Designing and Using Cached Map Services

Eric Rodenberg & Tom Shippee
What’s covered in this session

• Agenda
  - Map cache basics
  - Map cache best practices
  - Map cache administration
  - Map cache clients
What’s covered in other sessions

• Image service caching
  - Caching Imagery using ArcGIS
  - Understanding Pyramids, Overviews and Caching

• Caching in the cloud
  - Building and Maintaining ArcGIS Online Hosted Tiled Services
  - Using the Power of Amazon EC2 to Build Map Caches

• Advanced topics
  - Enterprise Architectures for Large Map Caching Projects
  - ArcGIS Server Performance & Scalability—Optimizing GIS Services
  - Automating Cache Workflows and Building Tile Usage Heat Maps
Map cache basics
Understanding caching concepts
How does a map cache work?
Three reasons to care about map caches

• Performance
• Scalability
• Cartographic quality
Many layers fused into one tile

Cache tile

1:16000
1:8000
1:4000
1:2000
1:32000

Transportation
Hydrography
Landbase
Authoring of map document

- Use scale dependent display
- Group layers by scale
- Use annotation
- For mash up use ArcGIS Online scales
What should you cache?

• Base maps

• Operational layers that are
  - High volumes of traffic
  - Don’t change often
  - Cover small scales only
Best performing image formats

- Vectors few colors
  - PNG (auto selects bit depth)
- Vectors many colors
  - MIXED with 90 quality
- Imagery
  - MIXED with 55 quality
- Vectors or labels + Imagery
  - MIXED with 90 quality
Choosing the best image format

- Large number of continuous colors
  - JPEG (start with quality = 55)
  - Mixed (if transparency required)

Which one looks better?

JPEG 96 – 25KB  JPEG 96 – 30KB
What is antialiasing?

- High quality line/label rendering on vector maps
- Web standard (Google, Bing, AGOL)
- Takes LONGER to cache
Building a map cache

Publish, configure and build a cached map service.
Map caching
best practices

Strategies and techniques
Understanding cache structure

- Bundle: 8 x 8 supertiles
- Supertile: 16 x 16 tiles
- Standard tile: 256 x 256 pixels
- Total tiles: 16,384
Track cache status

- **Status.gdb**
  - File geodatabase in caching folder
  - Accessed by reporting tools for status
  - Records cache task progress
  - Identify completed tiles

- Copied to an ArcMap folder
  - Cache error review
  - Time aware cache analysis
Anatomy of a cache

Explore a 3D time aware view of cache generation.
Supertiles and Labeling

• **ArcGIS Server Draws Large Areas**
  - Reduces duplicate labels

• **Duplication May Occur**
  - Use Annotation or MapPlex Labels with Rules
  - Use Map Server Cache Tiling Scheme To Polygons
You don't need to generate everything

- Cache by feature
  - Polygon features
  - Generates all tiles for intersecting supertiles

- Saves on...
  - Generation time
  - Processor resource
  - Disk usage

NM highway case study:
Build 20 of 64 supertiles for the bundle shown
Pre-create coupled with cache on demand

- Pre-create high use areas
  - Population centers
  - Parks, roads, attractions

- Features
  - Cover popular extent
  - Generate key tiles
  - All others generated on demand
The ideal tiles to cache on demand

• Few simple features
  - Barren homogenous area
  - Rarely accessed

• Large scales only
  - Draw relatively fast
Handling tiles you do NOT create

- Create “No Data” tile
  - Same image format (JPG or PNG)
  - Same size (256 x 256)
  - Save in cache folder
    ...
    ...
    ...
    ...
    ...
    ...

- How to
  - Knowledge base article 36939 has sample files
Build a test cache and note the following

- Creation time
- Appearance
- Client performance
- Cache size validation
Map cache administration
Generate and update techniques
Setting the Number of Instances

- Cache Tools Geoprocessing Service
  - Start with $N$
  - $N =$ CPU’s per server
  - See cloud session for Amazon recommendations

8CPU’s = 8 Instances

8 CPU
System caching services

• System services
  - Caching Tools: Sets caching instance per machine
  - Caching Controllers: Assign cache jobs to instances

• Manage Map server Cache Tiles
  - Controls instances per job
  - Set to -1 to use all instances
Update a cache using a staging server

**Staging ArcGIS Server Instance**
- Map service
- All layers for cartography of map service

**Production ArcGIS Server Instance**
- Map service
- Layers for TOC and Query

**Cache folder**

(On-demand caching needs the full map to build the cache)
Isolate caching to certain servers

- Organize GIS Servers into Clusters
  - Generate Cache on its own cluster
  - Scale or reconfigure while caching
Isolate caching to certain servers

• Organize GIS Servers into Clusters
  - Generate Cache on its own cluster
  - Scale or reconfigure while caching

ArcGIS Site
Cache failure & recovery – Out of the box

- Out of the box tools
  - Course grained for locating extents
  - Fix errors: Re-cache extents where errors reported

![Cache Status -- Streetmap](image)
Recovering cache failure

Use Status.gdb to identify and rebuild errors.
Cache failure & recovery – sample tools

- Cache Validation Tools
  - Fine grained for locating tiles by file size
  - Custom tool available via resources.arcgis.com

![Image of Cache Validation tool](image-url)
Cache update automation

• Use Model Builder to script update automation
  - Rebuild Specific Tiles
  - Export to Python
  - Schedule Run Time

• Useful update tools
  - Compare feature classes
  - Show edits since reconcile

• See demo theater
  - Automating Cache Workflows and Building Tile Usage Heat Maps
Cache export tool

• Export tiles
  - Based on extent or polygon features
  - Convert storage format
  - Use for cache import or as a disconnected cache

Exported using Nevada and Utah state boundary features.
Cache import tool

- Import tiles
  - Based on extent or polygon features
  - Must have same storage format

Import from a previously exported map cache.
Map cache clients

Optimizing web applications
Using multiple domains

• With multiple services
  - Use a different domain for each service

• With one service
  - API’s support multiple URLs for a single layer

```javascript
var layer = new esri.layers.ArcGISTiledMapServiceLayer(
    "http://www.mydomain.com/arcgis/rest/services/myMap/MapServer",
    { tileServers: [
        "http://cache1.mydomain.com/arcgis/rest/services/myMap/MapServer",
        "http://cache2.mydomain.com/arcgis/rest/services/myMap/MapServer"
    ] });
```

- Cautions
  - Use with small cache tiles
  - Reduce browser caching results in more HTTP connections
Request large scales as dynamic maps

- Supported by map services & Web API’s
  - Cached (most scales)
    - Use: ArcGISTiledMapServiceLayer
  - Dynamic (largest scales only)
    - Use: ArcGISDynamicMapServiceLayer
Dynamic & tiled hybrid

Configure a web application to make dynamic requests at large cache scales.
Thank you…

Please fill out the session evaluation

First Offering ID: 1224
Second Offering ID: 1336

Online – www.esri.com/ucsessionsurveys
Paper – pick up and put in drop box