Welcome

Southeast User Conference

May 4–6 | Nashville, Tennessee
ArcGIS for Server: Optimizing Performance and Scalability

Anthony Myers & Kevin Armstrong
Introduction

- Today we will discuss
  - Architecture
  - Map and Image Services
  - What’s New
    - Tips and Techniques Throughout

- Goal: Provide valuable information that you may take back to your Organization and improve the implementation of ArcGIS Server
Architecture

- Deployment Scenarios
Map Services
Antony Myers
Map Services

• **Types of map services**
  - What’s new in the last few releases

• **Factors of map service performance**
  - Data access
  - Rendering speed
  - Image size/compression
Map Services

- 9.3.1 – 10.0
  - MXD based map services
  - MSD based map services (optimized map service)

- Use the optimized map service for best quality and performance
  - Analyzer workflow guides you through potential problems
Map Services – 10.1 and Beyond

• One unified map service
  - An updated optimized map service
  - Supports additional capabilities, data types, layers, renderers

• Includes extension capabilities optimized map service lacked:
  - Network Analysis
  - Geoprocessing*
Map Services – Dynamic Layers

- New behavior with the map service that allows for per-request changes to the map
  - Optional capability of map services
- May allow you to reduce the total number of services you need

- Allows for:
  - Updating renderers and symbols
  - Removing and reordering layers
  - Changing layer data sources
  - Adding new layers from registered data sources
Map Services – Dynamic Layers

- **Simple updates to the map service**
  - Remove layers or reorder layers

- **Thematic mapping**
  - Updates to renderers

- **Adding content from your data sources**
  - Find data from registered workspaces
    - Including query layers
  - Add to the map on a per-request basis
Demonstration

Register Map Services in ArcGIS Online
Map service with Dynamic Layers
Map Services – Factors to Consider

• Data access
• Rendering
• Image compression / size

• Consider all of these when creating a map service
Image Services

Kevin Armstrong
Tile Cache Data Flow

Client App

Client request tile →

Server (Security/Loadbalancer/TileHandler)

Search Tile Index
Read tiles from storage

≈20KB

Return Tile

Storage

Bundle Tile Cache

Decompress tile & display
Cache Tile Locally

Bottleneck? : Typically Network

ArcGIS for Server Performance and Scalability: Optimizing GIS Services
Image Service – Mosaic Dataset – Data Flow

Bottlenecks?:
- Data Read x N
- Image Processing x N + 1
- Data Transmission

Client App

Client request Extent/Cols/Rows

Server (Security/Loadbalancer/ImageService)

Determine Intersecting Rasters
For Each N rasters
  - Read Metadata
  - Compute byte ranges

For Each N rasters:
  - Decompress?
  - Resample/Reproject/Orthorectify
  - Bandmath/Atmospheric/Stretch

Mosaic Rasters (Clip/Seamline)
Apply Server Function (Bandmath, Stretch, Hillshade,...)

Storage

Raster datasets

100KB – 8MB

Data Transmission

Return bytes ranges

Decompress & display on screen

100KB - 32MB

ArcGIS for Server Performance and Scalability: Optimizing GIS Services
Cache vs Image Service
Image Service or Tile Cache?

• What is faster?
  - Tile Cache – If structured requests – Due to caching
  - Image Service – If random request and optimized – Due to single request

• What is more scalable on a specific set of HW?
  - Tile Cache is more Scalable
    - Uses Preprocessed data and caching at Server, Client and Internet (edge cache)

• What is more flexible?
  - Image Services - Provide access to full information content

• Image analysis & Interpretation
  - Image Services – Provide wide range of server based image processing
Recommendations:

- **General**
  - If just want static background Use Tile Cache
  - For dynamic imagery use image services
  - Do not add imagery to a Map and Publish as Map. (Best practice is to keep image as separate service)

- **Will split recommendations into:**
  - Data – How to structure
  - Storage Infrastructure
  - Mosaic Dataset Design
  - Server
  - App Requests
Data Recommendations
Minimize the amount of data the needs to be read

You may or may not have control of the following:

- **Format – Tiled:** Eg GeoTIF with internal tiles
- **Compression**
  - Helps reduce volume of data read from disk
  - Can add CPU Load to decompress. (JPEG good, Wavelet (JPEG2000) can be expensive)
- **Include Pyramids**
  - Reduced data read at smaller scales
- **Projection**
  - If possible create in same as majority output. (Suggest that do not pre-reproject)
Image Format
Storage Infrastructure
Ensure fast transfer to Servers

- Disk Storage
  - Needs to be Fast
  - Stripe Disks
  - Tune NAS
  - Storage performance can vary significantly

- Internal Network
  - Min 1GB between server and storage
  - If necessary use dedicated network for imagery

- For Smaller implementations
  - Consider using DAS (Direct Access Storage)
  - Use DAS for file geodatabase (see later)

ArcGIS for Server Performance and
Mosaic Dataset Design – See Guidebook
Minimize Amount of Processing


- **Create Overviews on Mosaic Dataset** – Reduce number of images accessed at smaller scales
- **Location of Mosaic Dataset** – MDs are chatty. Best to keep on DAS
- **Processing Functions** – Review. Especially regional functions
- **Check NoData** – Use Footprints to constrain extents. Turn off ‘Footprint contains no data’ if possible
- **Sampling method** – Nearest, Bilinear -10% Cubic -18%
- **Footprints** – Balance complexity and approximation. Possibly shrink & generalize
- **Split Mosaic Dataset by data type** – Use Suitable # Bands / BitDepth
- **Size of Allowed Table Attributes** – Too many fields slow down table display
- **Index** – Add Indices to fields that will be queried
- **Projection of Mosaic Dataset** – Hardly any effect
Mosaic Dataset Design
Server

- CPU – Faster Better
- Memory – 2-4GB/Core
- Internet Access
- Virus Checker – Can be a real hog!
Stop Client Bandwidth - Compression for transmission
- Set to JPEG (Q>=80) for continuous
- Set to PNG for discrete
- On Web Reduce PNG request (caused by NoData) when using JPG/PNG

Start User Server Functions – Allow server to do processing

Size Cols/Rows of request
Client
Techniques & Tips
Anthony Myers & Kevin Armstrong

Esri SEUC | Technical Workshop | ArcGIS for Server: Optimizing Performance and Scalability
Techniques & Tips

- GIS Systems are bound by:
  - CPU - typically
  - Memory – when large number of services
  - Disk – Image Service, Synchronization
  - Network – low bandwidth deployment
  - Poorly configured virtualization can result in 30% or higher performance degradation
Techniques & Tips

• Monitor Server Statistics

• Take advantage of immutable IDs

• Utilize LERC for image Service Caches

• Enable Suggest with your 10.3 Geocode Services

• Use the “Check for software patches and updates” tool
## Pooling tab

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
<th>When to adjust it</th>
</tr>
</thead>
</table>
| Minimum number of instances per machine            | Minimum number of instances of the service to create on each node within the cluster. Default = 1 | • Increase if there are frequently more requests than available instances.  
• Decrease if there are frequently more available instances than requests. |
| Maximum number of instances per machine            | Maximum number of instances of the service to create on each node within the cluster. Default = 2 | • Increase if clients are frequently waiting for an available instance.  
• Decrease for infrequent requests. |
| The maximum time a client can use a service        | Maximum number of seconds an instance can service a request. Default = 600 seconds (10 minutes) | • Increase for requests that may take longer (e.g., geoprocessing jobs).  
• Decrease for services that complete quickly to free system resources. |
| The maximum time a client will wait to get a service| Maximum number of seconds the framework will wait for a free instance of the service before rejecting a service request. Default = 60 seconds | • Increase on a busy system. |
| The maximum time an idle instance can be kept running| Maximum number of seconds an idle instance of a service must be kept alive before it can be destroyed. Default = 1800 seconds (30 minutes) | • Increase if new instances are created frequently.  
• Decrease for infrequently requested services to free system resources. |
## Optimize Services

### Processes tab

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
<th>When to adjust it</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recycling Settings</td>
<td>Duration between service recycling. Default = 24 hours at 00:00</td>
<td>• Decrease to reclaim resources and clear HTTP connection issues.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• You require that a service recycles at a different time of day.</td>
</tr>
<tr>
<td>Health Checks</td>
<td>Duration between data connections check &amp; repair (idle instances). Default = 30 minutes</td>
<td>• Decrease if SDE connections frequently break.</td>
</tr>
</tbody>
</table>
# Optimize Services

## Parameters Tab

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
<th>When to adjust it</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum Number of Records Returned by Server</td>
<td>Maximum number of records returned for a service request. Default = 1000</td>
<td>• Increase if the service will be returning large numbers of records when fulfilling requests. This will increase network usage and possibly overload the client.</td>
</tr>
<tr>
<td>Lock Database Schema</td>
<td>Specifies if the map service will acquire schema locks for map layers that come from a geodatabase. Default = True (checked)</td>
<td>• Disable when you need to make schema changes to your data without stopping your services.</td>
</tr>
</tbody>
</table>
Techniques & Tips

- Backup your configuration store
Demonstration

Map Services

Monitor Server Stats
Techniques & Tips

- **Best Practices**
  - Don’t expose Server Manager or Admin interfaces to the Public
  - If you don’t need to Query ... Disable it
  - Avoid publishing public services to your Enterprise Database (Use Replication)
  - Services that need security should be secured with GIS or Web tier security
    - GIS tier: token based security
    - Web tier: Public Key Infrastructure (PKI) or Integrated Windows Authentication (IWA)
  - Principle of least privilege: Role Based Access Control (RBAC)
  - Encryption – when securing content communicate over HTTPS with a valid SSL Cert
Techniques & Tips

- http://www.arcgis.com
- owner:EnterpriseImp
- Show ArcGIS Desktop Content
Questions
Thank you

- Please take a moment to fill out the session survey
Esri
Understanding our world.