Regression & Load Testing BI EE 11g

Venkatakrishnan J
Who Am I?

Venkatakrishnan Janakiraman

- Over 8+ Years of Oracle BI & EPM experience
- Managing Director (India), Rittman Mead India
- Blog at [http://www.rittmanmead.com/blog](http://www.rittmanmead.com/blog)
  - Old & defunct blog [http://oraclebizint.wordpress.com](http://oraclebizint.wordpress.com)
- Oracle ACE
- EPM/BI Specialization
- Twitter
  - @krisvenkat
About Rittman Mead

• Oracle BI and DW specialized partner
• World leading specialist partner for technical excellence, solutions delivery and innovation in Oracle BI
• 50+ consultants all expert in Oracle BI and DW
• Global organisation, local delivery teams
  ‣ Offices in UK, US, Europe, India, Australia
• Skills in broad range of supporting Oracle tools:
  ‣ OBIEE
  ‣ OBIA
  ‣ ODI
  ‣ OWB
  ‣ Essbase, Oracle OLAP
  ‣ GoldenGate
  ‣ Exadata
  ‣ Exalytics
Agenda

• Overview of Testing
• Types of Testing & their importance in BI Rollouts
  › Unit Testing
  › Regression Testing
  › Load Testing
  › Stress Testing
• Regression Testing in BI EE
• Load Testing in BI EE
• Tools Available for Automated Testing
  › Importance of Automation
• RM Testing Framework/Accelerator
What is Testing?

• Testing  
  ‣ Means to find bugs/issues
• Different types of Bugs  
  ‣ Product Bugs  
  ‣ Design Bugs  
  ‣ Implementation Bugs
• For any software  
  ‣ BUGS are the only constant  
  ‣ There can never be bug-free software  
  ‣ BI EE is no different
• Treat BI EE as just another web application  
  ‣ Ensure Testing is part of every phase in a rollout
Testing - How important is it for BI Rollouts?

• Testing is fundamental to a BI Roll-out
• BI is the primary interface for data to end-users
  ‣ So Testing is critical
  ‣ Testing strategy used determines success/failure
  ‣ Testing is key for user adoption/perception of the BI System
• Failure of a BI System can be attributed 50% to bad testing strategies
BI EE Testing Areas

• Multiple Areas of Testing (End-user Perspective)
  ‣ Security Testing (Security properly being applied or not)
  ‣ Report Navigation Testing (Drills properly working or not)
  ‣ User Interaction Testing (Prompts being applied correctly or not)
  ‣ User Standards Testing (Look & Feel, Report Layouts, Number Formats)
  ‣ Scalability Testing
  ‣ Performance Testing
  ‣ Integration Testing (integration with other systems)

• Questions to be answered?
  ‣ How often do we test the above?
  ‣ During every migration or during development or both?

• Testing should be in all phases
BI EE Rollout - Iterative Testing Phases

BI Rollout

Discover ➔ Design ➔ Prototype ➔ Configure ➔ Validate ➔ Deploy ➔ Sustain

End-to-End Testing Solution

Test Management

Functional Test

Performance Test

Tuning

* - Siebel CRM Bookshelf

T : +44 (0) 8446 697 995 or (888) 631 1410 (USA)   E : enquiries@rittmanmead.com   W: www.rittmanmead.com
Common Testing Methodologies

• Relevant Testing Phases for BI
  ‣ Unit Testing
  ‣ Regression Testing
  ‣ Load Testing
  ‣ Stress Testing

• General practice in BI EE Rollouts (based on my experience)
  ‣ Good Unit Testing Strategy
  ‣ But No proper Regression Testing Strategy
  ‣ And No proper Load Testing Strategy
Regression Testing - What is it?

- Identifies new bugs introduced by a new feature
  - End-user specific
- Identify whether a bug-fix did not lead to other bugs
- Very important in a BI Roll-out
  - All the more important when the BI System is live for a while
    - Many users cannot tolerate change in behaviour, layouts as their daily activity is based on the system
    - As important as Regression testing a Web Application
      - Even more important for BI Apps roll outs
- Regression Testing
  - Involves identifying all test cases
  - Run all test cases during every migration to Production
  - No Production Migration unless all the test cases complete 100% successfully
Regression Testing - Example 1

• A new javascript has been implemented to clear the default selections in a dashboard prompt
  ‣ Java Script based feature implementations are common
  ‣ Require very careful testing
  ‣ Might not work across multiple browsers & multiple versions of the same browser
  ‣ Can break during upgrades
  ‣ Very important to document and repeat the functionality of the feature post an upgrade/migration
Regression Testing - Example 2

• A new set of reports/dashboards or Dashboard pages are going live
  ‣ Common migration practice is to copy the web catalog content over using catalog manager (or Patching in 11g)
  ‣ Apply security permissions manually post migration
  ‣ How do we test that the above migration has not affected security of other objects?
Regression Testing - Example 3

- A new set of RPD enhancements are going live
  - Common practice is to merge the objects from Dev to Prod/UAT (& then patching)
  - Or even manually copying/renaming objects over
  - How can we know for sure that none of the existing reports/dashboards have been impacted because of this?
  - How do we test this?
Regression Testing - Example 4

• Migrating to a new release (say 10g to 11g) or even 10.1.3.4.1 to 10.1.3.4.2
• How do we ensure all the customizations done to a dashboard work properly?
  ‣ Java Scripts
  ‣ Security
    - Complex Group based security
  ‣ Look and feel - UI
So Regression Testing is Important - But....

- Primary driver for Regression Testing not being done completely in a lot of projects
  - Lack of time (End users expect faster turn around time)
  - Lack of resources (Lots of end points need to be tested)
  - Monotony - Yes it can be too boring to do regression testing for each and every change
  - Possibilities of Human Error
Automated Regression Testing

- To avoid the rigours of manual Regression Testing
  - Automated tools available now
- Some Tools available to do this
  - Selenium (Open Source)
  - Oracle Application Testing Suite
  - QTP
  - Winrunner
  - .... etc
Regression Testing - Selenium

• Selenium
  ‣ Open Source IDE
  ‣ Very light weight
  ‣ Available as a firefox add-on
  ‣ Can be completely programmed
  ‣ Supports XPath
  ‣ Cross Platform support
  ‣ No support for Load Testing
Selenium - Regression Testing

Demo
Selenium - Recording a Test

- Supports auto recording of click actions
  - Login
  - Opening a Dashboard
  - Changing prompt values
  - Prompt Multi-select
  - Drills
  - Navigations
  - Any click based actions
Selenium - Regression Test

- Things to watch out for
  - There is no native BI EE support
    - Though a new instanceconfig.xml setting has been introduced
  - So requires tweaking of the code generated by Selenium
  - More changes required when a Javascript based action is recorded
  - Does not work with the new BI EE 11g feature of drag/drop columns in a Dashboard
  - Page IDs generated by BI EE are unique
Regression Testing - Oracle Application Testing Suite

• OATS
  ▶ Acquisition of Empirix
  ▶ Supports all kinds of testing
    - Load Tests
    - Integration Tests
    - Regression Tests
  ▶ Based on Java
    - One of the main reasons behind the
      - ADF adoption
  ▶ Supports Recording through OpenScript
  ▶ Monitors page load times and other metrics
    - Not available in Selenium out of the box
OATS - Recording a Test

- Native support for ADF
- Supports auto recording of click actions
  - Login
  - Opening a Dashboard
  - Changing prompt values
  - Prompt Multi-select
  - Drills
  - Navigations
OATS - Regression Test

• Things that work unlike Selenium out of the box
  ‣ Native BI EE support
  ‣ No issues with BI EE Unique Page IDs
  ‣ Works well with many java script based prompts
  ‣ Extensive Java Support

• Things to watch out
  ‣ Requires some tweaking for complex java scripts (like dashboard navigation menus etc)
Load Testing

• Very different from Regression Testing
• Very important for BI EE rollouts
• Typically done before a first major production rollout
• Very important for capacity planning
• Testing Done at HTTP Protocol level
  ‣ No Java Scripts
  ‣ No browser dependence
  ‣ Pure scalability testing
Load Testing - Why.....

• Typical rollouts will have capacity planning done much earlier. But....
  ‣ How can we judge whether BI Server can scale?
  ‣ How can we judge the number of Database connections?
  ‣ How can we judge the number of SQL queries?
  ‣ How can we find out whether there are poorly designed reports?
  ‣ How can we find out whether clustering is needed?
Load Testing - Example 1

- A well capacity planned BI EE system is going live
  - Estimated 200 concurrent users
  - How do we estimate Weblogic JVM memory required?
  - How do we estimate whether existing BI Server settings are sufficient?
  - How do we know where the latency is?
  - How do we know the exact breaking point, in case load test fails?
Load Testing - Example 2

- A system that is live for a year
  - Requirement to add 100 more Analyst users as part of enterprise roll-out
  - How do we pro-actively determine new user base will not affect performance for existing users?
  - How do we know where the performance bottleneck will be?
  - When there is a call to add a BI EE Component cluster, how do we justify that need?
Load Testing - 2 Parts

- 2 aspects of any Load Testing
  - Simulating the Load
    - Monitoring the metrics during the load the test
  - Simulating the Load
    - Done through an external tool
      - OATS
      - Apache JMeter etc
- Monitoring of Metrics
  - Usage Tracking
  - Using DMS
Load Testing - Key aspects

- Important aspect to do load testing
  - Availability of Metrics from the BI System
  - Ability to monitor Metrics
  - Ability to store the relevant Metrics
  - Monitor Metrics as a percentage of users, sessions
  - To an extent simulate the load similar to the behaviour analysis pattern of end users
Load Testing Tool - Key Aspects

- Load Testing Tool
  - Ability to increase the number of users/threads
  - Ability to do impersonation/login as multiple users
  - Ability to vary prompt values in a dashboard
  - Ability to simulate user behaviour pattern through the tool
Load Testing Tool - Architecture
Load Testing - Metrics in EM

- 100+ metrics available for monitoring
- Available through EM
  - Only real time monitoring
  - No ability to persist them and track them over a period of time
  - No ability to use BI EE reporting capabilities to report on these metrics
Load Testing - Metrics in DMS

• All Metrics made available through DMS - Diagnostic Monitoring Service

• No MBean access to metrics
  ▪ But can be obtained through WLST
  ▪ No other API currently apart from WLST

• 11.1.1.3 & 11.1.1.5 - DMS metrics monitoring is slightly different
  ▪ 11.1.1.3 - DMS shows all metrics when queried
  ▪ 11.1.1.5 - DMS shows metrics only when there is a change
Load Testing - Metrics in DMS

- Sample WLST Method
  - `displayMetricTables` method to get at a specific metric table
  - `displayMetricTableNames` to get all Metric tables available for monitoring
Load Testing - Metrics through OPMN

• DMS exposes non-J2EE metrics through Java.
• DMS not ideal (not that flexible) for non-J2EE metrics but preferred for Java metrics
• Same metrics can be obtained from OPMN utility

• OPMN sample script
  ‣ opmnctl metric op=list
  ‣ opmnctl metric op=query COMPONENT_NAME=coreapplication_obis1 format=xml
• No remote API for this
Load Testing - OATS

- Like Regression Testing, a recorder available to record test scripts
- Records only HTTP calls
- No browser based calls recorded
- All prompt actions, user actions
  ‣ triggered through POST method
Load Testing - OATS

• A web test manager available to replay scripts
• Number of users can be increased/decreased
• All users - Virtual users - Ideally OATS should be in a separate box
• Can be scripted to simulate impersonation
• Provides metrics of page load times etc
  ▸ No BI EE Metrics available out of the box
Load & Regression Testing Scripts

• Regression Testing
  › All driven from browser
  › Prompt values varied by simulating click actions
  › Navigations done through click actions
  › Java Script actions simulated as well

• Load Testing
  › All driven directly through HTTP
  › No browser involved
  › Prompt Values pushed through POST methods
  › Navigations done through GET method (URL Navigation)
Rittman Mead - Load Testing Accelerator

• Custom framework/accelerator built by Rittman Mead
• Can plug into any load testing tool
  ‣ Apache JMeter
  ‣ Oracle Application Testing Suite
  ‣ Load Runner
• Gathers metrics as a function of load
  ‣ Both UI Metrics provided by tools
  ‣ Component DMS metrics
• Fully automated
Rittman Mead - Load Testing Accelerator

- No extra load testing tool license needed
  - Apache JMeter
- Can plug into existing licensed load testing tools
- Input Parameters
  - User List
  - Each Dashboard page prompt selection list
    - For new systems, ability to generate the random list
- System Reports
  - Comprehensive set of reports/dashboards
    - Component level metrics
    - Also can use tool based metrics
Rittman Mead - Load Testing Accelerator
Rittman Mead - Load Testing Framework

Demo - Apache JMeter
Finding Rittman Mead at the conference and beyond

• Drop in a business card
  ‣ Get access to all of Rittman Mead’s presentations and white papers from the OUG and also Openworld and other global community events
  ‣ Latest insight into Exalytics

• One-to-one sessions
  ‣ Find Morgan McCarthy or one of our team at the back of the room to schedule in a time

• Tweet us
  ‣ @rittmanmead

• Mail us
  ‣ info@rittmanmead.com

• And socially at the conference
  ‣ At the focus pubs Monday evening - find us for a beer!
Load Testing

Oracle BI EE 11g