Metrics that Matter – Approaches to Managing High Performing Web Sites

Ben Rushlo
Director Keynote Professional Services
June, 22nd 2009
Agenda

- User Centric System Approach
- Performance Management Begins with Metrics
- Metrics That Matter
- Diagnostic Process
- Keys to Improving Performance
- Implementing A Total Site Quality Framework
Personal Background

- 9 years at Keynote – Keynote Consulting Practice
- 5 years at Keynote – Director of Keynote Consulting Practice
- Focus on mid/large enterprise sites
  - Wal-Mart
  - eBay
  - Honda
  - Ford
  - Schwab
- Background in capacity planning
- CIS/MIS degree
User Centric System Approach
Change Has Come

- Single data center → Cloud hosting/services
- HTML → ASP/JSP
- JS → AJAX
- Animated GIFs → Sites Completely in Flash
- Content Driven → Transaction Driven → Experience Driven
- US Market → Global Market
- Single domains → 20 domains per page
- Legacy systems → Outsourced web services
Your user has changed
  - Decreased tolerance, increased expectations
  - Utility/Always on
  - Integrated completely into our lives
    - When Larry King is using Twitter…
    - When outages are front page news…
“A **system** is a dynamic and complex whole, interacting as a **structured functional unit**”
Online Applications Are Complex Systems

- Content Delivery Network
- Application Code
- Third Party Web Services
- Front End Design
- Network/Servers/Infrastructure
- Tracking/Ad Tags
- Cloud Services
- ISPs
- Creative/Visual Content

Online Application User Experience
Online Applications Are Complex Systems

- JSP
- ASP
- DB Query
- Java Environment
- CSS Code
- Java Script Code
- Front End Design
- AJAX/XML
- Browser Threading
- Application Code
Online Applications Are Complex Systems
Online Applications Are Complex Systems

- While we have undergone rapid change in the area of web site design/technology/architecture has performance management changed with it?

- Or are we still living in a client server focused paradigm?

- Are we viewing the discrete and disconnected elements of the system and not the system?
  - CPU/Memory/IO etc.
  - Garbage collection rate/threads etc.
  - Locks/query time etc.
In any complex system, there is an overwhelming number of metrics (things to measure that describe elements of the system).

However, within any system there are key indicators of system health:
- Think of air speed, altitude
- Think of GDP or consumer confidence
- Think of blood pressure and weight
Top Order Metrics

- Top order metrics require a **top down** approach
- It is virtually impossible to combine low level metrics upwards to understand system health
  - Except for extreme cases (100% CPU, server down etc.)
  - Most performance management issues are not so simple
- Low level metrics are very useful once you have identified areas of focus/problem areas
Performance management must begin and end at the end users perspective

- The end user provides
  - A unifying approach to a very complex system
  - Key barometer of site/application success
  - A direct tie to business owner/goals and work of performance management team
Performance Management Beings with Metrics
Data Collection

- Beginning with the users perspective (unifying approach) how do we collect data?
  - Point in time?
  - Ongoing collection?
  - Data center or Internet?
  - Browser based?
  - Geographically distributed?
  - Connection speed?
  - How wide and how deep?
Point In Time Tools

- Point in Time Tools
  - User Feedback
  - Yslow
  - Google Page Speed
  - Firebug
  - HTTP Analyzer
  - HTTP Watch
  - KITE

- Good for rules based/best practice analysis and point in time data collection

- Free or almost free!
What a Difference a Couple Thousand Data Points Make

- Amazon Home Page
  - HTTP Analyzer Trace
  - 81 requests/responses
What a Difference a Couple Thousand Data Points Make

- **Amazon – Profile**
  - 15 slowest requests (Average and variability)
  - 2,000 data points in sample
Ongoing Measurement Approaches

- Passive technology “watches” network traffic
  - Benefits:
    - Can “see” all users (huge sample, actual visitors)
    - Allows for “measurements” of pages that are difficult to measure in any other way (like a purchase confirmation)
  - Challenges:
    - Security issues
    - Hybrid hosted sites and third party content (can’t see what is happening with browser and external sources)
    - Not good for availability (a key PM activity)
    - Highly variable sample
Ongoing Measurement Approaches

- Tagging technology uses JS to instrument areas on the page with timers
  - Benefits:
    - Real user data.
    - Large sample
    - End user perspective (can include client time)
  - Challenges:
    - Requires code changes (on each page)
    - Lacking in granularity
    - Management ongoing can be cumbersome and difficult
Active technology uses synthetic transactions to "simulate" users on the site

Benefits:
- Controlled and consistent environment (only variables originate from the site)
- Repeatable
- Large sample

Challenges:
- Not every path can be scripted
- Not every user configuration can be modeled
- Choosing the "right" path can be difficult
Inside or Outside?

- Where does the online **application live**?
  - No longer completely in the data center in most cases
  - Hybrid hosting, CDN, web services, third party content, third party tags etc.
  - Very incomplete view of performance/quality

- Where does the **user live**?
  - No users access the site from the data center
  - Performance management cannot be done effectively within a LAN environment
  - Impact of external latency **cannot be calculated**
Multiple Locations or Not?

<table>
<thead>
<tr>
<th></th>
<th>Vancouver Telus</th>
<th>Calgary Telus</th>
<th>Toronto Bell</th>
<th>Montreal Verizon</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home Page</td>
<td>2.00</td>
<td>2.41</td>
<td>3.48</td>
<td>3.99</td>
</tr>
<tr>
<td>Online</td>
<td>1.16</td>
<td>1.50</td>
<td>2.84</td>
<td>3.08</td>
</tr>
<tr>
<td>Online Edition</td>
<td>1.23</td>
<td>1.42</td>
<td>2.38</td>
<td>2.57</td>
</tr>
<tr>
<td>Get Started</td>
<td>7.31</td>
<td>8.80</td>
<td>18.98</td>
<td>20.91</td>
</tr>
<tr>
<td>Validation</td>
<td>0.56</td>
<td>0.56</td>
<td>1.09</td>
<td>1.15</td>
</tr>
</tbody>
</table>
Browser or Not?

<table>
<thead>
<tr>
<th></th>
<th>Time In Browser</th>
<th>Download Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home Page</td>
<td>1.36</td>
<td>1.41</td>
</tr>
<tr>
<td>Home</td>
<td>1.54</td>
<td>0.99</td>
</tr>
<tr>
<td>Photo - Video Gallery</td>
<td>1.70</td>
<td>0.46</td>
</tr>
<tr>
<td>Features</td>
<td>1.56</td>
<td>0.83</td>
</tr>
<tr>
<td>Specs</td>
<td>1.89</td>
<td>0.37</td>
</tr>
<tr>
<td>Dealer Results</td>
<td>1.89</td>
<td>1.67</td>
</tr>
</tbody>
</table>
Browser or Not?
Browser Or Not?

- The browser is “the” application engine
  - JS execution
  - Client side processing
  - Dynamic content
- It is almost impossible to emulate complexity of browser
  - Threading model
  - Blocking/Asynchronous characteristics
  - Dynamic JS and CSS engine
  - Flash/Silverlight/Flex load/dynamic paths and execution
  - Render related issues
Multiple Connection Speeds or Not?

- **Broadband**
  - 3.0Mbps → Above
  - DSL/Cable Home
  - Business

- **Midband**
  - Below 1.5Mbps
  - Entry level DSL

- **Narrowband**
  - 56Kbps
  - Dial-up
  - Consumer Satellite
On any site there are an extremely large number of pages that can be measured
- Can’t measure everything
- How do we choose?

User centric/business centric model
- What are the most common and most critical paths that the user takes throughout the site?
- What pages share similar architecture/design/dependencies?
- What pages/functions will wake up the CEO if they fail?

Even very large and complex sites can be measured in two to five key business paths typically.
Metrics that Matter
Context Is Everything

- Imagine if we all made up our own “goals” for cholesterol

- I consistently find performance people (CIO’s → performance analysts) who just make up what they think are appropriate goals/targets for key metrics
  - 99.999%?
  - 97%?

- A key component of any successful PM program is context, using appropriate goals/targets
  - Competitive data sets are a great way to get that context
  - Great point of connection with business owners/objectives
Search – Rental Cars

![Bar chart showing search times for various rental car companies.]

Source: Keynote Competitive Research – Rental Cars 2009
Total Transaction Availability – Rental Cars

- Dollar: 99.98%
- Enterprise: 99.93%
- Alamo: 99.76%
- Orbitz: 99.33%
- Hertz: 98.69%
- Median: 98.36%
- Travelocity: 98.03%
- Avis: 97.98%
- Expedia: 97.24%
- Thrifty: 96.53%
- Budget: 94.38%

Source: Keynote Competitive Research – Rental Cars 2009
Averages Are the Muddy Middle
Variability Is Very Important

![Render Time Statistical Summary]

- **Arithmetic Mean**
- **Geometric Mean**
- **Median**
- **85th Percentile**
- **95th Percentile**

**Seconds**

- Login
- Click Exchange
- Search - Orlando
- Submit
- Search - Cancun
Client Side Processing

- Client side processing is virtually unexamined in most performance management programs
  - Not tracked by most tools
  - Only beginning to be discussed as part of performance management
- Yet for many sites this is **the key contributor to poor performance**
To impact and improve user centric performance, focus on **9 core metrics**:

- Availability
- Outages
- Average Download Time - Geo Mean
- Time in Client Versus Time In Generation/Backend
- Variability - 85th and 95th percentiles
- Geographic Variability
- Hourly Variability (Load Handling)
- Third Party Quality
- Size/Element Count/Domains
Core PM Metrics

- Availability – 99.5% for multi-step transaction
- Outages – 1 hour per month
- Average Download Time - 1.5 -2.5s (broadband)
- Time in Client Versus Time In Generation/Backend – Less than 30% of page load
- Variability - 85th and 95th percentiles – No more than 1.5X the median
- Geographic Variability – No more than 2X (fastest versus slowest)
- Hourly Variability (Load Handling) – Less than 20% peak versus off peak
- Third Party Quality – Tags under 50MS each (limited variability, good availability)
- Size/Element Count/Domains – Depends! 😊
# Health Scorecard Example

<table>
<thead>
<tr>
<th>Page</th>
<th>Grade</th>
<th>Page Download (sec)</th>
<th>Generation/Network Time (sec)</th>
<th>Browser/Render Time (sec)</th>
<th>85th Pctile (sec)</th>
<th>95th Pctile (sec)</th>
<th>Geographic Uniformity</th>
<th>Load Handling</th>
<th>Peak Availability</th>
<th>Outage Hours</th>
<th>Size (k)</th>
<th>Elements</th>
<th>Domains</th>
<th>Network Connections</th>
<th>DIIS Lookups</th>
<th>JS Files</th>
<th>CSS Files</th>
<th>Missing Content</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Electronics - Sony Style</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Home Page</td>
<td>D+</td>
<td>6.2</td>
<td>3.4</td>
<td>2.8</td>
<td>6.7</td>
<td>8.5</td>
<td>10.9 %</td>
<td>3.3 %</td>
<td>100.0 %</td>
<td>0</td>
<td>2734.8</td>
<td>110.1</td>
<td>7.0</td>
<td>13.4</td>
<td>6.0</td>
<td>32.0</td>
<td>4.0</td>
<td>Yes</td>
</tr>
<tr>
<td>Search Results</td>
<td>C+</td>
<td>4.8</td>
<td>2.1</td>
<td>2.7</td>
<td>6.1</td>
<td>7.2</td>
<td>8.8 %</td>
<td>14.9 %</td>
<td>99.1 %</td>
<td>0</td>
<td>446.8</td>
<td>65.3</td>
<td>2.2</td>
<td>1.1</td>
<td>0.0</td>
<td>6.0</td>
<td>4.0</td>
<td>Yes</td>
</tr>
<tr>
<td>Category</td>
<td>B</td>
<td>4.7</td>
<td>1.4</td>
<td>3.3</td>
<td>6.2</td>
<td>7.3</td>
<td>9.2 %</td>
<td>5.9 %</td>
<td>99.7 %</td>
<td>0</td>
<td>46.0</td>
<td>13.1</td>
<td>4.1</td>
<td>2.1</td>
<td>0.0</td>
<td>3.0</td>
<td>1.0</td>
<td>Yes</td>
</tr>
<tr>
<td>Details</td>
<td>C-</td>
<td>5.8</td>
<td>2.5</td>
<td>3.1</td>
<td>6.3</td>
<td>7.9</td>
<td>21.9 %</td>
<td>11.1 %</td>
<td>99.9 %</td>
<td>0</td>
<td>284.6</td>
<td>59.1</td>
<td>8.0</td>
<td>6.2</td>
<td>1.0</td>
<td>26.3</td>
<td>2.0</td>
<td>Yes</td>
</tr>
<tr>
<td>Cart</td>
<td>C-</td>
<td>5.4</td>
<td>3.7</td>
<td>1.7</td>
<td>6.8</td>
<td>7.5</td>
<td>6.9 %</td>
<td>8.5 %</td>
<td>99.9 %</td>
<td>0</td>
<td>1231.8</td>
<td>65.7</td>
<td>4.9</td>
<td>3.0</td>
<td>1.0</td>
<td>24.5</td>
<td>4.0</td>
<td>No</td>
</tr>
</tbody>
</table>
## Health Scorecard Example

<table>
<thead>
<tr>
<th>Page</th>
<th>Grade</th>
<th>Page Download (sec)</th>
<th>Generation Network Time (sec)</th>
<th>Browser/Render Time (sec)</th>
<th>85th Pctile (sec)</th>
<th>95th Pctile (sec)</th>
<th>Geographic Uniformity</th>
<th>Load Handling</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Electronics - Sony Style</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Home Page</td>
<td>D+</td>
<td>6.2</td>
<td>3.4</td>
<td>2.8</td>
<td>6.7</td>
<td>8.5</td>
<td>10.9 %</td>
<td>3.3 %</td>
</tr>
<tr>
<td>Search Results</td>
<td>C+</td>
<td>4.8</td>
<td>2.1</td>
<td>2.7</td>
<td>6.1</td>
<td>7.2</td>
<td>8.8 %</td>
<td>14.9 %</td>
</tr>
<tr>
<td>Category</td>
<td>B</td>
<td>4.7</td>
<td>1.4</td>
<td>3.3</td>
<td>6.2</td>
<td>7.3</td>
<td>9.2 %</td>
<td>5.9 %</td>
</tr>
<tr>
<td>Details</td>
<td>C-</td>
<td>5.6</td>
<td>2.5</td>
<td>3.1</td>
<td>6.3</td>
<td>7.9</td>
<td>21.9 %</td>
<td>11.1 %</td>
</tr>
<tr>
<td>Cart</td>
<td>C-</td>
<td>5.4</td>
<td>3.7</td>
<td>1.7</td>
<td>6.8</td>
<td>7.5</td>
<td>6.9 %</td>
<td>8.5 %</td>
</tr>
</tbody>
</table>
## Health Scorecard Example

<table>
<thead>
<tr>
<th>Peak Availability</th>
<th>Outage Hours</th>
<th>Size (k)</th>
<th>Elements</th>
<th>Domains</th>
<th>Network Connections</th>
<th>DNS Lookups</th>
<th>JS Files</th>
<th>CSS Files</th>
<th>Missing Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>100.0%</td>
<td>0</td>
<td>2734.8</td>
<td>110.1</td>
<td>7.0</td>
<td>13.4</td>
<td>6.0</td>
<td>32.0</td>
<td>4.0</td>
<td>Yes</td>
</tr>
<tr>
<td>99.1%</td>
<td>0</td>
<td>448.8</td>
<td>65.3</td>
<td>2.2</td>
<td>1.1</td>
<td>0.0</td>
<td>6.0</td>
<td>4.0</td>
<td>Yes</td>
</tr>
<tr>
<td>99.7%</td>
<td>0</td>
<td>46.0</td>
<td>13.1</td>
<td>4.1</td>
<td>2.1</td>
<td>0.0</td>
<td>3.0</td>
<td>1.0</td>
<td>Yes</td>
</tr>
<tr>
<td>99.9%</td>
<td>0</td>
<td>284.6</td>
<td>59.1</td>
<td>8.0</td>
<td>6.2</td>
<td>1.0</td>
<td>25.3</td>
<td>2.0</td>
<td>Yes</td>
</tr>
<tr>
<td>99.9%</td>
<td>0</td>
<td>1231.8</td>
<td>65.7</td>
<td>4.9</td>
<td>3.0</td>
<td>1.0</td>
<td>24.5</td>
<td>4.0</td>
<td>No</td>
</tr>
</tbody>
</table>
Asynchronous/Blocking
Page Usability Metric – Pre Render Delay Versus Post Render
Diagnostic Process
Diagnostic Process

- Being with standards and good “change based” alerting
  - Are the metrics out of threshold (based on context)?
  - Or have they changed from where they have been?
Diagnose performance over time

- Yesterday
- Last week
- Last month and Month-To-Date
Diagnostic Process

- Do you see consistent performance problems over time?
  - If so, the page needs to be profiled to determine
    - Content (CDN or web server quality)
    - Application
    - Front-end design (e.g. Third party calls)
  - If so, has something changed?
    - New content? New requests?
  - Is there a time of day/hour/location pattern?
    - Capacity
    - Edge cache
    - ISP issue
Where Performance Problems Lie

- Front End Design: 40%
- Application: 35%
- Tags/Third Party: 10%
- Infrastructure/Hosting: 5%
- Content Delivery Network: 5%
- Craziness: 5%
Errors

- Categorize by type
  - Network
  - Server
  - Application

- Tool should have actual (not simulated) screen capture

- Tool should use a browser
  - Many errors (most) are custom application or malformed pages
  - Browser is much better at catching errors that “HTTP Request/Response Tool” because it is more sensitive to dynamic, real world issues
Keys to Improving Performance
Silo versus user “flow” based approach
- JS and CSS have no strategy for minimizing separate and isolated files
- Need to take into account “flow” of user throughout site

Combination of JS and CSS is key
- Reduces roundtrips
- Lessen impact of single threading on JS
- Combination (or packing) of files more critical than minification

Key: Combine JS/CSS. Think Paths not Pages.
JS Placement

Javascript files load one file at a time

None of these images were downloaded to the browser until 2.4 seconds into a 2.8 second page load

Key: Combine, Move Down External JS
Roundtrips

- Myth in front end design that page size/asset site is still significant
  - Reducing cookie “overhead”
  - GZip
  - Minification
  - Image optimization
  - Etc
- These are best practices but they cannot compare to the criticality of round trips
  - Network speed much more critical than bandwidth (above 3.0Mbps)

**Key:** Reduce roundtrips. CSS Sprite for static content
Third Party Tag Placement and Quality

Site Launched

Key: Place Third Party Content in Footer and Track Quality
Key: Identify and Reduce Client Side Processing
Cache Management

Key: Configure cache settings – Far Future Etc.
Slow and Variable Application Calls
Slow and Variable Application Calls

Key: Profile application call variability
Other!!!!

- Capacity issues
- Persistent connections
- Incorrectly sized content
- Network retrans
- Errors of every type
Implementing Your Total Site Quality Framework
Implementing Total Site Quality Framework

- Begin with the user centric approach
- Apply competitive context and business goals to create appropriate targets
- Collect 9 core PM metrics
  - Use an ongoing, external, geographically distributed, browser based solution to collect data
  - Path based, key pages/function approach
- Apply collected data against targets
  - Flag change/target exceeded
- Perform diagnostic process
www.fastwebrate.com

Submit your fast or slow site by July 15th

THE FASTWEB RACE
From fast to fast

It’s a race out there –
faster sites, cooler apps, better interactivity.

The FastWeb Race is all about making the Web faster. It’s a hands-on contest for Web developers, QA professionals, Web Operations experts, Performance Analysts and others – in short anyone who builds, tests, optimizes or manages Web sites and apps.

Submit your real-life sites/transactions. We’re accepting submissions through July 15, 2009.

THE JUDGES

Steve Souders
Google

Ben Rushio
Keynote Systems

Bob Bufhone
RockStarApps

Dawn Parzych
F5 Networks

CATEGORIES

A Fastest in the world
Who should submit?
- You have a faster than (or close to) light speed site and/or have achieved an order of magnitude

B Eager to become the fastest
Who should submit?
- You have tried (almost) everything to make your site faster, and you just don’t get the performance.

SUBMIT YOUR ENTRY!

Step-1 | Step 2

First Name*

Last Name*

Email*

Company*
How to Reach Me

ben.rushlo@keynote.com

(623) 547-7068

http://www.linkedin.com/in/benrushlo