Comparison of Script Characterization of web benchmarks

A presentation at SPECworkshop in Paderborn by the members of osgweb group

Barry Arndt - IBM
Hansfried Block - Fujitsu
Dean Chandler - Intel
Stephan Gradek - Fujitsu
Rema Hariharan - AMD
Sean Wu - Oracle
Overview

• The benchmark emulates web users accessing an application.
• Internet Banking, Ecommerce and Support download applications.
• Banking workload is 100% secure, Ecommerce workload is partly secure and Support is plain http workload.
• Applications run scripts in JSP/PHP/ASP
• How many user sessions that can be supported while meeting a pre-specified QOS?
• Web2009 also includes a power metric.
**SPECweb2009**

- **Prime Client**
  - FDR
  - reporter.jar
  - specweb.jar
  - PTDaemon
  - PTDaemon
  - PTDaemon
  - PTDaemon
- **Client 1**
  - specwebclient.jar
  - Temperature Sensor
  - Power Analyzer
  - USB
  - RS232/GPIB
  - TCP/IP
  - AC Power Source
- **Client 2**
  - specwebclient.jar
  - Temperature Sensor
  - Power Analyzer
  - USB
  - RS232/GPIB
  - TCP/IP
- **Client N**
  - specwebclient.jar
  - Temperature Sensor
  - Power Analyzer
  - USB
  - RS232/GPIB
  - TCP/IP
- **Web Server (Any OS)**
  - Any webserver
  - Workload Scripts
    - PHP/JSP/ASP.NET
  - HTTP
  - TCP/IP
  - Disk I/O
- **Storage Subsystem**
- **SUT**
- **BeSim**
  - Any webserver
  - API extension
  - (Linux, Solaris, Windows)
What is being characterized

- Performance of Scripts
- JSP, ASPX, PHP with SPECweb run on Linux and Windows
- Performance data running Olio (a web2.0 benchmark) on Solaris.
- All data collected with 2 processor systems
  - 8 cores
  - 1 Gb/s to Backend and 10 Gb/s to client
  - Local storage drives for data and logs
- Emulated 5000 user sessions
What we hope to achieve through this presentation

• Convince the audience about the performance differences between workloads and scripting methods used.

• Hint at the areas where software improvement might result in heavy performance improvements.

• Illustrate differences between web2.0 Olio based workloads and SPECweb workloads.
Platform independent and Script independent characteristics
• Request rate consistent between script types and software stacks
• This is a constant load based on QOS level
• Banking has highest request rate but lowest overall bytes per request
- Passing runs have constant send rate regardless of OS/script type
  - Banking 5058 bytes/session
  - Ecommerce 13908 bytes/session
  - Support 55490 bytes/session
Script Dependent characteristics

• CPU usage pattern
• Interrupts
• Context Switches
• DRAM usage
• Disk usage
- Banking workload creates highest CPU utilization due to SSL + encryption/decryption
- JSP lowest CPU utilization due to best pre-compiled performance
- PHP highest CPU utilization due to requirement to compile each request
- Banking PHP causes highest switch rate due to secure transactions and script compilations
- Linux lowest due to optimized SSL connections
Thread Context Switches per Request

- Ecommerce highest CS/req due to large amount of backend processing
- Banking CS/req is lower due to high number of requests and lowest network bytes per request
• Linux PHP workload has highest interrupt rate due to network I/O issues

• Linux JSP handles best for I/O and SSL handshake due optimized SSL stack
• Windows ASPX best memory usage better alignment to page size
• PHP scripts require higher memory bandwidth due to script compilation
DRAM Bytes per Request

- Windows ASPX best memory usage better alignment to page size
Olio

- This is a web2.0/cloud benchmark created by Sun/Oracle and UC Berkeley.

- Based on social event calendar application.

- Uses memcached, backend dB (mysql) and PHP scripts.
Olio Layout
Comparing Interrupts for Olio with SPECweb

![Chart comparing interrupts for Olio with SPECweb across different operating systems and applications](chart.png)
Comparing CPU for Olio with SPECweb
Ratio of Backend RCV to Client TX

![Graph showing the ratio of Backend RCV to Client TX for different services, with Olio having the highest ratio.](image-url)
Network bytes to Disk Bytes ratio

![Network to Disk Ratio Chart]

- Banking
- Ecommerce
- Support
- Olio
Highlights of differences between SPECweb workloads

• CPU usage load for Banking > Ecommerce > Support
• PHP cpu usage is lot higher than JSP or ASPX;
• PHP also has higher DRAM bandwidth usage due to script processing/compilation for each request.
• Windows shows lower DRAM bandwidth; but higher cpu utilization. (Reason ??)
• DRAM bandwidth for PHP scripts was lot higher than those for processed scripts like JSP and ASPX
How do the web2.0 workloads differ from what we have?

• Much higher backend traffic
• Much higher Disk traffic; perhaps close to SPECwebSupport.
• Much higher client to SUT traffic; includes a lot of images and data, resulting in higher writes to SUT/Backend.
• CPU usage is very similar to the Windows PHP/Support workload. Reason: it is handling PHP script processing.
• Network usage somewhat similar to SPECwebSupport.
Backup
DRAM bandwidth

**DRAM Bytes/s**

- Windows/Linux ASPX
- Windows/Linux JSP
- Windows/Linux PHP
- Ecommerce
- Windows/Linux ASPX
- Windows/Linux JSP
- Windows/Linux PHP
- Banking
- Windows/Linux ASPX
- Linux JSP
- Windows/Linux PHP
- Support

![Graph showing DRAM bandwidth comparison across different operating systems and applications.]