User Experience, not Metrics;
Performance Engineering Methodology

First Presented for:

Northern Virginia Rational User's Group

Reston, VA 2002

Scott Barber

Chief Technology Officer

PerfTestPlus, Inc.

© 2006 PerfTestPlus, Inc. All rights reserved.

Agenda

What are Performance Metrics?

What are User Experience Measurements?

Modelling Real Users and User Communities

Collecting Meaningful Times and Statistics

Interfacing with Developers and Architects

Iterative Tuning

Reporting to Stakeholders

Summary / Questions

© 2006 PerfTestPlus, Inc. All rights reserved.

What are Performance Metrics?

System Component Statistics

Often misleading

Generally impressive sounding

Generally NOT useful in making “Go Live” decisions

Often Misused and Misunderstood

Not an indication of interaction with other system/application tiers

No direct correlation to user experience

Not intuitively valuable

© 2006 PerfTestPlus, Inc. All rights reserved.

What are Performance Metrics?

Common Metrics Include

- Throughput
  - Number of bytes a Web Server can serve in a unit of time
  - Useful in comparing Web Server Hardware and Software
- Hits per Second
  - How many requests can a Web Server process in a second
  - Useful in comparing Web Server Hardware and Software
- Bandwidth
  - Same as Throughput – usually referring to networks or internet pipelines

© 2006 PerfTestPlus, Inc. All rights reserved.

What are Performance Metrics?

Common Misconceptions

- Scenario: Throughput is 200kbps. Average page size is 50kb.
  - Assumption: Site can process 4 page requests per second (or 14,400 per hour) with sub-second response time.
  - Truth: Throughput does not take into account time to process requests, query the database, execute server-side scripts, etc. Each page request may still take several (or more) seconds to process.
- Scenario: Web Server handles up to 100 hits per second.
  - Assumption: Web Server can handle 100 page requests per second.
  - Truth: A normal page request contains between 6 and 30 individual requests (hits). Hits per second only refers to the total number of requests that can be received into the server in a second, not how many get processed and returned (measures Gets, not Posts).

© 2006 PerfTestPlus, Inc. All rights reserved.

What are User Experience Measurements?

Total System Response Time from the perspective of an end user

- Directly representative what real users experience
- Expressed in time from user action to completed system response
- Generally useful in making “Go Live” decisions
- Do provide indications of interaction with other system/application tiers
- Intuitively valuable

© 2006 PerfTestPlus, Inc. All rights reserved.

What are User Experience Measurements?

Percent of pages/objects that load properly

- Directly representative what real users experience
- Expressed as pass/fail by page and/or object
- Generally useful in making “Go Live” decisions
- Often give an indication of interaction with other application tiers
- Intuitively valuable

© 2006 PerfTestPlus, Inc. All rights reserved.

What are User Experience Measurements?

User Experience Measurements answer the questions:

- Does my site meet the ‘8 second’ rule?
- How many users can use my site before it gets too slow?
- How slow will my site be if my users on on a 28.8 kbps Modem?
- Is my site stable under load?
- Does my site function as expected under load?
- Does my site return the expected content under load?

© 2006 PerfTestPlus, Inc. All rights reserved.

Modelling Real Users and User Communities

To predict system response to a real user, measure the response to a real user.

Tests that do not model real users can provide dramatically inaccurate results.

Key areas to model as realistically as possible are:

- How long it takes users to accomplish tasks
- Distribution of types of user activities
- Amount and variation of data (input)
- Distribution of types of users
- Caching?

© 2006 PerfTestPlus, Inc. All rights reserved.

Modelling Real Users and User Communities

Users Think

- Users think and type at different rates.
- When possible time actual people using the tested application or a similar one.
- Refer to published statistics.
- Time how long it takes YOU to use the application.
- Use common sense.
- Remember: Sometimes, real users go to get a drink or answer the phone while using an application.

© 2006 PerfTestPlus, Inc. All rights reserved.

Users Think – Normal Distribution Function

```
int func normdist(min, max, stdev) /* specifies input values for normdist function */

int min, max, stdev; /* min: Minimum value; max: Maximum value; stdev: degree of deviation */

{
  int range, iterate, result; /* declare range, iterate and result as integers – VuC does not support floating  point math */

  range = max - min;  /* range of possible values is the difference between the max and min values */

  iterate = range / stdev; /* this number of iterations ensures the proper shape of  the resulting curve */

  result = 0 ; /* integers are not automatically initialized to 0 upon declaration */

  stdev += 1 ; /* compensation for integer vs. floating point math */

   for (c = iterate; c != 0; c--)                              /* loop through iterations */
      result += (uniform (1, 100) * stdev) / 100; /* calculate and tally result */
  return result + min;                                         /* send final result back */
}
```

© 2006 PerfTestPlus, Inc. All rights reserved.

Modelling Real Users and User Communities

Users Think – Example

As Recorded

```
http_header_recv ["RDN_on_~233"]  304;  /* Not Modified */

http_nrecv ["RDN_on_~234"]  100 %% ;    /* 238 bytes - From Cache */
stop_time ["Home Page"];

start_time ["Page1"];
set Think_avg = 12342;

/* Keep-Alive request over connection www_noblestar_com */
http_request ["RDN_on_~235"]
```

Modified

```
http_header_recv ["RDN_on_~233"]  304;  /* Not Modified */

http_nrecv ["RDN_on_~234"]  100 %% ;    /* 238 bytes - From Cache */
stop_time ["Home Page"];

delay(uniform(6000,18000));  /* added to replace Think_avg below  */

start_time ["Page1"];
/* set Think_avg = 12342;  - replaced by delay above*/

/* Keep-Alive request over connection www_noblestar_com */
http_request ["RDN_on_~235"]
```

© 2006 PerfTestPlus, Inc. All rights reserved.

Modelling Real Users and User Communities

How often do users do this?

- A community of users, over time, will establish a pattern of activities.
- Research historical data.
- Use market research trends and statistics.
- If there are 3 ways to do something, real users will use all of them sometimes.
- Use common sense.
- Remember: Not every user does everything.

© 2006 PerfTestPlus, Inc. All rights reserved.

Modelling Real Users and User Communities

How often do users do this? - Example

As Recorded

```
http_header_recv ["RDN_on_~233"]  304;  /* Not Modified */

http_nrecv ["RDN_on_~234"]  100 %% ;    /* 238 bytes - From Cache */
stop_time ["Home Page"];

start_time ["Page1"];
set Think_avg = 12342;

/* Keep-Alive request over connection www_noblestar_com */
http_request ["RDN_on_~235"]
…
http_header_recv ["RDN_on_~242"]  200;  /* OK */

http_nrecv ["RDN_on_~243"]  100 %% ;    /* 9997 bytes */
stop_time ["Page1"];
```

© 2006 PerfTestPlus, Inc. All rights reserved.

Modelling Real Users and User Communities

How often do users do this? - Example

Header

```
#include <VU.h>
#include <Noblestar.h>

int  percent;              /* variable added to determine what % of the time a  particular  section
                              of the script will execute */
int  howManyTimes;  /* variable added to determine how  many times repeatable sections will
                              be executed */
{
percent = uniform(1,10);  /*chooses a random number compare against for  execution
                              percentage*/
howManyTimes = uniform(1,2); /*chooses how many entries will be entered in a multiple entry
                              block*/
```

© 2006 PerfTestPlus, Inc. All rights reserved.

Modelling Real Users and User Communities

How often do users do this? - Example

Modified for percent of users performing task

```
http_header_recv ["RDN_on_~233"]  304;  /* Not Modified */

http_nrecv ["RDN_on_~234"]  100 %% ;    /* 238 bytes - From Cache */
stop_time ["Home Page"];

if (percent<=6){

  delay(uniform(6000,18000));  /* added to replace Think_avg below  */

  start_time ["Page1"];
  /* set Think_avg = 12342;  - replaced by delay above*/

  /* Keep-Alive request over connection www_noblestar_com */
  http_request ["RDN_on_~235"]
  …
  http_header_recv ["RDN_on_~242"]  200;  /* OK */

  http_nrecv ["RDN_on_~243"]  100 %% ;    /* 9997 bytes */
  stop_time ["Page1"];
}
```

© 2006 PerfTestPlus, Inc. All rights reserved.

Modelling Real Users and User Communities

How often do users do this? - Example

Modified for number of times task performed

```
http_header_recv ["RDN_on_~233"]  304;  /* Not Modified */

http_nrecv ["RDN_on_~234"]  100 %% ;   /* 238 bytes - From Cache */
stop_time ["Home Page"];

for(i=0;i<howManyTimes;i++){

  delay(uniform(6000,18000));  /* added to replace Think_avg below  */

  start_time ["Page1"];
  /* set Think_avg = 12342;  - replaced by delay above*/

  /* Keep-Alive request over connection www_noblestar_com */
  http_request ["RDN_on_~235"]
  …
  http_header_recv ["RDN_on_~242"]  200;  /* OK */

  http_nrecv ["RDN_on_~243"]  100 %% ;   /* 9997 bytes */
  stop_time ["Page1"];
}
```

© 2006 PerfTestPlus, Inc. All rights reserved.

Modelling Real Users and User Communities

What do users type?

- Volume and type of data entered will vary greatly from one user to another.
- Users do not always enter data correctly – remember to simulate incorrect entries.
- Research historical data.
- Use common sense.
- Remember: Sometimes users skip fields.

© 2006 PerfTestPlus, Inc. All rights reserved.

Modelling Real Users and User Communities

What kind of user are you?

- Many applications have several levels of users; Administrators, Users, Guests, etc.
- Access level may have a significant effect on system performance.
- Determine how many of which types of users are accessing the application.
- Research historical data.
- Use common sense.
- Remember: Administration may not happen often, but when it does, it is a performance hog.

© 2006 PerfTestPlus, Inc. All rights reserved.

Modelling Real Users and User Communities

Caching?

- Find out if the application saves data on the client and/or server side. Account for the answer.
- If there is caching:
    - What is cached?
    - Where is it cached?
    - How long does it live?
- Automated tools are not always good at simulating caching.
- Remember: Client side is faster than Server side is faster than none.

© 2006 PerfTestPlus, Inc. All rights reserved.

Modelling Real Users and User Communities

No two users are exactly alike, model them that way.

Think about communities, not individuals.

Don't over-model.

Remember:

- Sometimes, real users go to the kitchen to get a soda while using your application.
- Real users have real cats that jump, unexpectedly, on real keyboards.
- In general,

Real users are RANDOM.

© 2006 PerfTestPlus, Inc. All rights reserved.

Collecting Meaningful Times and Statistics

Page load times (end-user perspective)
- Sterile vs. Production environment
- Various potential connection rates
- Various potential user-loads

Object load success rate

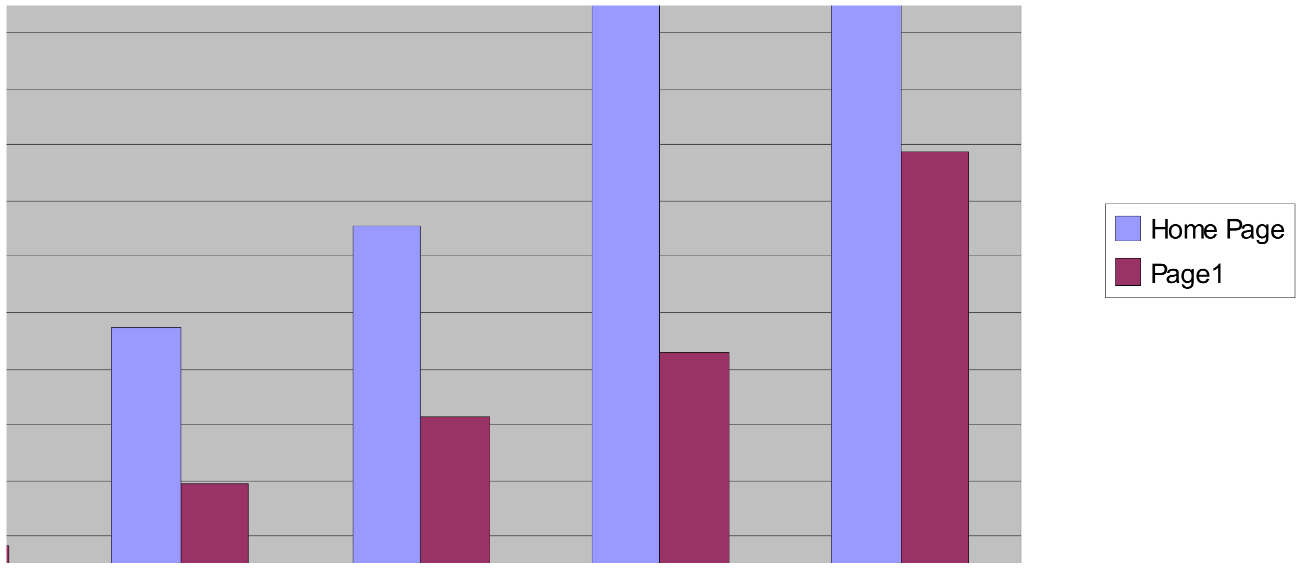
Page response success rate

Page load statistics
- Average by page
- Standard Deviation by page
- 95th Percentile by page

© 2006 PerfTestPlus, Inc. All rights reserved.

Collecting Meaningful Times and Statistics

Page Load Times - Example



CmdID	NUM	MEAN	STD DEV	MIN	50th	70th	80th	90th	95th	MAX
Home Page	99	4.53	1.47	3.33	4.08	4.57	4.87	5.77	7.99	11.05
Page1	100	2.94	0.85	2.26	2.59	2.91	3.48	4.08	4.65	6.44

© 2006 PerfTestPlus, Inc. All rights reserved.

Collecting Meaningful Times and Statistics

Page Load Times - Example

© 2006 PerfTestPlus, Inc. All rights reserved.

Collecting Meaningful Times and Statistics

Page Load Statistics

- Average time by page
  - Specific page across all times that page is requested during the execution of model (test run/Suite).
  - Good for generalizations, and/or if standard deviation is very small.
- Standard Deviation of page load times by page
  - Specific page during the execution of model (test run/suite).
  - Good for showing level of variance. Large Standard Deviations often mean more research/tuning is required.
- 95th Percentile by page
  - Specific page across all times that page is requested during the execution of model (test run/Suite).
  - Good for user expectation setting / determining requirement compliance.

© 2006 PerfTestPlus, Inc. All rights reserved.

Interfacing with Developers and Architects

Collaborative Testing/Tuning

- Involve developers and architects from the beginning
- Encourage them to assist in test designs
- Create tests to exercise areas of concern
- Share raw data and interpret data as a team
- Be willing to execute seemingly nonsensical tests to assist, but in the end only report user experience results from user community model test

Relate user experience measurements to specific architectural components.

Understand THEIR language.

Ask questions.

Be helpful, not confrontational.

© 2006 PerfTestPlus, Inc. All rights reserved.

Interfacing with Developers and Architects

For Developers/Architects to maximize user-experience, they need to know:

- What tier is the bottleneck on (web server, app server, db server, file server, network infrastructure, etc.)?
- What functions are slow (download graphics, db search, db write, authentication, field validation, etc.)
- The patterns of slow components.

To work collaboratively with Developers/ Architects testers must be prepared to:

- Customize tests to focus on slow components
- Re-run tests to validate changes
- Prepare comparisons between successive test-runs

© 2006 PerfTestPlus, Inc. All rights reserved.

Iterative Tuning

Performance Engineer every stable build

- Saves time and money in the long run
- Catches architectural flaws early
- Catches code flaws early
- Increases confidence on go live day

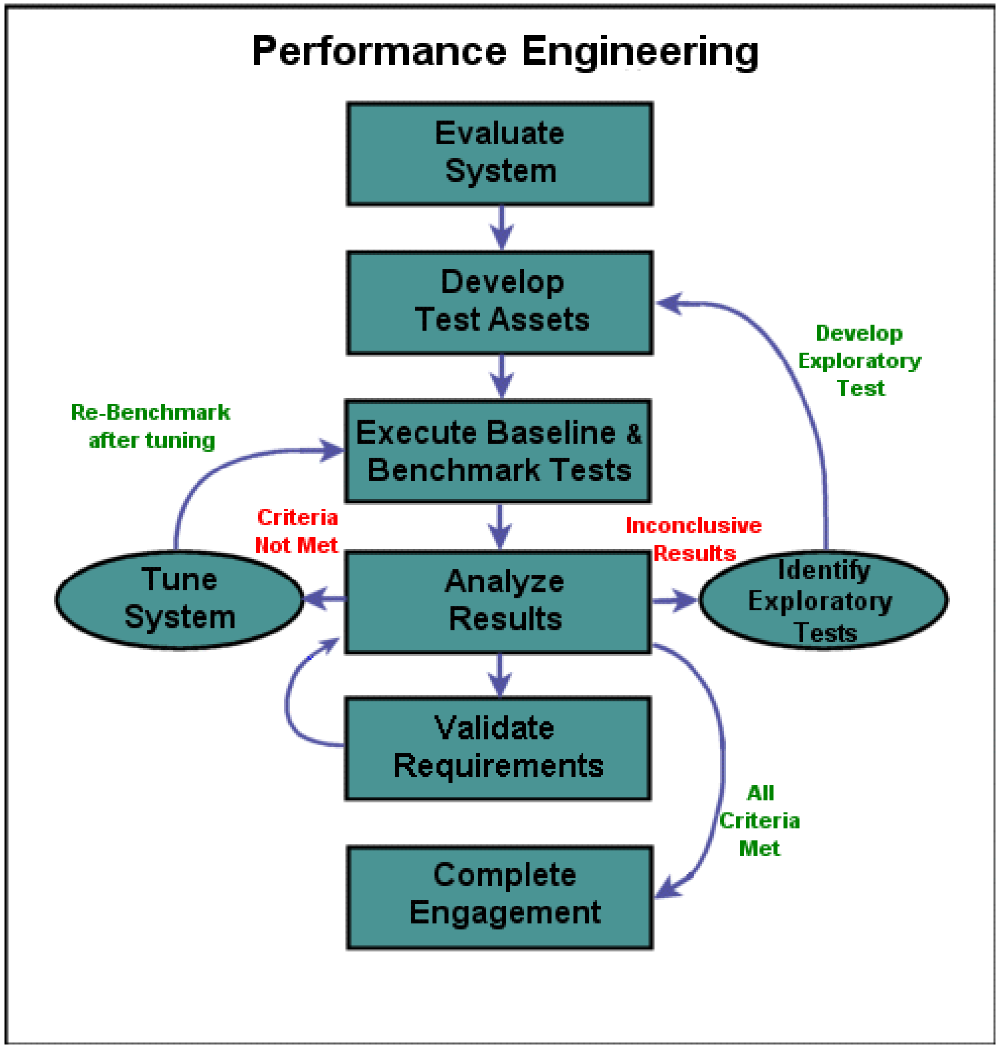
Re-Useable scripts

- Subsequent iterations take less time to build
- Flaws can be corrected using Trial and Error if necessary

Final Validation is easy if Performance Engineering was accomplished throughout development.

© 2006 PerfTestPlus, Inc. All rights reserved.

Iterative Tuning



© 2006 PerfTestPlus, Inc. All rights reserved.

Reporting to Stakeholders

Reports need to be meaningful and useful to both technical and management stakeholders.

Stakeholders want to see:

- Total number of concurrent users (while achieving goals).
- Total number of users per month (while achieving goals).
- Degradation points.
- Speed by page.
- Requirement compliance.
- Verification of stability over time.

© 2006 PerfTestPlus, Inc. All rights reserved.

Reporting to Stakeholders

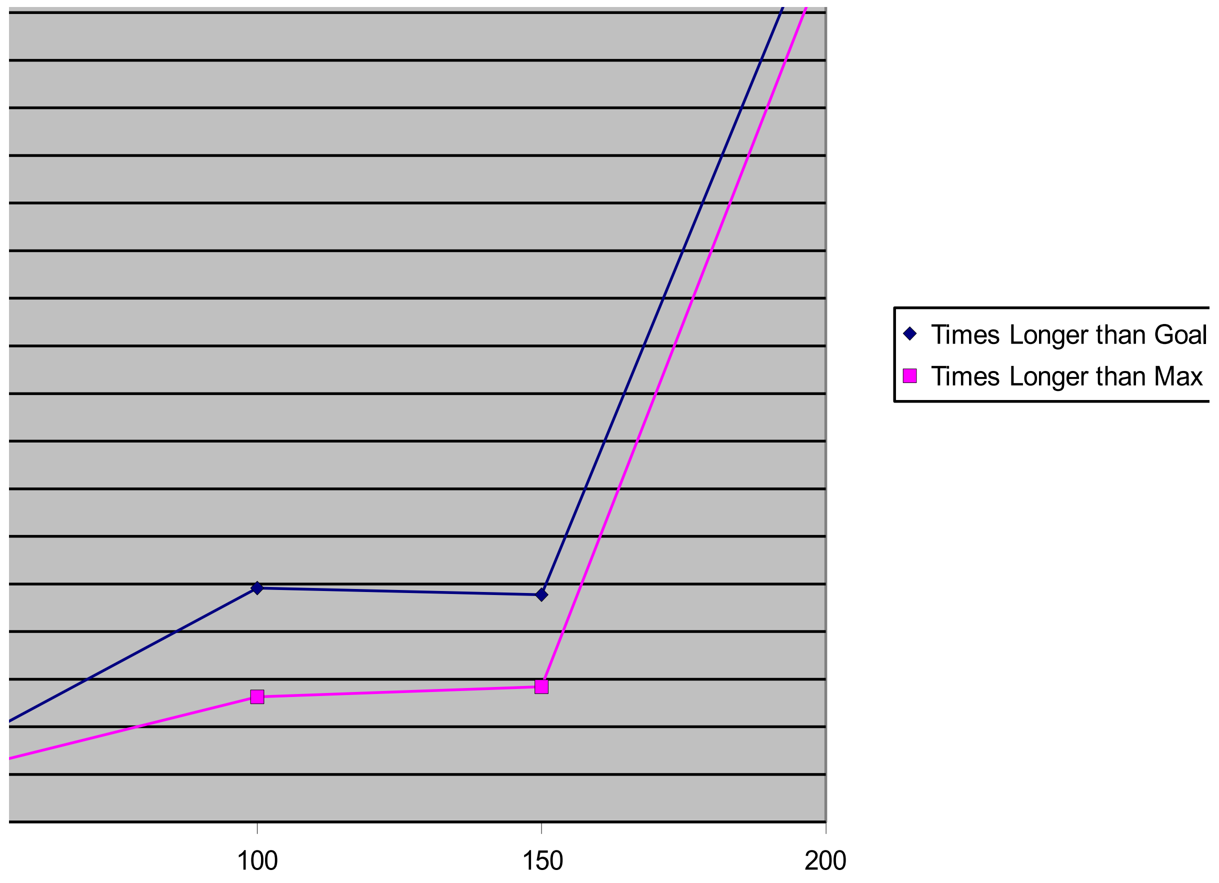
Page Load by Concurrent Users - Example

		Baseline			250 Users			500 Users	
Timer Name	Avg	95th	Max	Avg	95th	Max	Avg	95th	Max
ec_Main_Page	3.38	**10.46**	18.09	2.98	**6.41**	8.22	2.78	**6.33**	8.33
ec_logon_help	0.94	**0.98**	0.98	0.49	**0.56**	0.59	0.49	**0.55**	0.55
ec_login	3.10	**5.35**	7.92	2.43	**6.66**	11.84	2.47	**6.75**	17.03
quick_learns	3.20	**6.66**	6.67	1.44	**5.91**	10.98	1.47	**5.92**	11.02
view_quick_learn	6.45	**15.66**	17.11	1.06	**5.53**	10.72	0.96	**3.89**	10.61
view_faq_window	2.38	**2.45**	2.45	1.40	**1.47**	1.52	1.40	**1.53**	1.66
view_faq	0.65	**0.67**	0.67	0.51	**0.60**	0.63	0.51	**0.58**	0.69
view_ec_status	3.58	**8.08**	12.55	1.80	**1.73**	6.66	1.64	**1.80**	1.86

© 2006 PerfTestPlus, Inc. All rights reserved.

Reporting to Stakeholders

Performance Degradation - Example



© 2006 PerfTestPlus, Inc. All rights reserved.

Reporting to Stakeholders

Goals Compliance by User-Load - Example

© 2006 PerfTestPlus, Inc. All rights reserved.

Reporting to Stakeholders

Goals Compliance Summary - Example

Summary Comparison

Statistic/	LAN					128 kbs	56.6 kbs	28.8 kbs
Concurrent Users	1	50	100	150	200	100	100	100
Times Recorded	177	175	175	176	176	176	169	169
Times > Goal	31	15	43	43	170	98	122	136
% Times > Goal	17.5%	8.6%	24.6%	24.4%	96.6%	55.7%	72.2%	80.5%
Times > Max	4	10	23	24	160	35	106	123
% Times > Max	2.3%	5.7%	13.1%	13.6%	90.9%	19.9%	62.7%	72.8%
Typical Average Time By page	1.23	1.44	1.70	1.61	30.63	6.49	6.15	16.10
Typical 95th Percentile Time By page	3.42	2.98	4.67	4.06	53.51	7.68	8.74	17.72

© 2006 PerfTestPlus, Inc. All rights reserved.

Summary / Questions

Performance Metrics alone are evil.

User Experience Measurements are understandable and valuable on their own.

Modeling Real Users and User Communities yields real results.

Response times & success rates have meaning.

Developers and Architects are part of your team.

Tune as you go to save time and money.

Reports are for Managers AND Techies.

Questions.

© 2006 PerfTestPlus, Inc. All rights reserved.

Contact Info

Scott Barber

Chief Technology Officer

PerfTestPlus, Inc

E-mail: *sbarber@perftestplus.com*

Web Site: *www.PerfTestPlus.com*

© 2006 PerfTestPlus, Inc. All rights reserved.