ArcGIS for Server Performance and Scalability—Optimizing GIS Services

Andrea Rosso (Esri), Craig Williams (Esri), David Cordes (Esri), Eric Miller (Esri)
Optimizing map services

Craig Williams
Overview

• Types of map services
  - What’s new at 10.1?

• 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 at 10.1

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

- New extension capabilities:
  - Network Analysis
  - Geoprocessing*
Mapping capabilities

- Added data source support
  - XY events
  - Linear referencing events

- Added feature layer renderer support
  - Dot density
  - Charts
  - Geostats

- Added support for layers
  - Dimensions
  - Schematics
  - Network
  - Network Analysis
  - Tin
  - Terrain
  - Tracking
Dynamic Layers: The Concept

- 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
Dynamic Layers: Use Cases

- Simple updates to the map service
  - Remove layers or reorder layers
- Thematic mapping
  - Updates to renderers
- Adding content from a data library
  - Find data from registered workspaces
    - Including query layers
  - Add to the map on a per-request basis
Factors of map service performance

- Data access
- Rendering
- Image compression / size

Consider all of these when creating a map service.
Data access

- Local data will draw faster than remote data
- **Spatial index**
  - Do you have one? (e.g. XY Events)
  - Is it sized correctly?
  - Universal features, slow all draws
- **Attribute indexing**
  - Not always needed
Data access case study: X Y Events

- Often used in cases where data comes from external systems
- A draw typically requires a complete row scan

- Alternative
  - Use a native spatial type in your database
    - Query layers
  - Insert features via SQL in external systems
  - Features will be indexed and draw much faster
Data access troubleshooting

- Publishing analyzers indicate lack of a spatial index etc.

- Evaluate index efficiency
  - The number of features returned for each draw query
  - Large index grid sizes lead to too many features being drawn

- Evaluate I/O performance if using remote data

- Unnecessary attribute indexes
  - May confuse query plans in some databases
  - Don’t index fields just because they exist in a def query
Rendering speed

- Optimized map services were introduced at 9.3.1 to resolve performance bottlenecks at this stage

- Remaining areas to be concerned with:
  - Complex effects (e.g. geometric effects in representations)
  - Inline annotation (aka “Bloated” annotation)
  - Anti-aliasing performance
    - Higher levels use more RAM and are slower
    - Text anti-aliasing has a negligible effect in most cases
  - Layer transparency
Rendering speed: transparency

• Layer transparency is applied to a layer as a whole
  - Involves a full layer blend

• Alternative: Use color transparency
  - Capability of optimized map services
  - Enabled via an option from analyzer warning 10009
Image compression / size

- Smaller images are faster to download

- Image formats have limitations
  - e.g. limited color palettes, lossy compression

- Image compression itself has a performance penalty
  - Use the preview window to evaluate performance of image type

- Evaluate size and performance in a test service in network conditions
  - Balance size vs. quality when choosing the image type based on your needs
Image compression / size (con’t)

- Image type used for cached services affects:
  - Download size
  - Storage size of the cache
    - Portability

- For caching vector data
  - Consider new PNG image type at 10.1
  - Chooses the correct PNG type (8, 24, 32) for each tile based on content
  - Low content areas use less storage.
ArcGIS Online

Andrea Rosso
ArcGIS Online as a Case Study

- Very Large Scale deployment
- Multi-tenant
ArcGIS Online as a Case Study
**www.arcgis.com** Web Site

- The Face of ArcGIS Online
  - Must perform

- Javascript API Front End
  - Dojo
  - REST Client
    - ArcGIS Server APIs
    - Portal API
    - WMS
    - ...

Optimizing the Network

- **Builds**
  - Coalesce (build) Javascript Files into fewer files to reduce downloads

- **Compression**
  - Make sure Web Server Gzip compression is enabled

- **CDN + Versioning**
  - Edge cache static files in a CDN (Akamai, CloudFront, etc.) for faster performance around the world
  - Improvements of 200%+ in page load times
  - Version resources to enable you to update information periodically.
    - /cdn/1916/js/dojo/dojo/dojo.js
ArcGIS Online Basemaps
Initial Basemap System Requirements

- **Capability:** Publish Map Services for use in all ArcGIS Clients
- **Capacity:** Support > 1,000 Concurrent Users
- **Availability:** Provide 24x7, 99.9% Availability
- **Performance:** Sub-second Response on Server & High-speed access by User
- **Flexibility:** Rapidly Deploy New and Updated Maps
ArcGIS Online Basemaps over time
ArcGIS Online Basemap Architecture

Users

Web / ArcGIS Servers

Data Appliances

Akamai
ArcGIS Server 10.1
Performance Improvements

Eric Miller (Esri)
Source of 10.1 Performance Improvements

- ArcGIS Server is now fully 64 bit
  - No memory limit per service instance
  - Contributes to improved performance

- DCOM-free framework
  - Number of service instances no longer OS limited
  - Enables more scalability options and better performance not limited by firewall considerations

- Moved most WebService handler processing to core

- More efficient core services
Performance/Scalability improvements

- All services have improved performance
  - Windows
    - Faster transaction times and greater throughput
  - Linux
    - Much faster transaction times and much greater throughput

All this despite security being enabled by default

- All services and map caching scale linearly
Map Service Performance

% Throughput Change: 10.1 vs. 10.0 SP2
(Map Service export)

Throughput %Δ

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Throughput %Δ</th>
<th>Windows</th>
<th>Linux</th>
</tr>
</thead>
<tbody>
<tr>
<td>WorldSoils</td>
<td>57</td>
<td>65</td>
<td></td>
</tr>
<tr>
<td>Corine</td>
<td>37</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>EarthQuakes_Time</td>
<td>28</td>
<td>73</td>
<td></td>
</tr>
<tr>
<td>Portland</td>
<td>33</td>
<td>62</td>
<td></td>
</tr>
<tr>
<td>Colombia</td>
<td>27</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>StreetMap</td>
<td>27</td>
<td>43</td>
<td></td>
</tr>
<tr>
<td>Naperville</td>
<td>12</td>
<td>44</td>
<td></td>
</tr>
</tbody>
</table>

Increasing Map Complexity

Windows | Linux
Map Service Performance

% Throughput Change: 10.1 vs. 10.0 SP2 (Map Service Identify/Query)

Throughput % Δ

- Portland_Identify: 93% (Windows), 133% (Linux)
- Portland_SpatialQuery: 3% (Windows), 70% (Linux)
- Portland_AttributeQuery: 48% (Windows), 31% (Linux)

Map Service Operation

- Windows
- Linux
Map Service Scalability (Windows)

Map Service: Export

- **CORINE_export**
- **StreetMap_export**

Throughput (Tr/Hr) vs. Number of GIS Servers in Site
Geocode Service Performance

% Throughput Change: 10.1 vs. 10.0 SP2 (Geocode Service FindAddress/ReverseGeocode)

Throughput % Δ

Throughput:

- FAC_NAComposite
- FAC_NAComposite_SLL
- ReverseGC_NAComposite

Operating Systems:
- Windows
- Linux

Values:
- FAC_NAComposite: 32, 38
- FAC_NAComposite_SLL: 23, 29
- ReverseGC_NAComposite: 45, 209
Geocode Service Scalability (Windows)

Geocode Service: Find Address Candidates (FAC), Reverse Geocode (RGC)

Throughput (Tr/Hr)

Number of GIS Servers in Site

NAComposite_FAC
NAComposite_RGC
Feature Service Performance

% Throughput Change: 10.1 vs. 10.0 SP4:
Feature Service Edits/Queries
PostGres (PG), Versioned (V), Nonversioned (NV), Archiving (A)

Throughput % Δ

Feature Service Operation

Windows  Linux
Geoprocessing Service Performance (Windows)

% Throughput Change: 10.1 vs. 10.0 SP2 (Geoprocessing Service)

Geoprocessing Model

WorldViewshedM 130
WorldViewshedPY 27
WalkTimes 36
PortlandDataExtract 50
MailingList 135
PopByZip 25
HotSpots 161
Geoprocessing Service Scalability (Windows)

PopulationByZip Model

Throughput (Tr/Hr) vs. Number of GIS Servers in Site
Image Service Performance

% Throughput Change: 10.1 vs. 10.0 SP2
(ImageService export/query)

Data Source and Method

- Windows
- Linux
Image Service Scalability (Windows)

MDSurdexTIFF: export

Throughput (Tr/Hr) vs. Number of GIS Servers in Site

- Throughput (Tr/Hr) ranges from 0 to 500,000.
- Number of GIS Servers ranges from 1 to 8.

The graph shows an increase in throughput as the number of GIS servers increases.
WMS Service Performance

% Throughput Change: 10.1 vs. 10.0 SP2
(WMS getMap from MapService and Image Service)

Data Source and Service Type

Windows | Linux
WMS Service Scalability (Windows)

WMS Service: getMap

Throughput (Tr/H4) vs Number of GIS Servers in Site

- CORINE_MS_getMap
- StreetMap_MS_getMap
- MDSurdexTIFF_IS_getMap
Geometry Service Performance

10.1 % Change in Throughput vs. 10.0 SP2 (Geometry Service)

Method Name

<table>
<thead>
<tr>
<th>Method</th>
<th>Throughput % Increase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Union</td>
<td>720</td>
</tr>
<tr>
<td>TrimExtend</td>
<td>539</td>
</tr>
<tr>
<td>Simplify</td>
<td>606</td>
</tr>
<tr>
<td>Reshape</td>
<td>257</td>
</tr>
<tr>
<td>Relation</td>
<td>282</td>
</tr>
<tr>
<td>Project</td>
<td>275</td>
</tr>
<tr>
<td>Offset</td>
<td>67</td>
</tr>
<tr>
<td>Lengths</td>
<td>3</td>
</tr>
<tr>
<td>LabelPoints</td>
<td>542</td>
</tr>
<tr>
<td>Intersect</td>
<td>75</td>
</tr>
<tr>
<td>Generalize</td>
<td>548</td>
</tr>
<tr>
<td>Distance</td>
<td>283</td>
</tr>
<tr>
<td>Difference</td>
<td>140</td>
</tr>
<tr>
<td>Density</td>
<td>126</td>
</tr>
</tbody>
</table>

Windows
Linux
Cache Size and Completion Time (Windows)

Creation Time (Windows): 1 node, 8 service instances

Cache Creation Time (minutes)

Cache Size (MBytes)

Map Name and Version

Cache Size (Windows)
Cache Creation Scalability

Windows/Linux: Cache Tile Generation Rate (Bundles/Minute)

- Corine (Windows: Local)
- Corine (Linux: Local)
- Streetmap_UT_CO (Windows: Local)
- Streetmap_UT_CO (Linux: Local)
ArcGIS Server 10.1
Managing Performance
David Cordes (Esri)
Managing Performance at 10.1

- Managing Service Instances
- Managing Machines using Clustering
Managing Service Instances at 10.1

- 10.0: min/max instance settings are total *for all machines*
- 10.1: min/max instance settings are *per-machine*

- Simplifies sizing – removes the guess work
- Scales as your hardware scales
Managing Service Instances at 10.1

On each machine

- Start min # instances when service starts
- Start another instance \textit{if}
  - All existing instances busy
  - Less than max # of instances
- Stop an instance \textit{if}
  - Idle time for instance > idle time out
  - More than the min # of instances
Managing Service Instances at 10.1

- Typical use: cyclic and gradual
- Usually min instance at least 1.
- Hardware or license constrained?
  - Max instances less than natural
  - Idle time out may be less (default 30 min, but maybe 5, 10, or 15)
Managing Service Instances at 10.1

- Two other common use patterns
  - Blip
  - Constant

- Blip
  - Service is pre-dominantly not used.
  - Used for short periods of time by few people.
  - Min/Max is usually 0/1.
  - Idle time-out depends on business needs.

- Constant
  - Service usage does not vary much day-to-day. Usage often tied to a resource – workers, trucks, etc.
  - Min/Max are usually set to be equal. Variability is not desired.
Managing Machines through Clustering at 10.1

- Cluster is a set of ArcGIS Server machines.
- Each service is assigned to a cluster.

**Example:**
- Cluster default has Sample service
  - Machine1
  - Machine2
- Cluster 2
  - Machine3
  - Machine4

When you publish a service S1 to cluster 2, the service will run on Machine3 and Machine4
Managing Machines through Clustering at 10.1

- You can move machine between clusters on the fly.

- Example continued – move machine2 from default cluster to cluster 2.
  - Cluster default has sample and publishing services
    - Machine1
    - Machine2
  - Cluster 2 has service S1
    - Machine2
    - Machine3
    - Machine4
  - When you publish a service S1 to cluster 2, the service will run on Machine3 and Machine4
Managing Machines through Clustering at 10.1

- You can move services between clusters on the fly (requires service to be stopped and restarted).

- Example continued – move sample service from default cluster to cluster 2
  - Cluster default has publishing service
    - Machine1
    - Machine2
  - Cluster 2 has S1 and sample services.
    - Machine2
    - Machine3
    - Machine4
  - When you publish a service S1 to cluster 2, the service will run on Machine3 and Machine4
Managing Machines through Clustering at 10.1

- **Performance implications:**
  - More important services can have dedicated resources.
  - Heavy services (caching or geoprocessing) can be isolated from lighter services (map)
  - You can rebalance load as needed.
  - Important: All machines in a cluster should have identical hardware.
Further Reading & Sessions
## Performance Related UC Sessions

<table>
<thead>
<tr>
<th>Session</th>
<th>Time &amp; Place</th>
<th>Short URL</th>
</tr>
</thead>
</table>
## Performance Related Reading

<table>
<thead>
<tr>
<th>Resource</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enterprise GIS Resource Center</td>
<td><a href="http://resources.arcgis.com">http://resources.arcgis.com</a> (Enterprise GIS Community)</td>
</tr>
<tr>
<td>Geoprocessing Services</td>
<td><a href="http://bit.ly/NrUGSb">http://bit.ly/NrUGSb</a></td>
</tr>
</tbody>
</table>
Evaluation & Questions
Evaluation

- Your feedback is extremely important
- Please fill out evaluations online
- [http://Esri.com/ucsessionsurveys](http://Esri.com/ucsessionsurveys)
  - Offering ID (Wednesday): 672
  - Offering ID (Thursday): 1938