Implementing a Well-Performing and Reliable Portal

Adam Rybicki   arybicki@unicon.net

December 5, 2006

Agenda

I.   Introduction

II.   Tools Needed

III. Capacity Planning

IV.   Load Testing

V.   Automatic Failover

VI.   Performance Monitoring

VII. Performance Tuning

December 5, 2006

UNICON

Introduction

- ## What qualifies me to speak about this?

  - Have been working with uPortal since the beginning

  - Performed uPortal scalability study with Sun

  - Presented the study's results at the Winter, 2002 JA-SIG conference

  - Co-authored a white paper with Sun titled "JA-SIG uPortal Sizing Study" (http://www.sun.com/products-n-solutions/edu/whitepapers/pdf/uPortalatiForce.pdf)

  - Worked with many Unicon clients implementing uPortal to help them load-test uPortal

December 5, 2006

Introduction

Load-balancer

Database

Other Backend Systems

Tomcat/uPortal

December 5, 2006

UNICON™
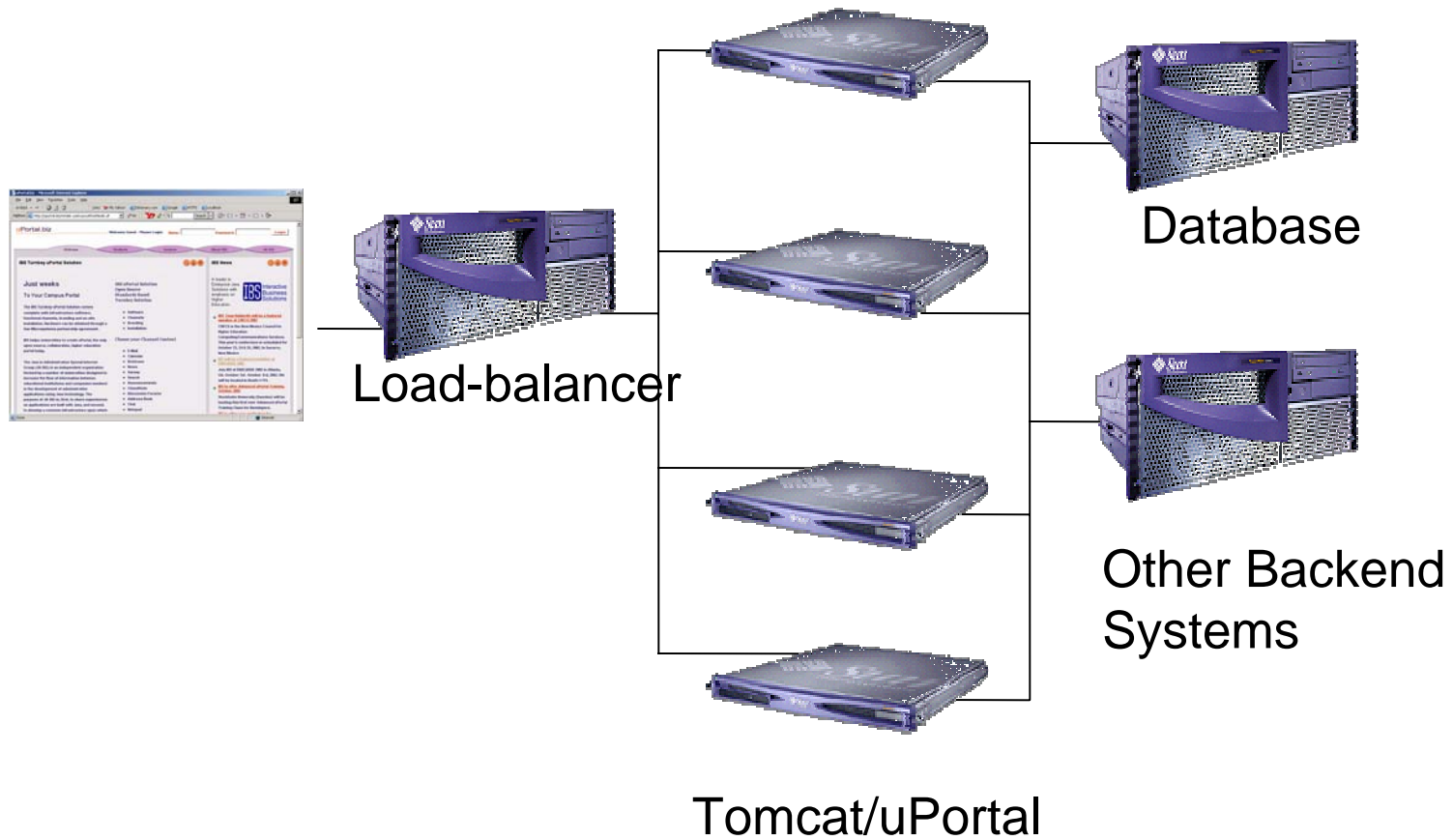
Definitions

- ## Scalability
  property of a system, a network or a process, which indicates its ability to either handle growing amounts of work in a graceful manner, or to be readily enlarged[1]

- ## Performance
  largely a function of the frequency and nature of inter-component communication, in addition to the performance characteristics of the components themselves, and hence can be predicted by studying the architecture of a system[2]

[1] wikipedia.org

[2] P.C. Clements *Coming Attractions in Software Architecture*, No.CMU/SEI-96-TR-003, Software Engineering Institute, Carnegie Mellon University, February 1996.

December 5, 2006

UNICON

Definitions

- uPortal Translations:

  - Performance
    response time to an individual HTTP request

  - Scalability
    an ability of the system to handle many individual HTTP requests while keeping the response time acceptably low

December 5, 2006

UNICON™

Definitions

- Vertical Scalability

  To scale vertically or scale up means to add resources to a single node in a system, such as adding memory or a faster hard drive to a computer.

- Horizontal Scalability

  To scale horizontally or scale out means to add more nodes to a system, such as adding a new computer to a clustered software application.

Source: wikipedia.org

December 5, 2006

UNICON

Tools Needed

- Load-generation

  Used to simulate many simultaneous Web users.

- Options:
  - Free*:
    - Apache JMeter
    - OpenSTA
  - Commercial:
    - RadView WebLOAD
    - Empirix e-Load
    - Mercury LoadRunner

* Nothing here is really free. The investment required to master the use of these tools will be significant.

December 5, 2006

UNICON

Tools Needed

- **Performance Monitoring**
  Used in this context to obtain and optimize the servers' memory and other resources usage.

- **Options:**
  - Free:
    - Sun's jconsole (from JDK 5)
    - CPU / Network / Memory utilization tools (usually supplied with the OS)
    - uPortal's Stats Recorder interface
    - CRuntimeData uPortal channel

    - ORCA (http://www.orcaware.com/orca/)
    - Others?
  - Commercial:
    - Quest Software's JProbe
    - Borland's Optimizeit
    - YourKit Java Profiler*

* Free licenses available to test open source code.

December 5, 2006

UNICON

Tools Needed

- **Load-balancing**

  Used to distribute the load among multiple servers, detect failed servers, and optionally encrypt HTTP traffic.

- **Options:**

  – Free:

  - Apache JK 1.2.x

  - Apache HTTP Server 2.x with mod_proxy

  - Pound

  - Others?

  – Commercial:

  - Zeus (both software and hardware)

  - CISCO (hardware)

  - F5 Networks Big IP (hardware)

UNICON

Capacity Planning

- Determine the acceptable response time

- Develop simulation scripts:
  - Record
  - Edit
  - Add random user generation
  - Add random delays
  - Deploy load-generation "drones"
  - Ramp up the load to avoid server overload

- Run the simulation on a single server

- Make sure that the server is well utilized before you decide what's the number of peak concurrent users per server

- Graph the results to visualize the server's response to increasing load

- Analyze the results

December 5, 2006

UNICON

Capacity Planning Demo

- Start HSQL
- Start Tomcat
- Start jconsole
- Log on to uPortal and select every tab to "prime" the server
- Run JMeter
- Wait for the average page time to stabilize
- Switch to "production" JVM settings
- Restart Tomcat
- Re-connect jconsole
- "Prime" the server
- Reset and re-run JMeter
- Unless there are other external performance factors, the response time should improve

December 5, 2006

UNICON

Capacity Planning

December 5, 2006

UNICON
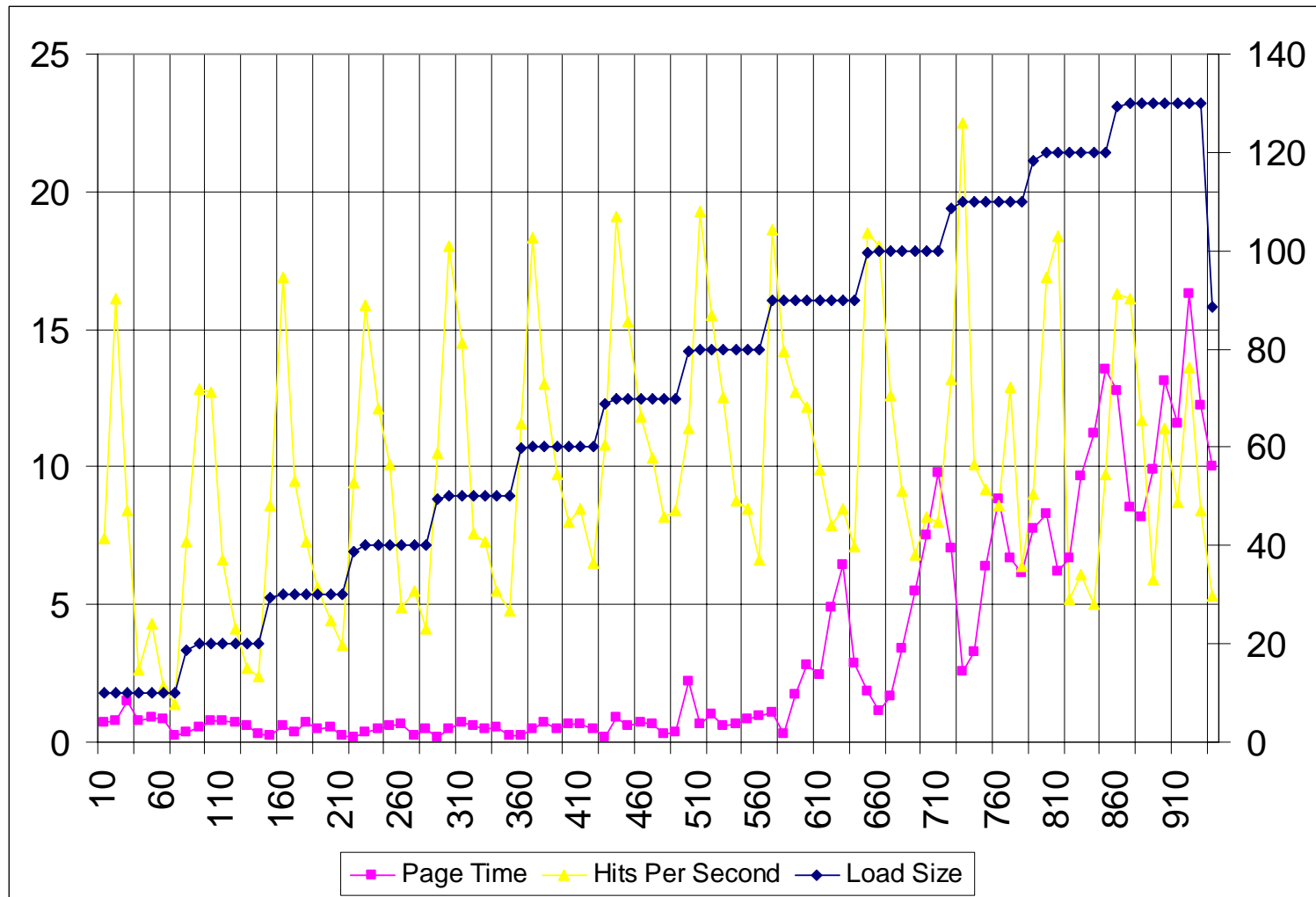
Capacity Planning

- This exercise assumes that the back end systems (RDBMS, LDAP, SIS, etc.) will scale to meet portal's demand

- Estimate the peak load (maximum number of concurrent users)

- Usually 5-10% of total user population

- Your number will depend on how peak-usage-prone your portal applications are (class registration, grade report, etc)

- Never plan to operate at or near peak capacity

- Arrange your server farm so that you can re-purpose your QA servers to production during peak times.

December 5, 2006

Load Testing

- Get a representative sampling of the target population
- Enhance the test scripts to randomly select users
- Consider using load-generation drone computers
- If practical, use the load-generation computers from different locations
- Monitor the load on all servers, including the back-end databases, directories, e-mail, etc.
- Detect bottlenecks, eliminate or improve, retest
- Repeat load testing with every significant new application added to the portal

December 5, 2006

UNICON

Performance Monitoring

- Page time (the load-generation tool will do that)

- Memory usage (jconsole from the JDK)

- If memory leaks are suspected, one of the commercial profilers may be needed

- "External" tools may not be able to pinpoint performance bottlenecks

- Utilize uPortal's *Stats Recorder* facility to gather per-portlet stats

- Performance monitoring tools may also be used to detect failed servers

- Some load-balancing solutions can monitor systems' performance to distribute the load according to actual load rather than the number of sessions

December 5, 2006

UNICON™

Performance Monitoring

- Many universities have developed custom Web pages to report their servers' cumulative statistics on one page

- Should some of this work be contributed back and included in the framework?

- Preserving historical stats essential for reports

December 5, 2006

Memory Leaks

- Memory leaks lead to performance degradation

- Over time this degradation will result in server failure

- uPortal framework has been extensively tested for resource leaks

- Some leaks may be caused by libraries uPortal uses

December 5, 2006

Living With Memory Leaks

- With enough memory the system's performance may remain good for a while

- Server restarts:
  - Scheduled downtime
  - Detect server failures and restart automatically
  - Load balancer should automatically pause sending clients to downed servers

December 5, 2006

JVM Tuning

- Many universities have tuned their JVM settings to optimize their portals' performance (http://www.ja-sig.org/wiki/display/UPC/JVM+Configurations)

- Sun's GC tuning page reads like a dissertation (http://java.sun.com/docs/hotspot/gc5.0/gc_tuning_5.html)

- Since no two portal implementations are the same, start simple and add more options only when necessary

December 5, 2006

UNICON

Sun's Garbage Collectors

- Copying
  - Default
- Parallel copying
  - -XX:+UseParNewGC
  - Multi-processor
- Parallel scavenging
  - -XX:+UseParallelGC
  - Multi-processor
  - Lots of memory (GBs)
- Incremental
  - -Xincgc
  - Single processor
- Concurrent
  - -XX:+UseConMarkSweepGC
  - Single processor
  - Lots of memory (GBs)

December 5, 2006

UNICON

A Pragmatic Start

- Use a "smart" garbage collector setting "`-XX:+AggressiveHeap`"
    - Inspects machine resources (memory, processors)
    - Attempts to set various parameters to be optimal for long-running, memory allocation-intensive jobs
    - Adapts the sizes of the young generation and tenured generation based on the application's behavior
    - Picks the best garbage collector for the server based on the number of CPUs and amount of memory
    - Grabs up to ½ of system's memory
- Add the `-server` option
- `-Xverify:none` can improve the startup time

December 5, 2006

Questions and Answers

Adam Rybicki

arybicki@unicon.net

info@unicon.net

www.unicon.net

December 5, 2006

UNICON