Chapter 1: Performance Testing Fundamentals

- Identify test environment
- Identify acceptance criteria
- Plan and design tests
- Prepare test environment
- Record test plan(s)
- Run the tests
- Analyze results
- Perform tuning/Change configuration

```
bin      
  examples    
    jsp   
    templates   
  docs   
    resources   
  extras   
  lib   
    ext   
    junit   
  licenses   
  bin   
  src   
  printable_docs
```

187 directories
Chapter 2: Recording Your First Test
Template showing how to record

JMeter Configuration

- JMeter Server Proxy port is set to 8888, you may want to change this

Browser Configuration

Configure your browser so that it uses the JMeter proxy, set:

- the host of the machine that runs JMeter
- the port you setup in JMeter Server Proxy

Useful links

Local Area Network (LAN) Settings

Automatic configuration
Automatic configuration may override manual settings. To ensure the use of manual settings, disable automatic configuration.
- [ ] Automatically detect settings
- [ ] Use automatic configuration script

Address: [ ]

Proxy server
- Use a proxy server for your LAN (These settings will not apply to dial-up or VPN connections).
- Address: localhost  Port: 7000  Advanced
- Bypass proxy server for local addresses

[OK]  [Cancel]
## Aggregate Report

**Name:** Aggregate Report  

**Comments:**

- Write results to file / Read from file

### Results Table

<table>
<thead>
<tr>
<th>Label</th>
<th># Samples</th>
<th>Average</th>
<th>Median</th>
<th>90% Line</th>
<th>Min</th>
<th>Max</th>
<th>Error %</th>
<th>Throughput</th>
<th>KB/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>/</td>
<td>600</td>
<td>550</td>
<td>378</td>
<td>1123</td>
<td>263</td>
<td>3681</td>
<td>0.00%</td>
<td>8.4/sec</td>
<td>36.9</td>
</tr>
<tr>
<td>/ads/button</td>
<td>300</td>
<td>150</td>
<td>110</td>
<td>122</td>
<td>107</td>
<td>2347</td>
<td>0.00%</td>
<td>4.3/sec</td>
<td>2.5</td>
</tr>
<tr>
<td>/changes...</td>
<td>600</td>
<td>548</td>
<td>527</td>
<td>1090</td>
<td>264</td>
<td>2369</td>
<td>0.00%</td>
<td>8.5/sec</td>
<td>55.6</td>
</tr>
<tr>
<td>/username...</td>
<td>600</td>
<td>489</td>
<td>270</td>
<td>1017</td>
<td>264</td>
<td>3175</td>
<td>0.00%</td>
<td>8.5/sec</td>
<td>43.9</td>
</tr>
<tr>
<td>TOTAL</td>
<td>2100</td>
<td>475</td>
<td>272</td>
<td>1049</td>
<td>107</td>
<td>3681</td>
<td>0.00%</td>
<td>29.2/sec</td>
<td>136.9</td>
</tr>
</tbody>
</table>

---

**View Results Tree**

**Name:** View Results Tree  

**Comments:**

- Write results to file / Read from file

---

**Response data**

```
org.apache.http.NoHttpResponseException: The target server failed to respond
  at org.apache.http.impl.conn.DefaultHttpConnectionMetricsTracker$DefaultHttpConnectionPeer
    .reset(www.apache.org, 95)
  at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java, 715)
  at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java, 715)
  at org.apache.http.protocol.HttpRequestExecutor.execute(HttpRequestExecutor.java, 12)
  at org.apache.http.client.defaults.HttpRequestRetryDirector$1.execute(HttpRequestRetryDirector.java, 420)
  at org.apache.jmeter.protocol.http.sampler.HTTPHC4Sampler.sample(HTTPHC4Sampler.java, 32)
```

---

**Search:**

- Find
- Case sensitive
- Regular expression
Graph Results

Name: Graph Results
Comments:
Write results to file / Read from file
Filename Browse... Log/Display Only: Errors Successes Configure

Graphs to Display
- Data
- Average
- Median
- Deviation
- Throughput

1057 ms

No of Samples 2100 Latest Sample 270 Average 475
Deviation 398 Throughput 1,753.507/minute Median 272
Chapter 3: Submitting Forms
Chapter 4: Managing Sessions

```
HTTP Cookie Manager

Name: HTTP Cookie Manager
Comments:

Options
- Clear cookies each iteration?
  - Cookie Policy: compatibility
  - Implementation: HC3CookieHandler

User-Defined Cookies

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
<th>Domain</th>
<th>Path</th>
<th>Secure</th>
</tr>
</thead>
</table>

[Buttons: Add, Delete, Load, Save]
```
<table>
<thead>
<tr>
<th>HTTP URL Re-writing Modifier</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name:</strong> HTTP URL Re-writing Modifier</td>
</tr>
<tr>
<td><strong>Comments:</strong></td>
</tr>
<tr>
<td><strong>Session Argument Name:</strong> jsessionid</td>
</tr>
<tr>
<td><strong>Path Extension (use &quot;,,&quot; as separator)</strong></td>
</tr>
<tr>
<td><strong>Do not use equals in path extension (Intershop Enfinity compatibility)</strong></td>
</tr>
<tr>
<td><strong>Do not use questionmark in path extension (Intershop Enfinity compatibility)</strong></td>
</tr>
<tr>
<td><strong>Cache Session Id?</strong></td>
</tr>
</tbody>
</table>
Chapter 5: Resource Monitoring
http://localhost:8080  
12/15/14 6:09 PM
<table>
<thead>
<tr>
<th>Label</th>
<th># Samples</th>
<th>Average</th>
<th>Median</th>
<th>90% Line</th>
<th>Min</th>
<th>Max</th>
<th>Error %</th>
<th>Throughput</th>
<th>KB/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apple Home</td>
<td>900</td>
<td>1321</td>
<td>857</td>
<td>2554</td>
<td>518</td>
<td>7695</td>
<td>0.00%</td>
<td>8.6/sec</td>
<td>630.4</td>
</tr>
<tr>
<td>iTunes</td>
<td>900</td>
<td>1104</td>
<td>772</td>
<td>2083</td>
<td>542</td>
<td>5524</td>
<td>0.11%</td>
<td>8.6/sec</td>
<td>204.2</td>
</tr>
<tr>
<td>Featured</td>
<td>900</td>
<td>637</td>
<td>450</td>
<td>1321</td>
<td>336</td>
<td>5086</td>
<td>0.00%</td>
<td>8.9/sec</td>
<td>116.0</td>
</tr>
<tr>
<td>Songs</td>
<td>900</td>
<td>715</td>
<td>470</td>
<td>1676</td>
<td>329</td>
<td>12980</td>
<td>0.00%</td>
<td>8.0/sec</td>
<td>132.6</td>
</tr>
<tr>
<td>Albums</td>
<td>900</td>
<td>200</td>
<td>63</td>
<td>327</td>
<td>18</td>
<td>4607</td>
<td>0.00%</td>
<td>8.1/sec</td>
<td>108.5</td>
</tr>
<tr>
<td>TV Shows</td>
<td>900</td>
<td>672</td>
<td>436</td>
<td>1766</td>
<td>308</td>
<td>4895</td>
<td>0.00%</td>
<td>8.0/sec</td>
<td>132.1</td>
</tr>
<tr>
<td>Movies</td>
<td>900</td>
<td>214</td>
<td>56</td>
<td>317</td>
<td>19</td>
<td>12474</td>
<td>0.00%</td>
<td>8.1/sec</td>
<td>95.8</td>
</tr>
<tr>
<td>Movie Rentals</td>
<td>900</td>
<td>217</td>
<td>57</td>
<td>317</td>
<td>19</td>
<td>12429</td>
<td>0.00%</td>
<td>8.1/sec</td>
<td>102.0</td>
</tr>
<tr>
<td>Free Apps</td>
<td>900</td>
<td>287</td>
<td>61</td>
<td>428</td>
<td>16</td>
<td>12574</td>
<td>0.00%</td>
<td>8.1/sec</td>
<td>99.9</td>
</tr>
<tr>
<td>Paid Apps</td>
<td>900</td>
<td>322</td>
<td>62</td>
<td>832</td>
<td>17</td>
<td>4700</td>
<td>0.00%</td>
<td>8.1/sec</td>
<td>99.9</td>
</tr>
<tr>
<td>Music Videos</td>
<td>900</td>
<td>371</td>
<td>72</td>
<td>1014</td>
<td>17</td>
<td>12668</td>
<td>0.00%</td>
<td>8.2/sec</td>
<td>108.5</td>
</tr>
<tr>
<td>TOTAL</td>
<td>9900</td>
<td>549</td>
<td>341</td>
<td>1502</td>
<td>16</td>
<td>12880</td>
<td>0.01%</td>
<td>76.6/sec</td>
<td>1524.9</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Label</th>
<th># Samples</th>
<th>Average</th>
<th>Median</th>
<th>90% Line</th>
<th>Min</th>
<th>Max</th>
<th>Error %</th>
<th>Throughput</th>
<th>KB/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apple Home</td>
<td>90</td>
<td>2519</td>
<td>1857</td>
<td>3170</td>
<td>564</td>
<td>16798</td>
<td>0.00%</td>
<td>41.3/min</td>
<td>50.4</td>
</tr>
<tr>
<td>iTunes</td>
<td>90</td>
<td>2365</td>
<td>1814</td>
<td>3899</td>
<td>611</td>
<td>9138</td>
<td>0.00%</td>
<td>43.8/min</td>
<td>17.5</td>
</tr>
<tr>
<td>Featured</td>
<td>90</td>
<td>1333</td>
<td>1083</td>
<td>2203</td>
<td>329</td>
<td>9340</td>
<td>0.00%</td>
<td>45.9/min</td>
<td>10.0</td>
</tr>
<tr>
<td>Songs</td>
<td>90</td>
<td>1948</td>
<td>1285</td>
<td>2880</td>
<td>320</td>
<td>16510</td>
<td>0.00%</td>
<td>46.7/min</td>
<td>13.0</td>
</tr>
<tr>
<td>Albums</td>
<td>90</td>
<td>570</td>
<td>473</td>
<td>1211</td>
<td>20</td>
<td>1674</td>
<td>0.00%</td>
<td>47.6/min</td>
<td>11.2</td>
</tr>
<tr>
<td>TV Shows</td>
<td>90</td>
<td>1243</td>
<td>1116</td>
<td>2120</td>
<td>322</td>
<td>3445</td>
<td>0.00%</td>
<td>47.7/min</td>
<td>13.5</td>
</tr>
<tr>
<td>Movies</td>
<td>90</td>
<td>600</td>
<td>471</td>
<td>1214</td>
<td>19</td>
<td>5291</td>
<td>0.00%</td>
<td>48.3/min</td>
<td>9.6</td>
</tr>
<tr>
<td>Movie Rentals</td>
<td>90</td>
<td>521</td>
<td>474</td>
<td>1088</td>
<td>22</td>
<td>1655</td>
<td>0.00%</td>
<td>48.9/min</td>
<td>9.7</td>
</tr>
<tr>
<td>Free Apps</td>
<td>90</td>
<td>696</td>
<td>409</td>
<td>1113</td>
<td>20</td>
<td>15982</td>
<td>0.00%</td>
<td>49.2/min</td>
<td>10.1</td>
</tr>
<tr>
<td>Paid Apps</td>
<td>90</td>
<td>604</td>
<td>315</td>
<td>1089</td>
<td>23</td>
<td>12594</td>
<td>0.00%</td>
<td>49.6/min</td>
<td>10.1</td>
</tr>
<tr>
<td>Music Videos</td>
<td>90</td>
<td>497</td>
<td>375</td>
<td>1126</td>
<td>19</td>
<td>1537</td>
<td>0.00%</td>
<td>50.1/min</td>
<td>11.1</td>
</tr>
<tr>
<td>TOTAL</td>
<td>990</td>
<td>1172</td>
<td>818</td>
<td>2268</td>
<td>19</td>
<td>16510</td>
<td>0.00%</td>
<td>7.0/sec</td>
<td>140.7</td>
</tr>
<tr>
<td>Label</td>
<td># Samples</td>
<td>Average</td>
<td>Min</td>
<td>Max</td>
<td>Std. Dev.</td>
<td>Error %</td>
<td>Throughput</td>
<td>KB/sec</td>
<td>Avg. Bytes</td>
</tr>
<tr>
<td>----------------</td>
<td>-----------</td>
<td>---------</td>
<td>------</td>
<td>-------</td>
<td>-----------</td>
<td>----------</td>
<td>------------</td>
<td>--------</td>
<td>------------</td>
</tr>
<tr>
<td>Apple Home</td>
<td>2400</td>
<td>3357</td>
<td>413</td>
<td>16013</td>
<td>1931.38</td>
<td>0.00%</td>
<td>17.0/sec</td>
<td>1244.04</td>
<td>75083.5</td>
</tr>
<tr>
<td>iTunes</td>
<td>2400</td>
<td>2996</td>
<td>604</td>
<td>10933</td>
<td>1695.84</td>
<td>0.00%</td>
<td>19.1/sec</td>
<td>457.05</td>
<td>24451.0</td>
</tr>
<tr>
<td>Featured</td>
<td>2400</td>
<td>1878</td>
<td>315</td>
<td>9489</td>
<td>1397.97</td>
<td>0.00%</td>
<td>19.2/sec</td>
<td>256.21</td>
<td>13648.4</td>
</tr>
<tr>
<td>Songs</td>
<td>2400</td>
<td>1675</td>
<td>295</td>
<td>8861</td>
<td>1207.71</td>
<td>0.00%</td>
<td>19.2/sec</td>
<td>325.76</td>
<td>17338.4</td>
</tr>
<tr>
<td>Albums</td>
<td>2400</td>
<td>359</td>
<td>4</td>
<td>6248</td>
<td>590.50</td>
<td>0.00%</td>
<td>19.3/sec</td>
<td>261.36</td>
<td>13889.0</td>
</tr>
<tr>
<td>TV Shows</td>
<td>2400</td>
<td>1612</td>
<td>287</td>
<td>7565</td>
<td>1219.42</td>
<td>0.00%</td>
<td>19.3/sec</td>
<td>293.01</td>
<td>15569.9</td>
</tr>
<tr>
<td>Movies</td>
<td>2400</td>
<td>333</td>
<td>4</td>
<td>9763</td>
<td>608.44</td>
<td>0.00%</td>
<td>19.4/sec</td>
<td>227.19</td>
<td>11978.0</td>
</tr>
<tr>
<td>Movie Rentals</td>
<td>2400</td>
<td>392</td>
<td>4</td>
<td>4005</td>
<td>636.43</td>
<td>0.00%</td>
<td>19.5/sec</td>
<td>225.84</td>
<td>11845.0</td>
</tr>
<tr>
<td>Free Apps</td>
<td>2400</td>
<td>397</td>
<td>4</td>
<td>5208</td>
<td>665.48</td>
<td>0.00%</td>
<td>19.6/sec</td>
<td>241.07</td>
<td>12586.0</td>
</tr>
<tr>
<td>Paid Apps</td>
<td>2400</td>
<td>392</td>
<td>4</td>
<td>6760</td>
<td>671.94</td>
<td>0.00%</td>
<td>19.7/sec</td>
<td>243.28</td>
<td>12676.8</td>
</tr>
<tr>
<td>Music Videos</td>
<td>2400</td>
<td>371</td>
<td>4</td>
<td>4613</td>
<td>606.73</td>
<td>0.00%</td>
<td>19.7/sec</td>
<td>277.73</td>
<td>14419.5</td>
</tr>
<tr>
<td>TOTAL</td>
<td>26400</td>
<td>1251</td>
<td>4</td>
<td>16013</td>
<td>1557.90</td>
<td>0.00%</td>
<td>172.4/sec</td>
<td>3421.14</td>
<td>20316.8</td>
</tr>
</tbody>
</table>
Summary Report: railway_2
This test simulated up to 50 concurrent users for 6 minutes starting on 2015-01-19 23:48:45 UTC from US East (Virginia). A mean response time of 577 ms was observed with a standard deviation of 213 ms. The 95th percentile was 961 ms and the 50th percentile (median) was 562 ms. The Apdex was Excellent 1.0 [3000]. A mean throughput of 247.70 Mbps was observed with a peak of 372.48 Mbps. A total of 10.9 GB was transferred. A total of 34,258 requests were successfully simulated with no errors observed. The mean request rate was 6,861.00 rpm.
Test Report for: railway

These reports are a result of running a test named: railway on Fri, 01/23/2015 - 06:07.

This report is a result of spawning numerous servers in the cloud, executing a JMeter test-script using distributed JMeter architecture. All graphs represent KPIs (Key Performance Indicators) measured during the load.
Chapter 7: Helpful Tips
DROP TABLE IF EXISTS TEST;
CREATE TABLE TEST(ID INT PRIMARY KEY, NAME VARCHAR(255));
INSERT INTO TEST VALUES(1, 'Hello');
INSERT INTO TEST VALUES(2, 'World');
### JDBC Connection Configuration

**Name:** JDBC Connection Configuration

**Comments:**

#### Variable Name Bound to Pool

**Variable Name:** testPool

#### Connection Pool Configuration

- **Max Number of Connections:** 10
- **Pool Timeout:** 10000
- **Idle Cleanup Interval (ms):** 60000
- **Auto Commit:** True
- **Transaction Isolation:** DEFAULT

#### Connection Validation by Pool

- **Keep-Alive:** True
- **Max Connection age (ms):** 5000
- **Validation Query:** Select 1 from dual

#### Database Connection Configuration

- **Database URL:** jdbc:h2:tcp://localhost//tmp/test;MVCC=TRUE
- **JDBC Driver class:** org.h2.Driver
- **Username:** sa
- **Password:**
Name: HTML Response
Comments:

Successfull sample:
Response Code (eg 200): 200
Response Message (eg OK): OK
Latency (milliseconds): ${__Random(1,100)}
Response Time (milliseconds): ${__Random(100,1000)}
Simulate Response Time (sleep):

Request Data: Dummy Sampler used to simulate requests and responses without actual network activity. This helps debugging tests.

Response Data:
<html>
<head>
</head>
<body>
This is a test
</body>
</html>

<title>Welcome to Debug Sampler</title>