

Written by
Theodore Bullock

With Assistance from
Eric Thiessen
Sheridan Thirsk
Andrew Hately

httperf

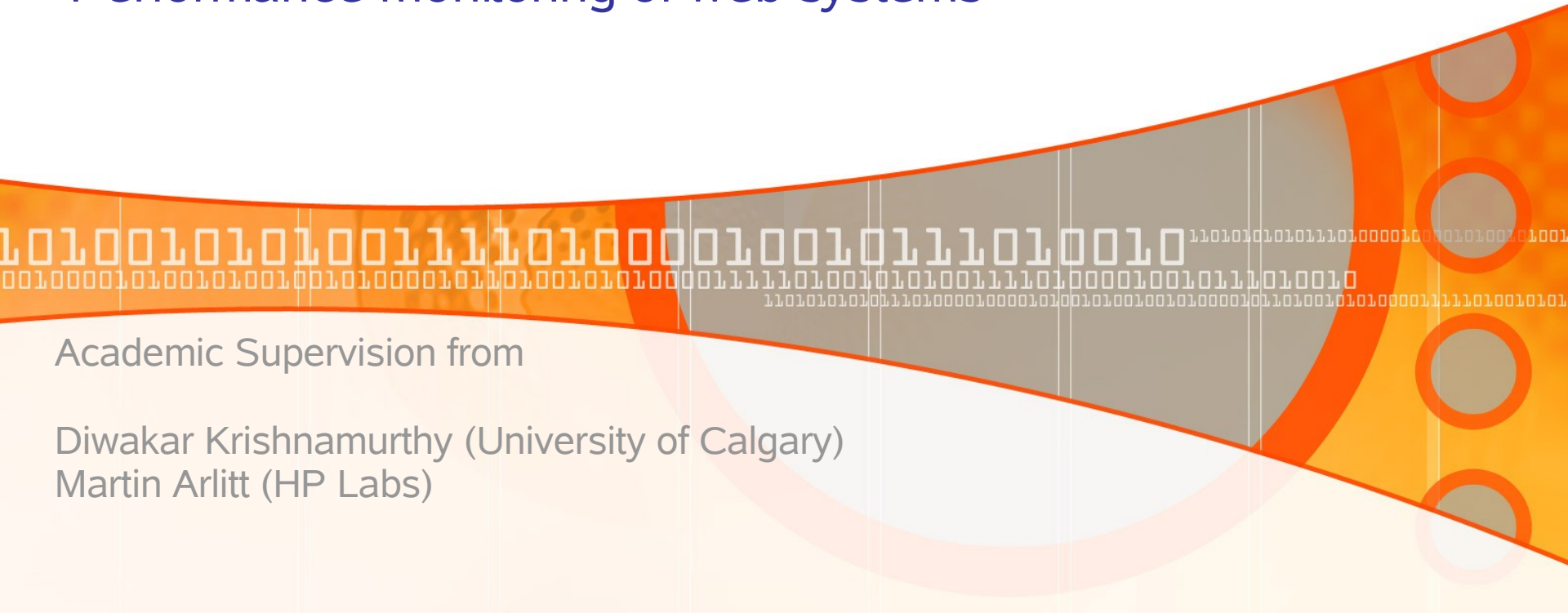
web workload generator

Performance monitoring of web systems

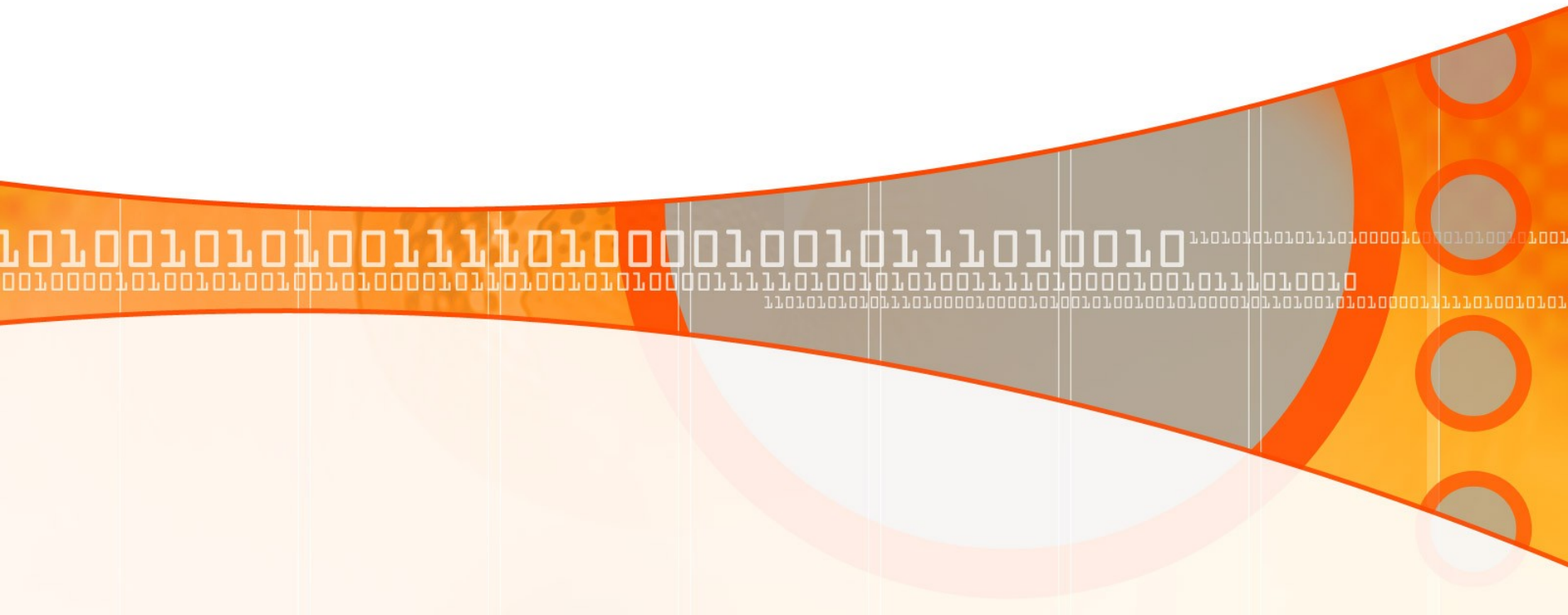
Academic Supervision from

Diwakar Krishnamurthy (University of Calgary)

Martin Arlitt (HP Labs)



***You Cannot Manage
That Which
You Cannot Measure***



What is httperf?

- **httplib** is an open source software tool for measuring web server performance
- Provides a flexible facility for generating various types of HTTP workloads and measuring server performance
- Available under the GNU General Public License (GPL)

Why httperf?

Three distinguishing characteristics:

1. **Robust:** Tuning options for Micro and Macro level adjustments to benchmarks
2. **Support:** Speaks HTTP/1.0, HTTP/1.1 and SSL
3. **Extensibility:** Straightforward to add new workload generators and performance measurements

Community & Involvement

- httpperf is currently maintained by Martin Arlitt of HP Labs
- Maintained and supported through mailing list <http://www.hpl.hp.com/research/linux/httpperf/mail.php>
- Yahoo! has shown interest in using httpperf to load test their web servers
- New releases of httpperf should provide the community, HP and Yahoo! with a stable and maintainable tool

History 0.8

httperf 0.8

- Released October 2000
- Developed by HP Labs
- Some contributions on the project mailing list over past 7 years but no new releases from HP

History 0.8.1

httperf 0.8.1

- Released April 2007
- Developed by student Software Engineering team at the University of Calgary
- Fixes known issues identified over past 6 years

History cont.

httperf 0.9.0

- Released April 2007 (Same time as 0.8.1)
- Developed by student Software Engineering team at the University of Calgary
- Provides identical functionality to 0.8.1
- Refactored build system to allow cross platform compilation:
 - Linux, FreeBSD 6.0, OpenBSD 4.0
 - Mac OS X, Solaris 8, HP-UX 11i
 - Others untested

Basic Examples

```
httpperf --hog -server=www
```

Creates one connection to host www and requests root document <http://www/>

Basic Examples cont.

```
httperf --hog --server=www --num-conns=100  
--rate=10 --timeout=5
```

As before, except that 100 connections are created at a fixed rate of 10 per second

Basic Examples cont.

```
httperf --hog --server=www --wsess=10,5,2  
--rate=1 --timeout=5
```

Generates a total of 10 sessions at a rate of 1 session per second. Each session consists of 5 calls that are spaced out by 2 seconds.

Workload Generators

Do not intermix!

Default

- Fixed number of HTTP GET requests

--wlog

- Iterates over a list of URIs (eg. Replay server log)

--wsess

- Measures sessions (collections of burst requests)

--wsesslog

- Replays sessions out of session description file

--wset

- Iterates over a set of URIs at a given rate

Output Explained!

101001010100111101000010010111010010 11010101010111010000100010100101001
001000010100101001001010000101101001010100111101000010010111010010
110101010101110100001000010100101001001010000101101001010100001111010010101

A decorative graphic at the bottom of the slide. It features a horizontal band of binary code (0s and 1s) in white and orange. The band is set against a background of orange and grey curved shapes, including a large grey semi-circle on the right and several smaller orange circles. The overall design is modern and tech-oriented.

Output Explained

httperf reports performance metrics for all experiments

- Total
- Connection
- Request
- Reply
- CPU
- Errors
- Session (only for wsess and wsesslog)

Output Explained: Connection

Connection rate: 49.0 conn/s (20.4
ms/conn, <=94 concurrent connections)

Connection time [ms]: min 1.6 avg 1030.3
max 3105.8 median 2003.5 stddev 1012.2

Line 1:

- Connection initiation rate/period and the max number of concurrent connections

Line 2:

- Lifetime statistics for successful connections

Output Explained: Reply

```
Reply rate [replies/s]: min 34.6 avg 190.7  
max 200.0 stddev 37.0 (20 samples)
```

```
Reply time [ms]: response 7.1 transfer 0.5
```

Line 1:

- The minimum, mean, max, standard deviation of the reply rate and number of samples used in calculation

Line 2:

- The average time for the server to respond to a request and the average time it took to fully receive the reply

Output Explained: Reply

Reply size [B]: header 234.0 content
95050.0 footer 0.0 (total 95284.0)

Reply status: 1xx=0 2xx=19222 3xx=0 4xx=0
5xx=0

Line 1:

- Average length of reply headers, content, and footers

Line 2:

- A histogram of received status codes

Output Explained: Misc

```
CPU time [s]: user 17.43 system 81.37  
              (user 17.1% system 79.8% total 96.9%)  
Net I/O: 17556.9 KB/s (143.8*10^6 bps)
```

Line 1:

- Summarizes the CPU statistics over the course of the experiment.
- Note that is the total CPU utilization is not close to 100%, then the results cannot be trusted

Line 2:

- Network I/O used to transfer and receive all data throughout the experiment.

Output Explained: Errors

```
Errors: total 195 client-timo 195 socket-  
timo 0 connrefused 0 connreset 0
```

```
Errors: fd-unavail 0 addrunavail 0 ftab-  
full 0 other 0
```

Simple histogram of the number and category of various errors encountered during the experiment. For details refer to quickstart guide or man page.

- Increase the `--timeout` and `--think-timeout` values to bring client-timo errors to zero.
- Decrease the `--timeout` and `--think-timeout` values to bring fd-unavail errors to zero.

Output Explained: Sessions

```
Session rate [sess/s]: min 7.80 avg 47.12  
max 50.01 stddev 9.48 (4805/5000)  
Session: avg 1.00 connections/session
```

Line 1:

- Vital statistics of all sessions initiated during execution, also the number of successful sessions

Line 2:

- The average number of connections per session (usually 1 with persistent connections)

Output Explained: Sessions

```
Session lifetime [s]: 1.0
```

```
Session failtime [s]: 1.0
```

```
Session length histogram: 195 0 2402 0 0 0  
2403
```

Line 1:

- Average time to complete session successfully

Line 2:

- Average time before an unsuccessful session has failed

Line 3:

- Number of sessions with 0, 1, 2, etc replies during the session

Future Plans!

A decorative graphic at the bottom of the slide. It features a horizontal band with a gradient from orange to yellow. Overlaid on this band are several elements: a series of binary digits (0s and 1s) in white and orange, a large, semi-transparent orange circle, and a vertical stack of four smaller orange circles on the right side. The background of the slide is white with a subtle grid of vertical lines.

101001010100111101000010010111010010110101010111010000100010100101001
001000010100101001001010000101101001010100111101000010010111010010
11010101011101000010000101001001001010000101101001010100001111010010101

Future Plans

httperf 1.0 – Enhancement release of selected functionality from the mailing list

- Support for Loading Multiple Servers
- IPv6 Support
- Providing structured data output via XML
- HTTP Basic Authentication Support

Future Plans

httperf 1.1 – Enhancement release of several additional features to be developed:

- GUI Interface to Manage Experiments
- Server Statistical Load Measurement
- Support for Multiple Sessions and Cookies

Conclusion

- httpperf produces empirical results on web system performance with which to make better decisions
- Robust tuning options for detailed measurements
- Latest version is 0.9.0, available from HP Labs FTP <ftp://ftp.hpl.hp.com/pub/httpperf/>

Thanks for listening!

Any questions?

References:

- Correspondence with Dr. Krishnamurthy, University of Calgary
- Correspondence with Martin Arlitt, HP Labs
- Correspondence with Mark Nottingham, Yahoo!
- httpperf man page and README documentation
- HP Labs httpperf mailing list archives, October 2000 – May 2007
- “httpperf: A Tool for Measuring Web Server Performance”
by D. Mosberger and T. Jin