jmli by Dept</td><td>Approve/Deny Journals</td><td></td><td>2009-09-30-0000162876</td><td>Mark Worked</td><td>Reassign</td></tr><tr><td>Keszler,Elissa J.</td><td>09/24/2009</td><td>Approve Jmli by Dept</td><td>Approve/Deny Journals</td><td></td><td>2009-09-30-0000162877</td><td>Mark Worked</td><td>Reassign</td></tr></tbody></table> | From                 | Date From             | Work Item | Worked By/Activity    | Priority    | Link     | Mark Worked | Reassign | Keszler,Elissa J. | 09/23/2009 | Approve Jmli by Dept | Approve/Deny Journals |  | 2009-09-30-0000162872 | Mark Worked | Reassign | Keszler,Elissa J. | 09/24/2009 | Approve Jmli by Dept | Approve/Deny Journals |  | 2009-09-30-0000162876 | Mark Worked | Reassign | Keszler,Elissa J. | 09/24/2009 | Approve Jmli by Dept | Approve/Deny Journals |  | 2009-09-30-0000162877 | Mark Worked | Reassign |
| From  | Date From   | Work Item            | Worked By/Activity    | Priority  | Link                  | Mark Worked | Reassign |             |          |                   |            |                      |                       |  |                       |             |          |                   |            |                      |                       |  |                       |             |          |                   |            |                      |                       |  |                       |             |          |
| Keszler,Elissa J.                           | 09/23/2009  | Approve Jmli by Dept | Approve/Deny Journals |           | 2009-09-30-0000162872 | Mark Worked | Reassign |             |          |                   |            |                      |                       |  |                       |             |          |                   |            |                      |                       |  |                       |             |          |                   |            |                      |                       |  |                       |             |          |
| Keszler,Elissa J.                           | 09/24/2009  | Approve Jmli by Dept | Approve/Deny Journals |           | 2009-09-30-0000162876 | Mark Worked | Reassign |             |          |                   |            |                      |                       |  |                       |             |          |                   |            |                      |                       |  |                       |             |          |                   |            |                      |                       |  |                       |             |          |
| Keszler,Elissa J.                           | 09/24/2009  | Approve Jmli by Dept | Approve/Deny Journals |           | 2009-09-30-0000162877 | Mark Worked | Reassign |             |          |                   |            |                      |                       |  |                       |             |          |                   |            |                      |                       |  |                       |             |          |                   |            |                      |                       |  |                       |             |          |



# Load Test Profile

## ▶ Performance Characteristics

- How many users have been concurrently logged into the system at the busiest time?
- How many concurrent users will be used for the test?
- How will users be divided among the script tasks to be tested, i.e. test 4 actions during test and divide users among scripts – 100 for script 1, 200 for script 2, 50 for script 3 and 150 for script 4.
- Number of users: the total number of concurrent and simultaneous users who access the application in a given time frame
- Rate of requests: the requests received from the concurrent load of users per unit time.
- Patterns of requests: A given load of concurrent users may be performing different tasks using the application. Patterns of requests identify the average load of users, and the rate of requests for a given functionality of an application.
- Will user pause before re-logging in again to perform action, i.e. wait 30 seconds before logging in again?
- How long will users spend on each web page of application during test (think time)
- How long will the test run?

# Application Usage Patterns

## ▶ Scenario

- Search and book flight.
- Search for flights without booking.
- Change an existing reservation.
- Ticket agent running a report.

| Script                             | % of users | Total |
|------------------------------------|------------|-------|
| Search and book flight             | 20         | 200   |
| Search for flights without booking | 60         | 600   |
| Change an existing reservation     | 10         | 100   |
| Ticket agent running a report      | 10         | 100   |
| Total                              | 100        | 1000  |

# Load Test Profile Considerations

- ▶ **User activity**
  - Some transactions occur more often than others and therefore should be a larger part of the test data and scenarios.
- ▶ **Ramp-up time**
  - A major component of stress on a site/application is how many and how quickly users log into the system.
- ▶ **Ramp-down time**
  - Load test ramp down implies gradually stopping the threads during a load run in order to detect memory leaks and check system recovery after the application has reached a threshold.
- ▶ **Think Times**
  - The time it takes for a user to respond to an application page has a significant impact on the number of users the application can support without performance issues.
- ▶ **Run Length**
  - The length of the actual load test can vary based on many factors. You should aim to run the test long enough to get a representative example of true system performance.
- ▶ **Workload/users**
  - Calculating the number of users for the load test is critical in order to have a valid test which accurately forecasts performance.

# Load Profile

| Scenario 1 - 20 Users |                                    | Load Profile |  |      |           |                         |                         |                 |
|-----------------------|------------------------------------|--------------|--|------|-----------|-------------------------|-------------------------|-----------------|
| Script                | Action                             | # Iterations | Think Time sec   | User | Pacing    | # Users Ramp Up         | # Users Ramp Down       | Duration (mins) |
| Script01              | Search & Book Flight               | 1            | Random of recorded think time: 50% - 150%. Limit 10 seconds. | 10   | 0 Seconds | 1 User every 10 seconds | 1 User every 10 seconds | 30              |
| Script02              | Search for flights without booking | 1            | Random of recorded think time: 50% - 150%. Limit 10 seconds. | 7    | 0 Seconds | 1 User every 10 seconds | 1 User every 10 seconds | 30              |
| Script03              | Change existing reservation        | 1            | Random of recorded think time: 50% - 150%. Limit 10 seconds. | 3    | 0 Seconds | 1 User every 10 seconds | 1 User every 10 seconds | 30              |

# Load Test Design

- ▶ Application accounts are created for the test with appropriate privileges.
- ▶ Test data feeds are properly implemented for testing. Test data feeds are data which is inputted into the system during the load test to create variances in activity thus mimicking real life.
- ▶ Validation of scenario transaction on the webserver, database, application server and hosting server.
- ▶ Special data used in application which needs to be incremented for proper execution such as product ID.



# Test IDs

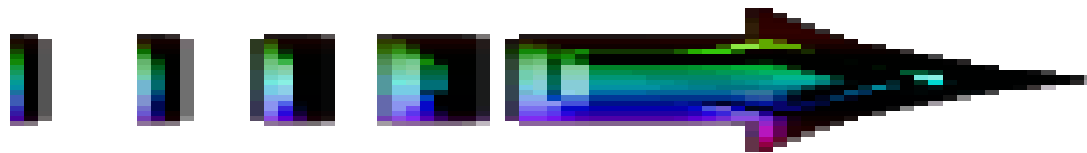
- ▶ Create test id(s)
- ▶ Application owner role
- ▶ Appropriate privileges
- ▶ ID data masking
- ▶ Unique ids
- ▶ Parameterization of values



| Login ID | Password |
|----------|----------|
| Test040  | pass01   |
| Test041  | pass01   |
| Test042  | pass01   |
| Test043  | pass01   |
| Test044  | pass01   |
| Test045  | pass01   |
| Test046  | pass01   |
| Test047  | pass01   |
| Test048  | pass01   |
| Test049  | pass01   |
| Test050  | pass01   |
| Test051  | pass01   |
| Test052  | pass01   |
| Test053  | pass01   |
| Test054  | pass01   |
| Test055  | pass01   |
| Test056  | pass01   |
| Test057  | pass01   |
| Test058  | pass01   |
| Test059  | pass01   |
| Test060  | pass01   |
| Test061  | pass01   |
| Test062  | pass01   |
| Test063  | pass01   |
| Test064  | pass01   |
| Test065  | pass01   |
| Test066  | pass01   |
| Test067  | pass01   |
| Test068  | pass01   |
| Test069  | pass01   |
| Test070  | pass01   |
| Test071  | pass01   |
| Test072  | pass01   |
| Test073  | pass01   |

# Test Data Feeds

- ▶ What is test data?
- ▶ Where is test data used?
- ▶ How to use test data
- ▶ Creation of test data
- ▶ Parameterization of test data
- ▶ Unique/non-unique data
- ▶ Oracle Flashback
- ▶ Data restore for non-Oracle systems

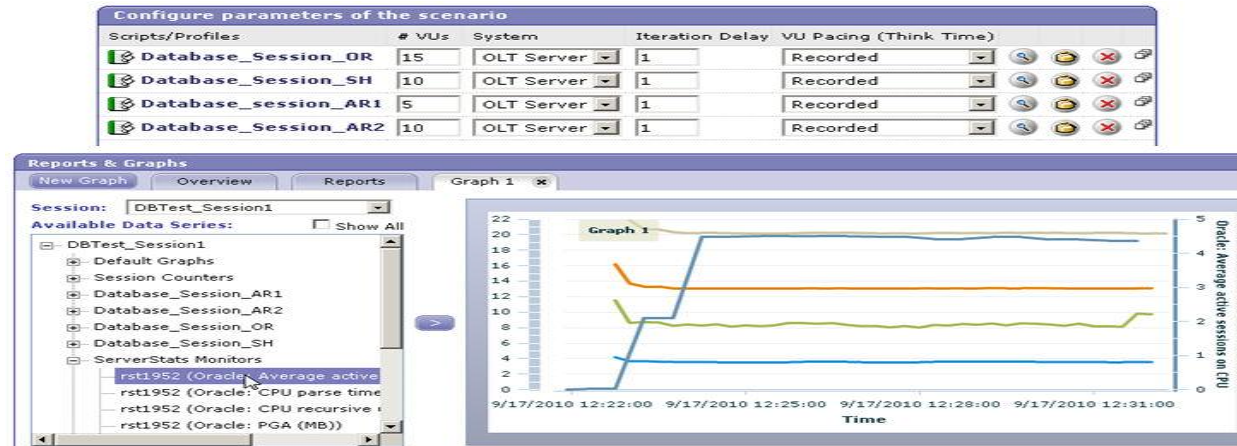






# Oracle Real Application Testing VS Load Testing

- ▶ What is Oracle Real Application Testing (RAT)?
- ▶ Why RAT should not be used by itself for formal load testing.
- ▶ How is RAT used?
- ▶ Oracle Real Application Testing Suite
- ▶ How to use Real Application Testing Suite with RAT to create load tests



# Monitoring the Load Test

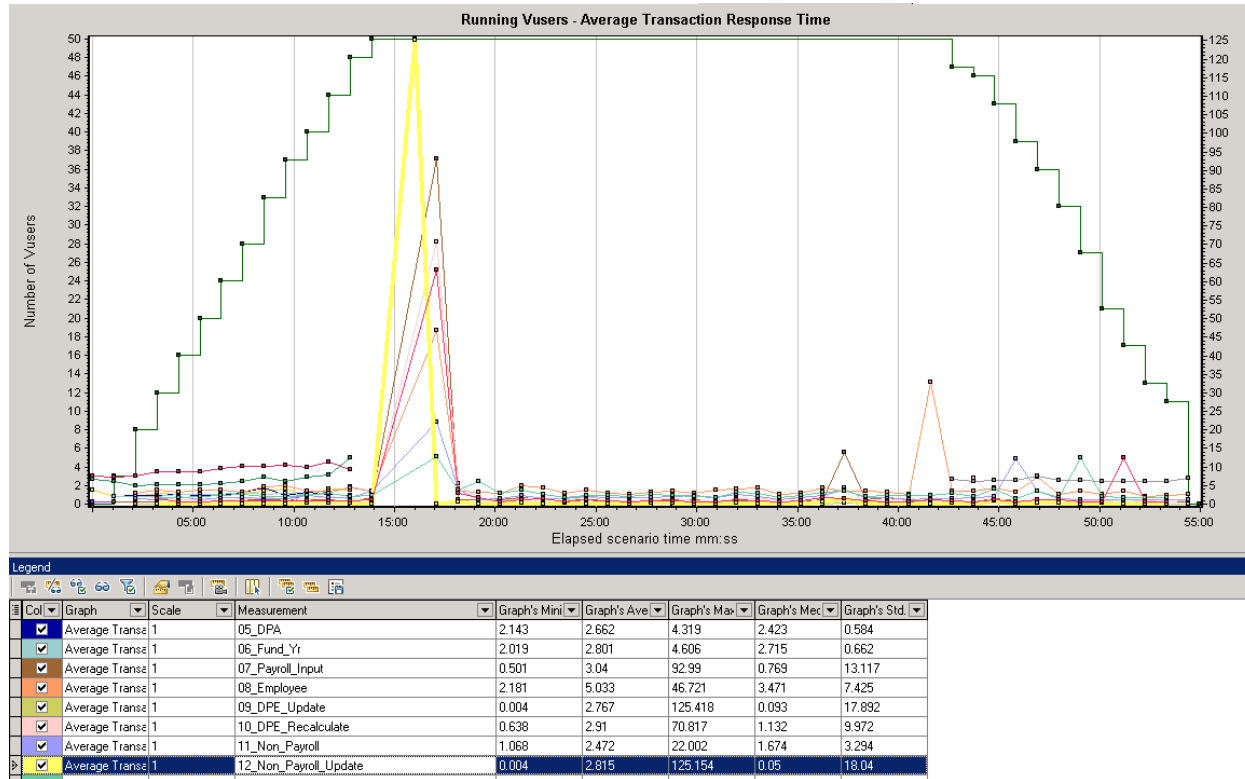
- ▶ Identify metrics to be used for the load test
  - OS metrics which could include CPU, memory, disk I/O and network I/O statistics.
  - Database metrics to watch for are waits, top SQL, Buffer gets, etc.
  - Network metrics would include the overall health of your routers, switches and gateways.
  - Webserver metrics would include hits per second, number of requests, etc.
- ▶ Identify team members who will monitor during the test
  - What areas will be monitored?
  - What tools will be used?
  - How will the tools be used to monitor (invasive/non-invasive?)
  - How will the results be reported at the conclusion of the test?



# Analyze and Report Test Results

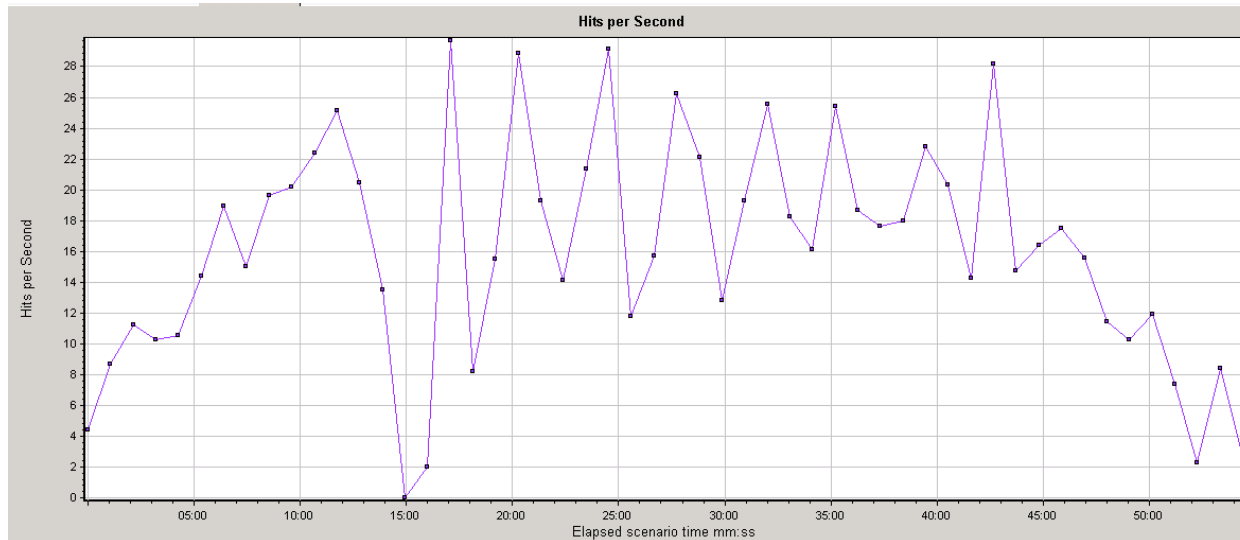
- ▶ Pinpoint bottlenecks found during load test
  - Examples:
    - Checkout should not take longer than 4 seconds under 15 hits per second workload
    - CPU utilization for the test should not exceed 80% during any transaction
- ▶ Transactions in average response time graph breach the performance acceptance criteria.
- ▶ Response time that is worse than what the SLA for the customer allows.
- ▶ Correlate transactions with other graphs and measurements to find bottlenecks.
- ▶ Involve developers, system and network administrators in investigation.
- ▶ Overlay response time graph with monitoring graphs and look for performance trends.
- ▶ Check standard deviation of transactions if transactions behave inconsistently identify the cause.
- ▶ Transactions availability: (count of pass transactions/count of total transactions) x 100. Usually applications have around 99.9% availability. Lower values don't necessarily indicate an issue, but should be verified why lower.

# Load Test Report Graphs



- ▶ LoadRunner graph showing a performance irregularity occurred between 14 and 18 minutes into the test

# Hits per Second Graph - LoadRunner

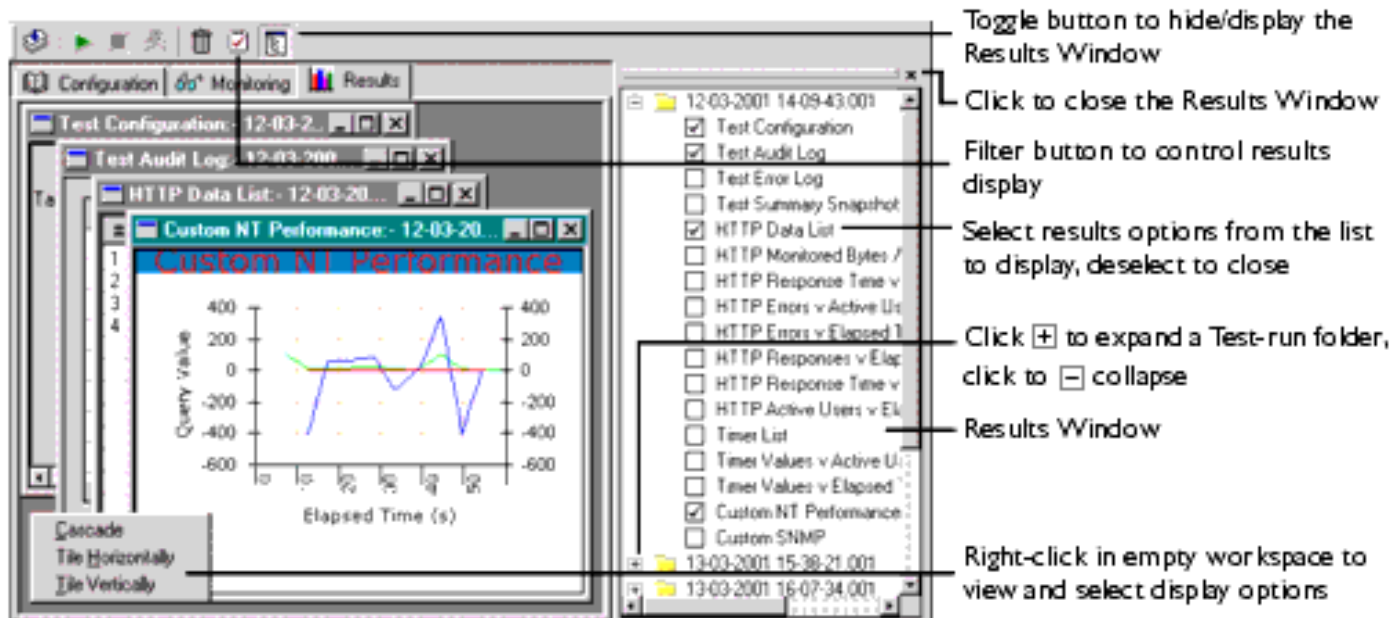


- ▶ Same load test as previous slide – Note drop off in hits at 15 minute point in test

# Reporting Audience

- ▶ **Management/Project Managers/Team Leads?**
  - Technical aptitude
  - Length of report
  - Type of report
- ▶ **Suggested Graphs to include in report**
  - Running Users – Shows the impact of ramp-up and ramp-down as well as when errors begin to occur during the load test.
  - Hits per Second – Shows the amount of load generated by virtual users in terms of hits to web servers.
  - Throughput – Shows the amount of load virtual users create on network resources – raw amount of bytes client receives each second as a result of hits on the web server.
  - Average Transaction Response Time – Shows the actual load generated by each transaction throughout the entire load test.

# OpenSTA Graph Reporting



The screenshot displays the OpenSTA software interface. The main window is titled "Custom NT Performance" and contains a line graph with "Query Value" on the y-axis (ranging from -600 to 400) and "Elapsed Time (s)" on the x-axis. The graph shows several data series with fluctuating values. To the right of the graph is a "Results Window" containing a list of test results for two different test runs. The list includes various metrics such as "Test Configuration", "Test Audit Log", "Test Error Log", "Test Summary Snapshot", "HTTP Data List", "HTTP Monitored Bytes /", "HTTP Response Time v", "HTTP Errors v Active Us", "HTTP Errors v Elapsed T", "HTTP Responses v Elap", "HTTP Response Time v", "HTTP Active Users v EL", "Timer List", "Timer Values v Active U", "Timer Values v Elapsed", "Custom NT Performance", and "Custom SNMP".

Annotations on the right side of the image provide instructions for interacting with the Results Window:

- Toggle button to hide/display the Results Window
- Click to close the Results Window
- Filter button to control results display
- Select results options from the list to display, deselect to close
- Click  to expand a Test-run folder, click to  collapse
- Results Window
- Right-click in empty workspace to view and select display options

At the bottom left of the graph window, there are window management options: Cascade, Tile Horizontally, and Tile Vertically.

- ▶ Graphs can be customized for presentation purposes
- ▶ Data from graphs can be exported to Excel for further analysis

# Load Test Summary

- ▶ Load test summary report
  - Top 5 Slowest Pages
    - Slowest pages encountered during load test
  - Top 5 Slowest Transactions
    - Do not meet the performance criteria defined at test inception
  - Top 5 Slowest SQL Statements
    - Slowest SQL which is typically 80% of performance problems

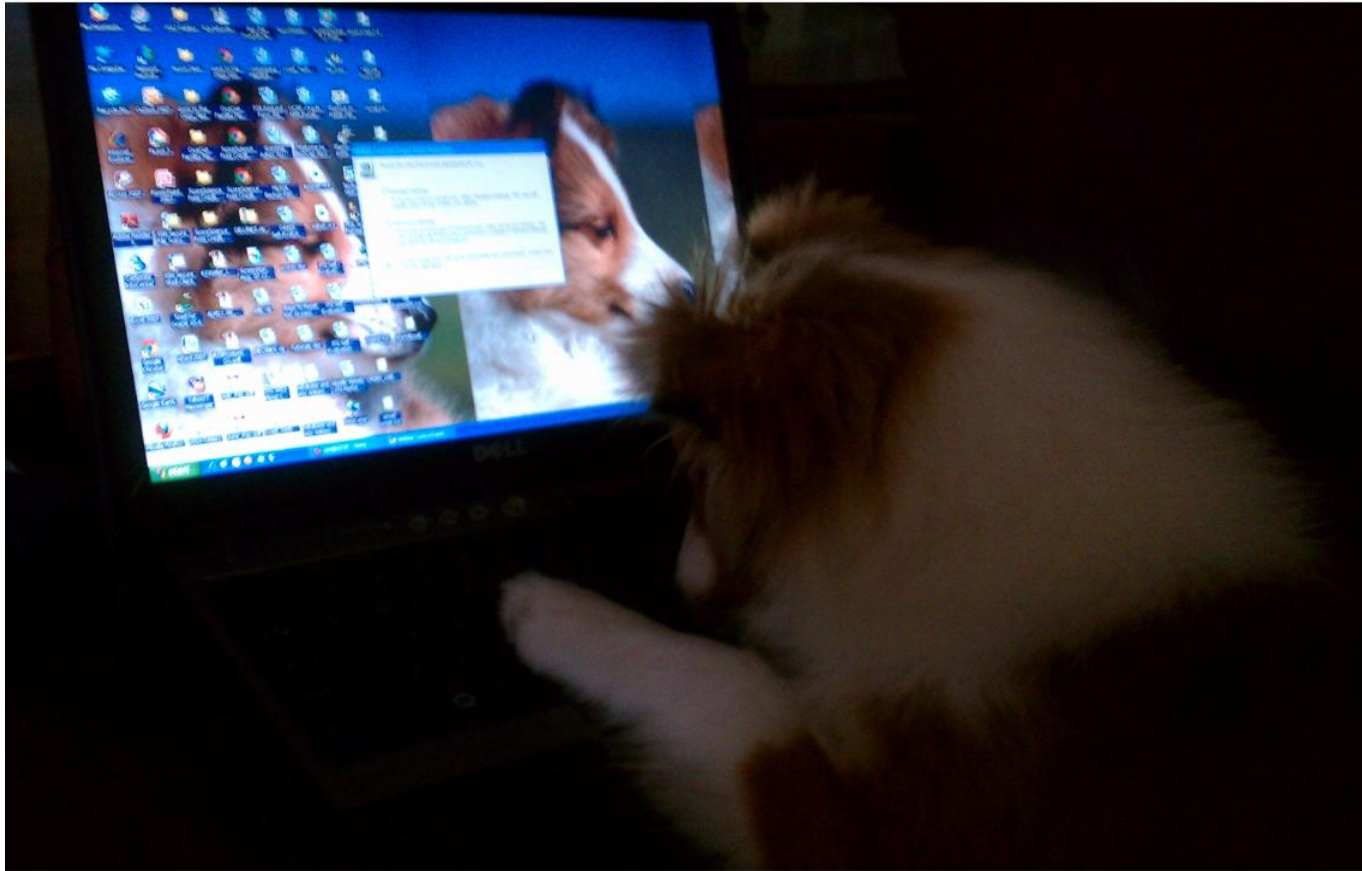




# Find Bottlenecks, Tune System - Retest?



# CONCLUSIONS



# DEMONSTRATION