

# WELCOME!

## Back To School With SharePoint



Presented By:

David Haver  
Managing Consultant  
TCSC  
dbh@tcsc.com

Jim Brocato  
Managing Consultant  
TCSC  
jmb@tcsc.com

SharePoint

# Performance Optimization

# High Level Agenda



- Tools for Performance Tracing
- Base lining Your Performance
- Performance Optimization for WCM Sites
- General Performance Optimizations

# Tools for tracing Performance



- Fiddler
  - Fiddler is a fantastic client side HTTP debugging tool
  - Fiddler allows you to track all HTTP traffic outbound and inbound from a given client and report on many critical facts like:
    - Server HTTP Response (200, 304, 401 etc...)
    - Request /Response Size
    - Request/Response Time
    - Request/Response Headers, Cookies & etc...
- neXpert Fiddler Add On
  - neXpert is a Fiddler add in that provides trace aggregation and reporting
- URL Ping
  - This is a simple command line program for reporting page load times.
  - The source is provided in a MSFT whitepaper but you have to compile it into a program.
- SQL Profiler
  - Allows you to view SQL server performance and tracing information

# Tools

Demonstration



# Base Lining your SharePoint Performance



- The first step in doing performance optimization in SharePoint is to gather baseline information
- You can't improve what you can't measure
- Baseline Suggestions:
  - Schedule **urlping.exe** to run periodically throughout the day/week to retrieve page load data. Average this data out to get a baseline time for getting the first byte of data from the server.
  - Use Fiddler to capture the full page cycle information and timings for major pages and locations in your farm.
  - Use the neXpert add on to generate aggregated reports of the Fiddler data
- Spend time reviewing and validating these measurements in order to accurately develop your baseline information

# TCSC.com Baseline info



- With no caching enabled/optimizations made the TCSC.com website once visited had the following statistics over a DSL connection:
  - Response Time: 5 seconds
  - Round Trips: 21 (to retrieve files)
  - Bytes Sent: 15.33 kb
  - Bytes Received: 13.79 kb
- URLPing results (latency to load first byte from server)
  - 1.3 seconds
  - 1.6 seconds
  - 1.2 seconds...
  - Average of 1.3 seconds

# Web Optimizations



- SharePoint can run in many different environments and for many different purposes.
- When using SharePoint in a web content management (WCM) focused web site there are some specific optimizations that you can consider.
- Most internet facing SharePoint sites leverage WCM
- These optimizations are specific to WCM deployments because they typically have these characteristics:
  - Content that does not change dynamically (static content)
  - Anonymous Users
  - Page Performance over smaller connections (modem, DSL, cable...) is critical
- NOTE: Some of these capabilities are only enabled with the MOSS Standard publishing features

# Output Caching



- Microsoft Office SharePoint Server 2007 uses output caching technology native to ASP.NET 2.0 to manage when and how page content is served
- On a heavily accessed Office SharePoint Server 2007 site, caching frequently accessed pages for even a minute at a time can result in substantial throughput gains.
- While a page is cached by the output cache, subsequent requests for that page are served from the output page without executing the code that created it for the specified duration of the cache.
- SharePoint uses Cache Profiles to help you determine how it should cache content for a given site/site collection.
- While configuring your cache settings and working with various profiles you can enable cache debugging information at the bottom of the page.

# Object Caching



- The object cache is used internally to optimize page rendering by storing properties of sites, page layouts, and pages. The object cache reduces the amount of traffic between the Web server and a SQL database.
- You can optimize the object cache for a site collection by specifying several settings:
  - The size of the object cache. Specifying a larger number can enhance performance for some large sites at the cost of memory on each front-end Web server.
  - When to check the server for changes in a cross-list query. A cross-list query is one that displays content from multiple lists or libraries. Some Web Parts, particularly the Content Query Web Part, often display items from multiple lists and libraries. You can specify to check the server for every cross-list query or you can specify an amount of time (in seconds) in which the results can be drawn from the cache. This can improve performance, particularly for site collections that contain numerous cross-list queries.
  - A multiplier to use to retrieve more results than are requested. The multiplier is used to ensure that all client requests are accommodated in a cross-list query and a valid set of results is delivered.
- You can also use the Object Cache Settings page to reset (flush) both the object cache and the disk-based cache on an individual server or for the entire server farm.

# IIS Compression



- To more efficiently use available bandwidth, enable IIS HTTP compression for your static SharePoint resource files. HTTP compression provides faster transmission time between compression-enabled browsers and the IIS web server.
- Two types of compression are supported by IIS:
  - **Static file compression** – this is compression of the static resource files used in a web site/web application (html, js, etc...). These files are physically located on a file system.
  - **Dynamic file compression** – this is compression of dynamic elements that are produced as responses from your web application. Typically these files don't exist on the file system but are generated dynamically from database calls.
- **Switching on dynamic compression for SharePoint sites is not helpful and should be avoided.**
- Some older networking equipment (firewalls, switches, etc...) will not process compressed traffic well so be sure to test your configuration.
- IIS 6.0 – to enable HTTP compression in IIS 6.0 you need to use the adsutil to set the property for the w3svc. (Full instructions available on TechNet)
- IIS 7.0 – comes with two UI checkboxes in the IIS Manager tool and these are turned on by default

# Disk based Cache settings aka "BLOB" Caching



- If your Web application contains large files such as images and multimedia files, enabling disk-based caching improves page delivery time because the cache stores files on the front-end Web server, thus reducing database traffic.
- To enable the disk based caching you have to modify the web config file of the target web application on each web front end.
- Once you make the change all site collections stored in that web application will leverage the cache.
- The disk cache is only used for items stored in SharePoint libraries as its primary benefit is the reduction of database roundtrips
- Properties
  - Location – where to store the cached files
  - Path – extensions of file types to cache
  - Max Size - #GB of cache to support
  - Max Age (Optional) – specifies the maximum amount of time in seconds that the client browser caches BLOBs downloaded to the client computer. If the downloaded items have not expired since the last download, the same items are not re-requested when the page is requested.

Example:

```
BlobCache location="C:\blobcache"  
path="\".(gif|jpg|png|css|js|swf)$ "  
maxSize="10" enabled="false" />
```

# TCSC.com After Optimizations



- TCSC.com results with output caching, object caching & disk based “Blob” caching enabled over a DSL connection:
  - Response Time: 3.16 seconds - **68% improvement**
  - Round Trips: 6 – **Reduction of 15 files, 350% improvement**
  - Bytes Sent: 4.8 kb – **Reduction of 10k, 319% improvement**
- URLPing.exe results (latency to open first byte of data from server):
  - .32 seconds
  - .35 seconds
  - .32 seconds
  - Average of .33 secs – **Reduction of 1 second, 74% improvement**

# Web Optimizations

DEMO



# General Optimizations



- There are some general system performance optimizations that can help any SharePoint implementation.
- Hardware Optimizations include:
  - 64 Bit Servers
    - More memory for worker processes
    - Helps to avoid conflicts and memory errors from caching, application pools and OS sharing limited memory
    - Support for next version of SharePoint (2010)
  - Dual Gigabit Ethernet NICs
    - Optionally one NIC reserved for server to server traffic on a specific subnet

# Software Optimizations



- Windows 2008 & IIS 7.0
  - Better performance in IIS 7.0
  - Support for next SharePoint version
  - Windows 2008 has improved logging & monitoring features
- Keep plenty of space free on the C: drive for temporary files, cached files, solution & feature files (which require installation on C drive)
- If possible store your IIS Logs and ULS trace logs on a different disk from the main OS.
- Schedule daily application pool recycling (rolling blackouts) so that worker processes and caching do not suffer from memory bloat.

# Scale Control – Large Lists & Libraries



- Large Lists & Libraries pose a significant performance risk/bottleneck
- Large lists/libraries = a list or library with > 2,000 - 3,000 items in a single folder/view
- This is a soft threshold that as it grows will pose slower and slower page performance
- Creating smaller scoped views will solve the problem for web browsing traffic but does not solve the problem for alternate clients (Office Applications, WebDAV, etc..)
- Your best mitigation options are:
  - Create folders for items/files and ensure no folder has more than 3,000 items
  - When you create views of items ensure the view does not display more than 2,000 – 3,000 items at a time.
  - Consider splitting very large lists/libraries up into new lists/libraries

# Scale Control – Large Content DBs



- Excessively large content databases also pose a major performance risk.
- Content Databases are the container in SQL server for SharePoint site collections.
- There is no hard limit but generally speaking you want to limit to a max of 50 – 75 GB
- A content database should not grow beyond a comfortable backup and restore size and performance window
- The larger the content database the more items in shared tables that need to be searched through for any operation.
- General Rule Many small content DBs is better than a few large DBs
- Consider growth potential when you begin allocating site collection to content databases.

# Authentication Performance



- It is possible for authentication to become a performance bottleneck in your SharePoint environment if the domain controller (DC) receives requests more quickly than it can respond.
- For environments using user authentication such as NTLM, MSFT recommends a ratio of 3 Web servers per DC with a max supported limit of 4 Web servers per DC.
- Generally speaking the following is a ranking from fastest to slowest for authentication mechanisms.
  - Anonymous
  - Kerberos
  - NTLM
  - Basic
  - Forms
- General rule of thumb is that if you are running an intranet focused SharePoint farm you should use Kerberos for optimal performance.

# Large List Performance

Demonstrate



# Q&A



- Questions?

# References



- Microsoft IT SharePoint Optimization Whitepaper  
<http://download.microsoft.com/download/0/E/B/0EBA7263-A555-4279-B9DD-1720F0139A2E/SharePointOptimizationTWP.doc>
- How to Optimize a SharePoint Server 2007 Web Content Management Websites (MSDN Whitepaper)  
<http://msdn.microsoft.com/en-us/library/bb727371.aspx>
- Tune Web Server Performance (TechNet Article)  
<http://technet.microsoft.com/en-us/library/cc298550.aspx>
- Prescriptive Guidance for SharePoint Server 2007 Web Content Management Sites (TechNet Article)  
<http://msdn.microsoft.com/en-us/library/cc879144.aspx>
- Optimizing Office SharePoint Server for WAN environments (TechNet Article)  
<http://technet.microsoft.com/en-us/library/cc263099.aspx>