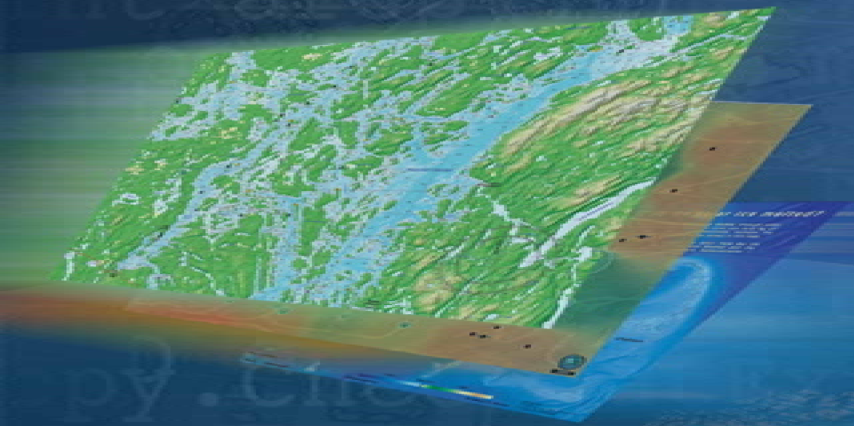
ESRI Developer Summit

March 22–25, 2010
Palm Springs, CA

Advanced Map Caching Topics

Sterling Quinn

Tom Brenneman



Schedule

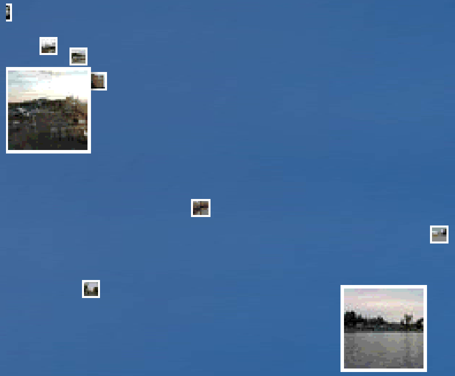
Please!

Turn OFF cell phones
and paging devices

- Advanced topics
  - Key caching properties
  - Caching strategies
    - Test caches
    - Strategic cache creation
    - On-demand caching
    - Creating a very large cache
    - Update strategies
  - What’s new in ArcGIS 10
- We will answer questions at the end of the session

What should you cache?

- Base maps
- Operational layers that satisfy one of the following:
  - High volumes of traffic
  - Don’t change often
  - Cover small scales only



Choosing coordinate system and scales

- ArcGIS Online (legacy)
  - WGS 1984 coordinate system
- Google Maps & Bing Maps
  - WGS 1984 Web Mercator coordinate system
- Create your own

Overlaying the new ArcGIS Online

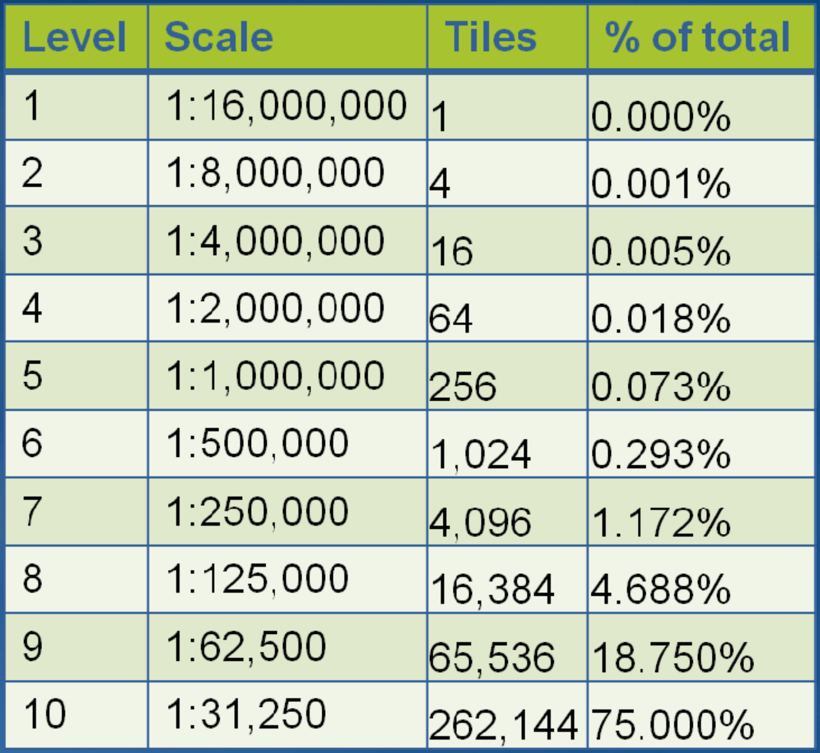
- **Choice 1: Project your map to WGS 1984 Web Mercator and choose Google Maps / Microsoft Virtual Earth from the ArcCatalog dropdown.**
  - Datum transformations more difficult this way
  - Won’t work in some clients (.NET ADF)

- **Choice 2: Project your map to WGS 1984 Web Mercator (Auxiliary Sphere) and follow the steps in KB 37329**
  - Ideal choice if you have datum transformations
  - Works in all clients

Creating your own scales

- Build just the scales you need
  - Determine closest scale (Raster resolution)
  - Divide scale by 2 for each subsequent scale
  - Adjust smallest scale to full extent
- Consider Web Mercator scales

Sample 10 level cache

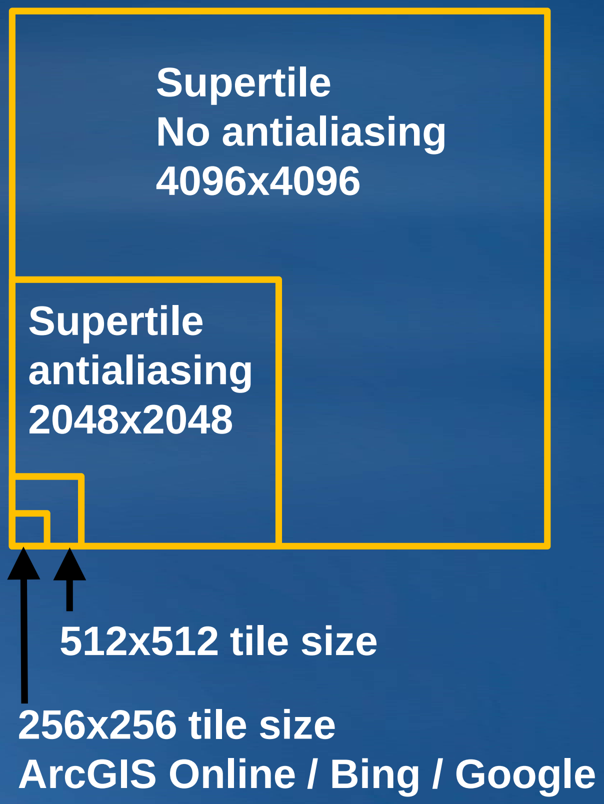


Level	Scale	Tiles	% of total
1	1:16,000,000	1	0.000%
2	1:8,000,000	4	0.001%
3	1:4,000,000	16	0.005%
4	1:2,000,000	64	0.018%
5	1:1,000,000	256	0.073%
6	1:500,000	1,024	0.293%
7	1:250,000	4,096	1.172%
8	1:125,000	16,384	4.688%
9	1:62,500	65,536	18.750%
10	1:31,250	262,144	75.000%

Final level is ~75% of the total

Authoring labels for the map

- Individual tiles are cut from large area (supertile)
  - 4096 x 4096
  - 2048 x 2048 if using antialiasing
- Supertile necessary to
  - Reduce duplicate labeling
  - Reduce requests to map service when caching
- Labeling rules can repeat across super tile boundaries
  - Maplex places better labels
  - Annotation



Tile size

- Pixel dimensions of each image
- 256x256 is defacto web standard
  - 512 X 512 : legacy ArcGIS Online
- Larger dimensions are faster to build, but tiles take longer to download

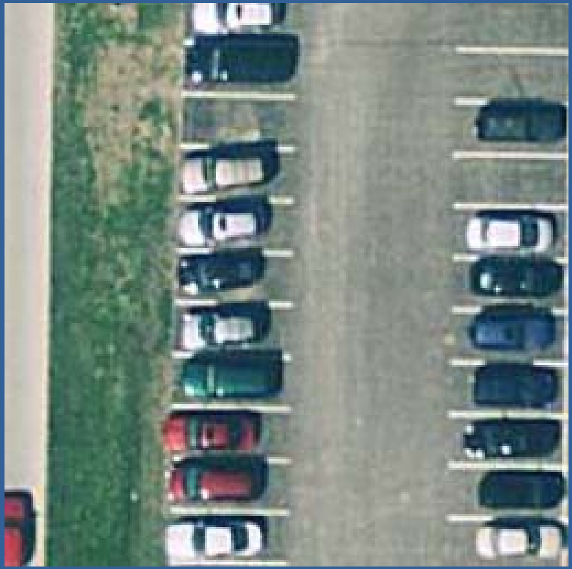
Choosing an image format

- Image format affects
  - Tile storage space requirements
  - Web application performance (speed and supported browsers)
  - Tile image quality
  - Tile transparency

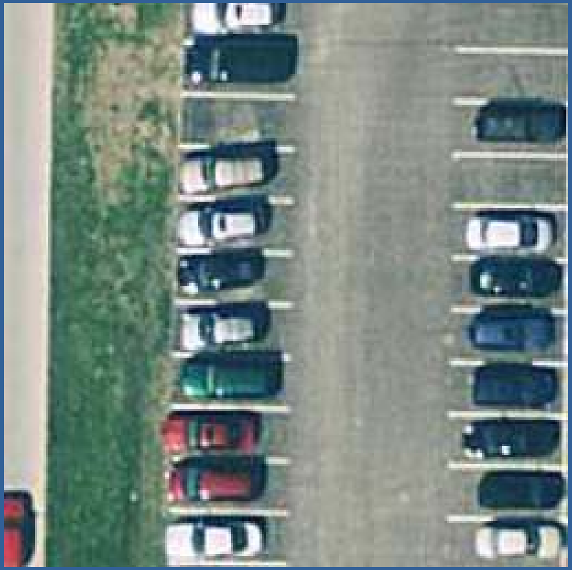
Basemaps: Image format guidelines

- Many colors (Continuous symbology)
  - JPEG
    - Small file size for many colors, no transparency
    - Quality settings range from 55 (orthos) - 90 (vector maps)
- Fewer colors (~256)
  - PNG 8

Which one looks better?



JPEG 90 – 21KB

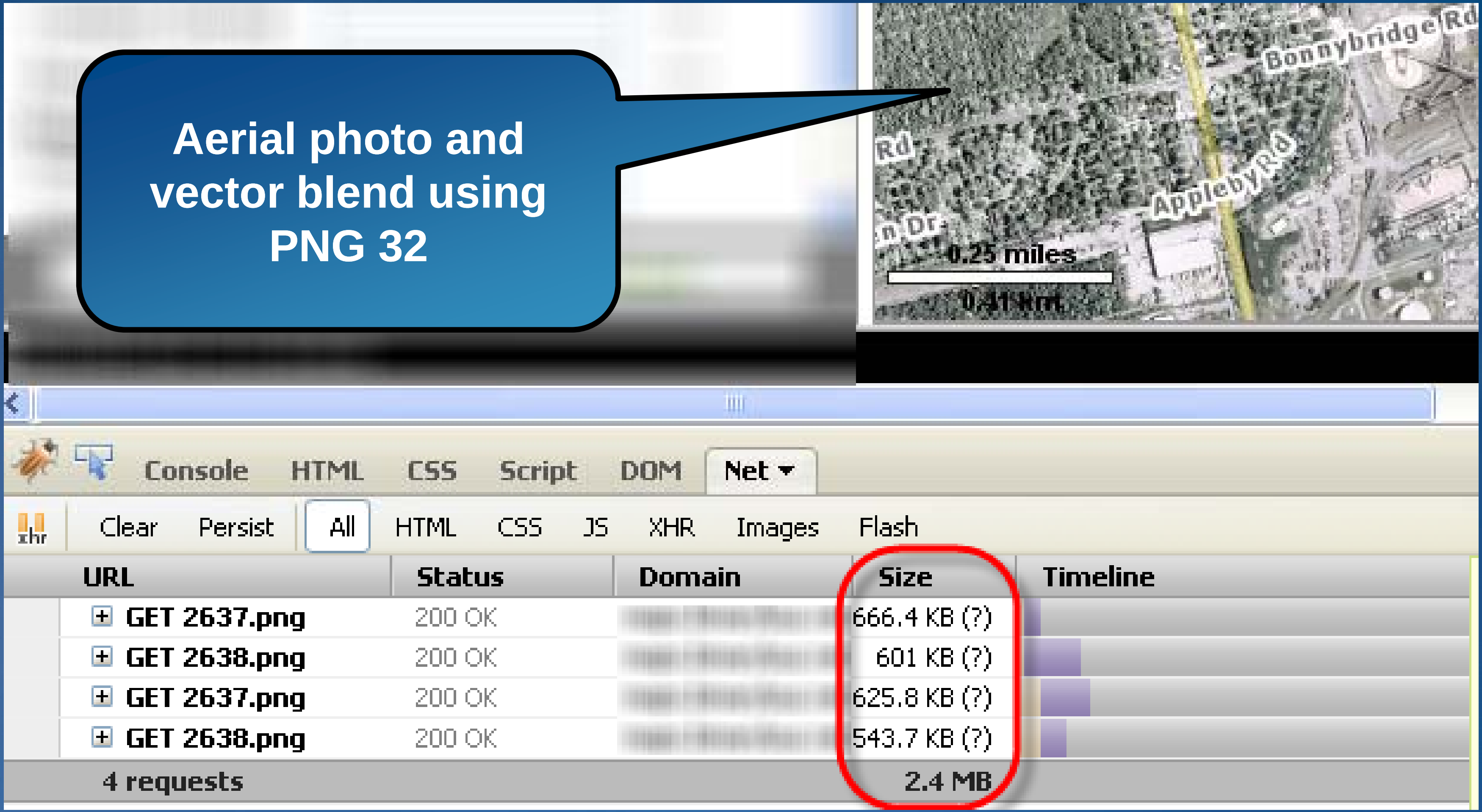


JPEG 55 – 10KB

Operational layers: Image format guidelines

- PNG 8
  - Small size on disk + transparency support
  - Not for imagery
  - Use MSD-based service + heavy testing if over 256 colors
- PNG 32
  - Over 256 colors
  - Good for vector overlays with antialiasing
  - Caution: Large tile sizes
- (PNG 24)
  - Avoid in Web apps (poor IE 6 support)
- Solar Boston

Example: Tiles are too large



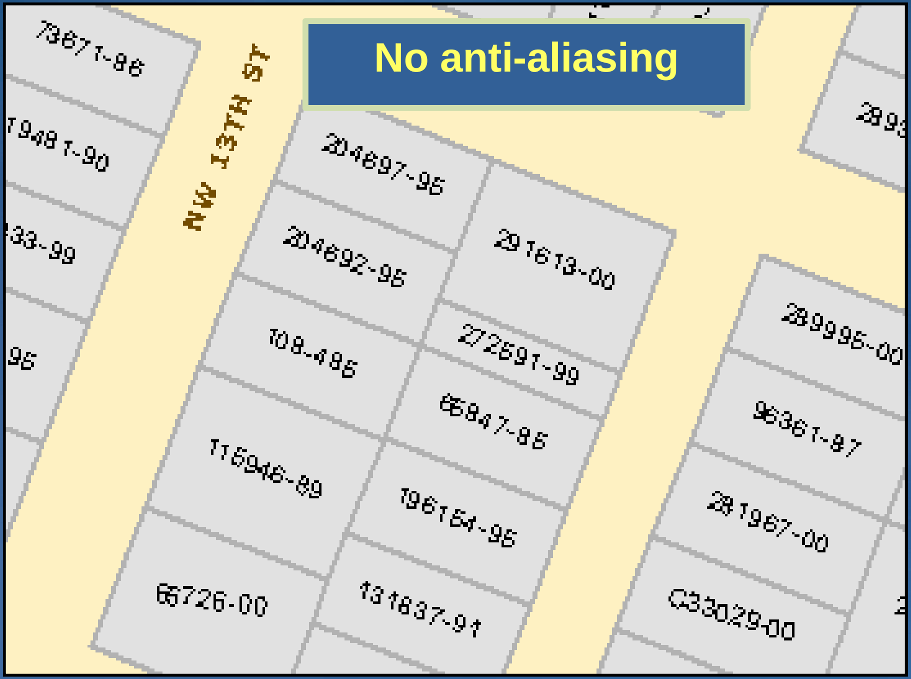
Aerial photo and vector blend using PNG 32

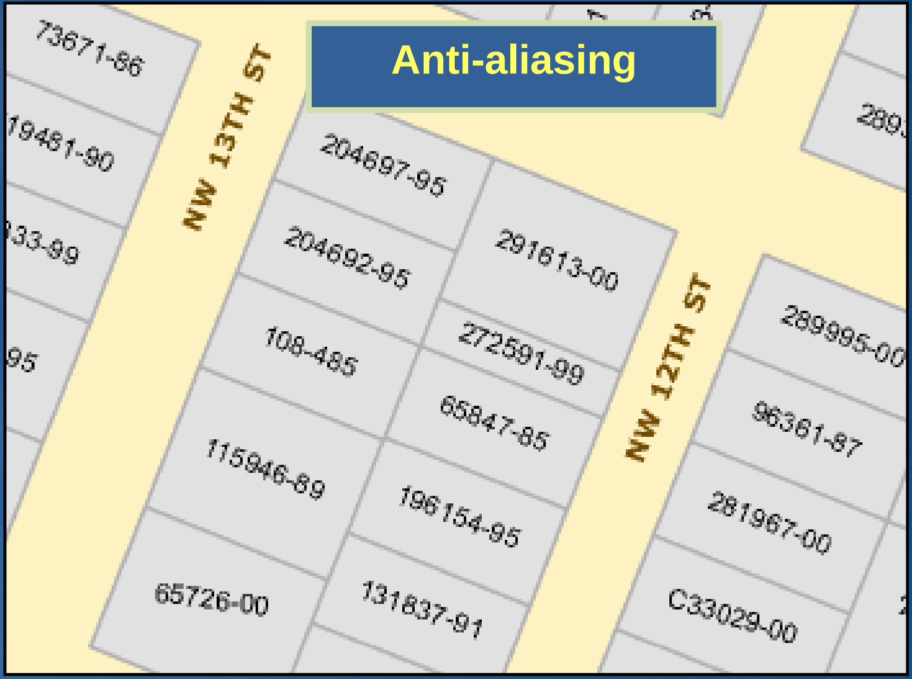
Bonnybridge Rd
Rd
Appleby Rd
n Dr
0.25 miles
0.41 km
Console HTML CSS Script DOM Net
Clear Persist All HTML CSS JS XHR Images Flash

URL	Status	Domain	Size	Timeline
⊞ GET 2637.png	200 OK	[redacted]	666.4 KB (?)	
⊞ GET 2638.png	200 OK	[redacted]	601 KB (?)	
⊞ GET 2637.png	200 OK	[redacted]	625.8 KB (?)	
⊞ GET 2638.png	200 OK	[redacted]	543.7 KB (?)	
4 requests			2.4 MB	

When should I use antialiasing

- High quality line and label appearance on vector maps
- Web standard (Google, Bing, AGOL)





- Optimized map services preferred for antialiasing (speed and appearance)

Caching strategies

- Test caches
- Strategic cache creation
- On-demand caching
- Creating a very large cache
- Update strategies

TEST CACHES

Creating a test cache

1. Select test area with varying geography
2. Create a simple feature class covering the test area
3. Create tiles at all scale levels based on this feature class

Davis County
Bountiful
68
Salt Lake City
215
270
West Valley City
Magna
Millcreek
East Millcreek
Taylorsville
Kearns
89
Murray
Salt Lake County
West Jordan
Cottonwood Heights
Sandy
71
Camp Williams Military Res

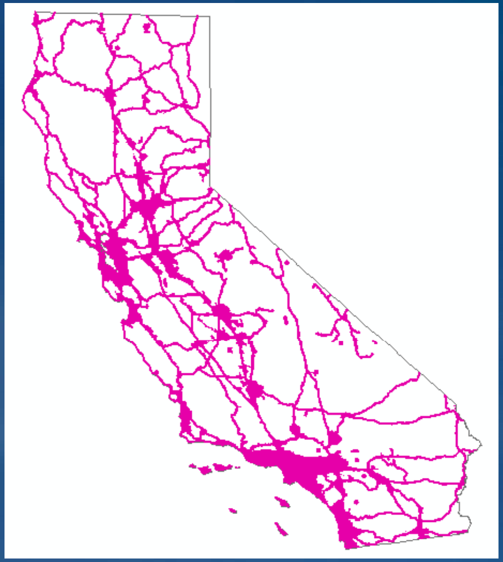
Note the following from your test cache

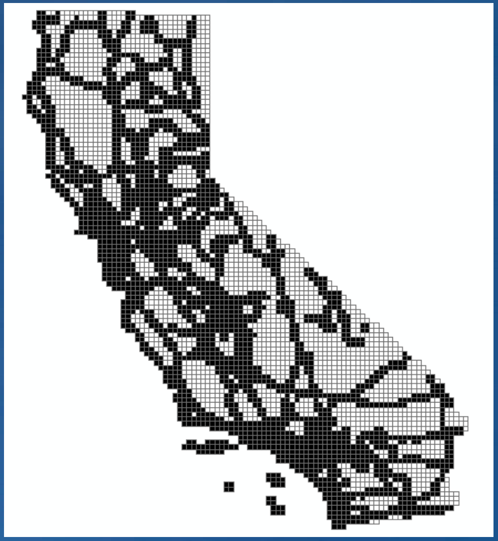
- Appearance of tiles
- Performance of tiles in client
- Cache creation time
- Cache size on disk

STRATEGIC CACHE CREATION

Strategic cache creation

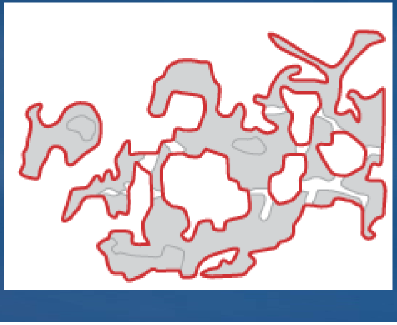
- Feature class covers about 25% of California
- Contains about 97% of California's population
- Caching with this layer saves:
  - 943,000 tiles
  - 9.3 GB of space
  - 17 hours of caching time
- Bing analysis of access
  - http://hotmap.msresearch.us



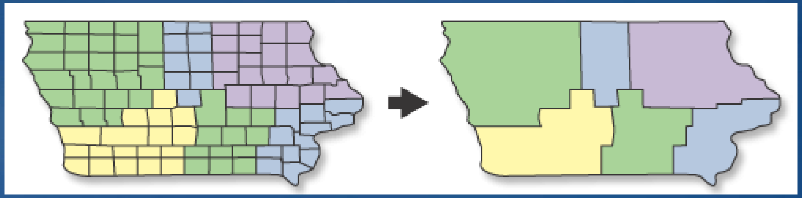


More tips for caching by feature class

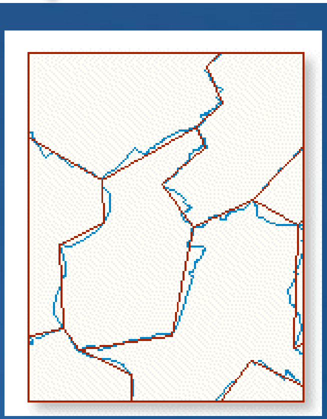
- Feature class coordinate system should match map you’re caching



- Avoid numerous small features
  - Aggregate Polygons tool
  - Dissolve tool



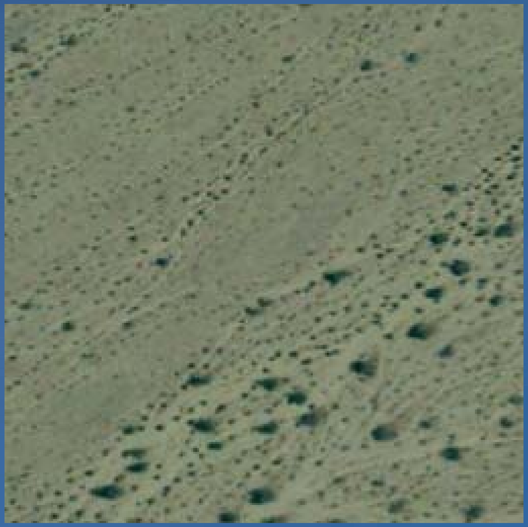
- Avoid excessive vertices
  - Simplify Polygons tool



- Tips for caching by feature class blog post

The ideal tiles to cache on demand

- Few simple features
  - Barren homogenous area
- Rarely accessed
- Draw relatively fast
- Large scale



Displaying a “missing” tile

1. Create a missing tile
   - Same image format as cache (missing.png or missing.jpg)
   - Same dimensions as cache tile (e.g. 256x256)

2. Save file in arcgiscache\MapService\Layers_alllayers

3. Web ADF clients – remove virtual directory from server directory

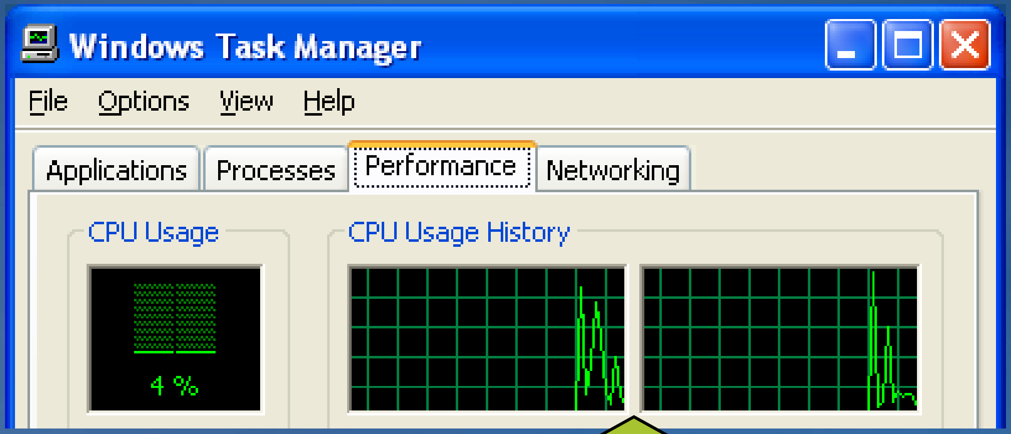
- Support article 36939 has sample files

CREATING A VERY LARGE CACHE

Setting the number of instances

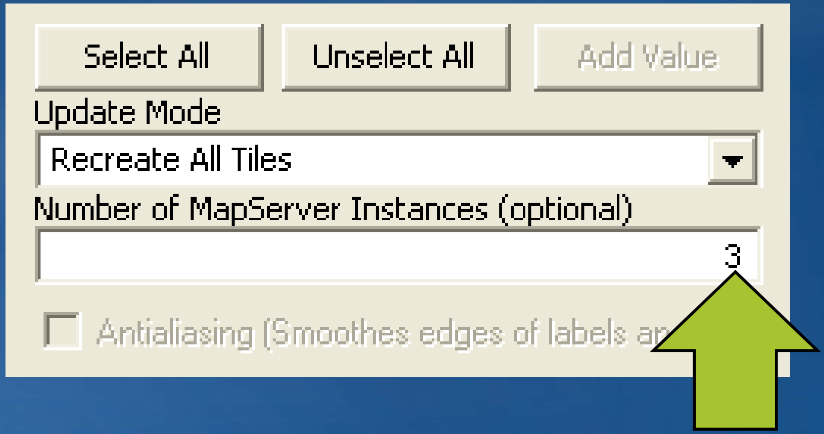
- Keep your CPU below 100%
  - Start with N+1 instances where N = the number of cores on server
  - Adjust number of instances based on test cache

Task Manager



2 CPU’s

Manage Map Server Cache Tiles



Update Mode
Recreate All Tiles

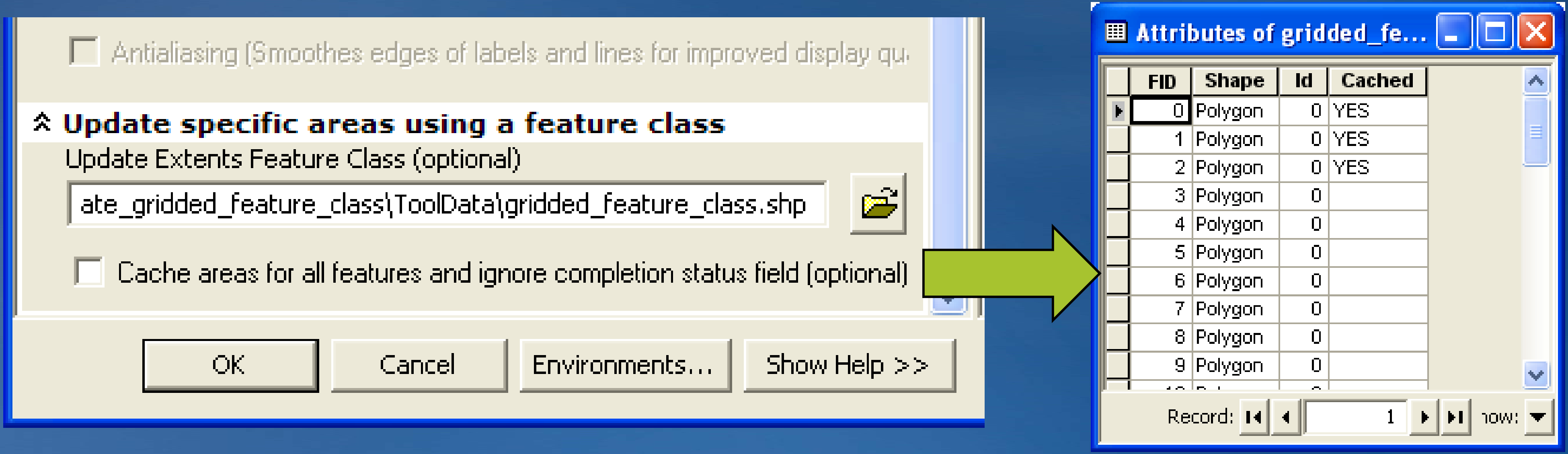
Number of MapServer Instances (optional)
3

□ Antialiasing (Smoothes edges of labels an

3 Instances

Creating a very large cache

- Use Update specific areas using a feature class option
  - Use status field to track creation
  - Divide study area into manageable chunks
    - i.e. 4 hours worth of cache creation
    - Helpful scripts: Create gridded feature class for tracking ArcGIS Server map caching jobs



☐ Antialiasing (Smoothes edges of labels and lines for improved display qu

Update specific areas using a feature class

Update Extents Feature Class (optional)

ate_gridded_feature_class\ToolData\gridded_feature_class.shp

☐ Cache areas for all features and ignore completion status field (optional)

FID	Shape	Id	Cached
0	Polygon	0	YES
1	Polygon	0	YES
2	Polygon	0	YES
3	Polygon	0	
4	Polygon	0	
5	Polygon	0	
6	Polygon	0	
7	Polygon	0	
8	Polygon	0	
9	Polygon	0	

Record: 1

State of Indiana Imagery Basemap case study

- Requirements
  - Single statewide basemap with most recent data
    - Maintain 6” resolution where available
  - Web Mercator tiling scheme
  - Integrate neighboring data for small scales
  - Record metadata for imagery in basemap
- Stats
  - Image format: JPEG 55
  - 62,245,522 tiles
  - Total cache size 452.72 GB
  - Average tile size 7.63 KB
  - 3 weeks to build

Cache built in stages

Web Mercator Tiling Scheme

6" scale	576.00	20	Individual 6" counties
	1,128.50	19	Use gridded tile
	2,256.99	18	
	4,513.99	17	
	9,027.98	16	Clip to state boundary
	18,055.96	15	
	36,111.91	14	
	72,223.82	13	
	144,447.64	12	
	288,895.29	11	IN Full extent
	577,790.58	10	
	1,155,581.15	9	
	2,311,162.31	8	
	4,622,324.61	7	
	9,244,649.23	6	
	18,489,298.45	5	Not Built
	36,978,596.91	4	
	73,957,193.82	3	
	147,914,387.60	2	
	295,828,775.30	1	
	591,657,550.50	0	

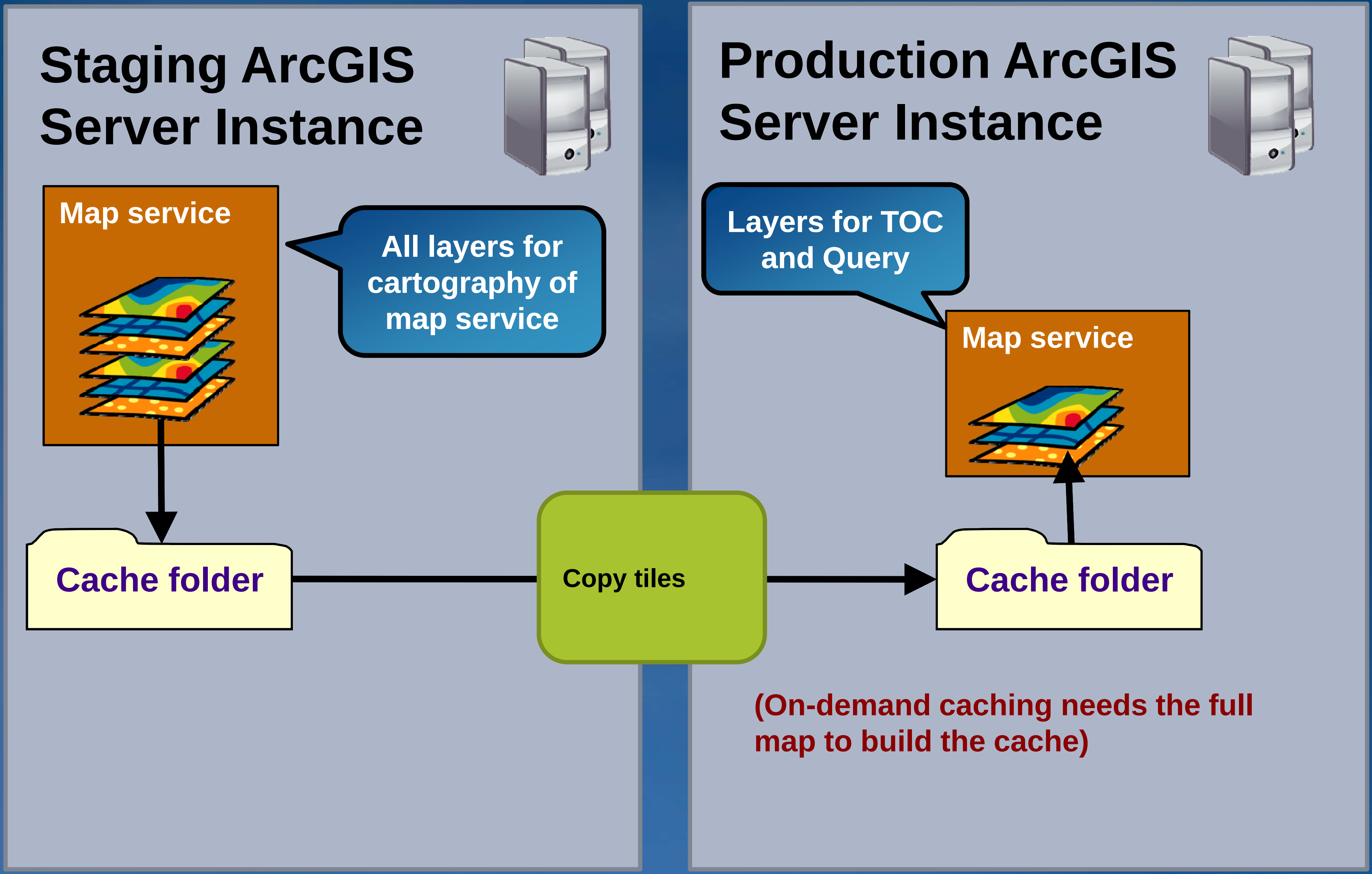
UPDATE STRATEGIES

What affects cache update strategies?

- Size of cache
  - Rebuild entire cache
- Scales containing updates
  - Rebuild specific scales
- Location of updates
  - Rebuild specific areas

- Cache updates can be scripted with geoprocessing
  - With ArcGIS 9.3.1

Update a cache using a staging server



Staging ArcGIS Server Instance

Map service

All layers for cartography of map service

Cache folder

Copy tiles

Production ArcGIS Server Instance

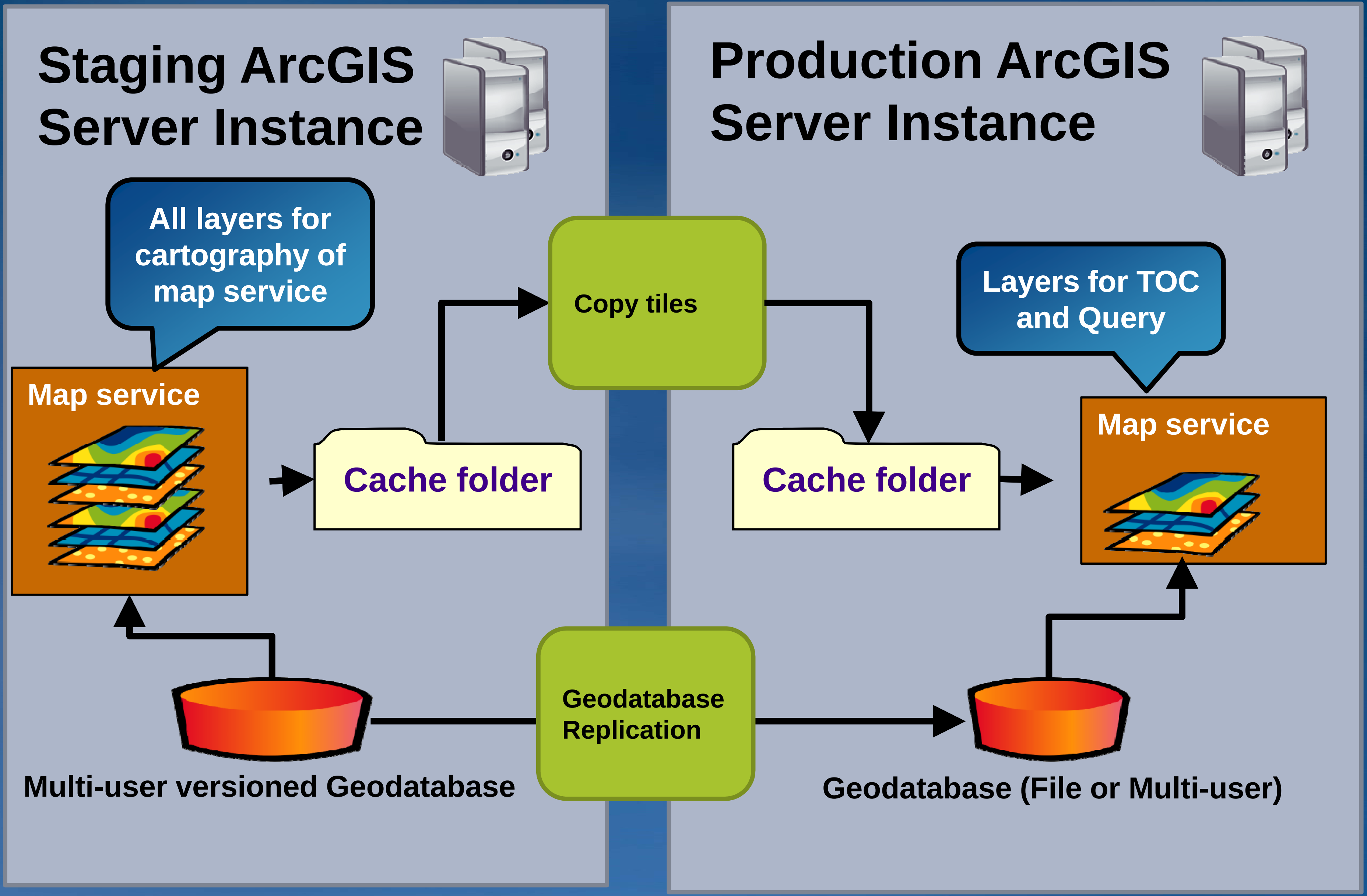
Layers for TOC and Query

Map service

Cache folder

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

Update a cache and data using a staging server



Staging ArcGIS Server Instance

All layers for cartography of map service

Map service

Cache folder

Copy tiles

Multi-user versioned Geodatabase

Geodatabase Replication

Production ArcGIS Server Instance

Layers for TOC and Query

Cache folder

Map service

Geodatabase (File or Multi-user)

Copying caches

- Windows Copy/Paste inefficient for large caches
- Use XCOPY/ROBOCOPY command
  - /D switch will only copy new tiles
- Third party utilities (SecureCopy) may be helpful for large caches

Disk maintenance

- Disk cluster size can bloat storage for small tiles (PNG8 @ 256x256)
  - Default minimum cluster size on Windows is 4k
  - Lowering to 1k (or less) can reduce “Size on disk”
  - Requires you to store cache on dedicated partition or disk
- Turn off backup
- Turn off virus scanning

Cache distribution

- HTTP 1.1 spec constricts browsers to two simultaneous downloads

- Improve cache retrieval performance by using multiple hosts
  - Can be one server with multiple DNS entries (cache1.mydomain.com, cache2.mydomain.com)

- Geographically distribute ArcGIS Server instances

Using multiple domains

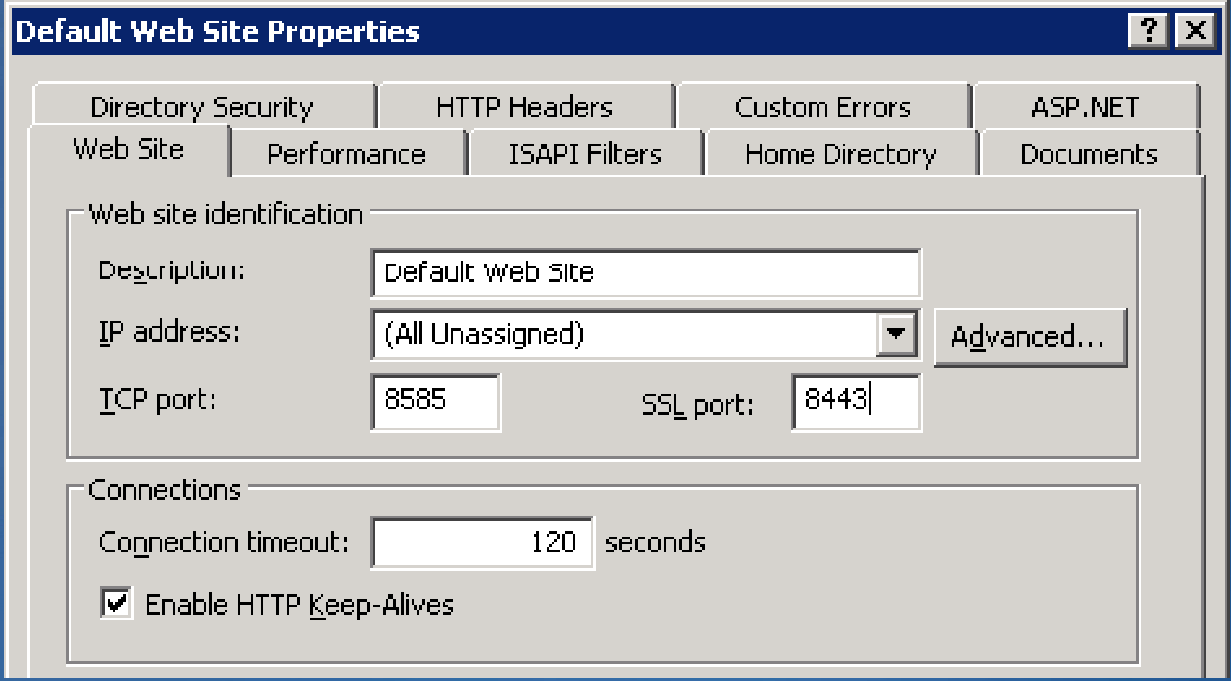
- With multiple services
  - Use a different domain for each services
- With one service
  - API’s support multiple web services endpoints for a single layer

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

- Use with small cache tiles
- Can reduce browser caching and result in more HTTP connections

HTTP connections

- Turn On HTTP KeepAlive for best performance
  - Multiple image requests don’t need to open a new connection
  - Biggest benefit with one host (~2-3 times faster tile requests)
- Apache: KeepAlive directive
- IIS



Web site identification

Description: Default Web Site

IP address: (All Unassigned)

TCP port: 8585 SSL port: 8443

Connections

Connection timeout: 120 seconds

☑ Enable HTTP Keep-Alives

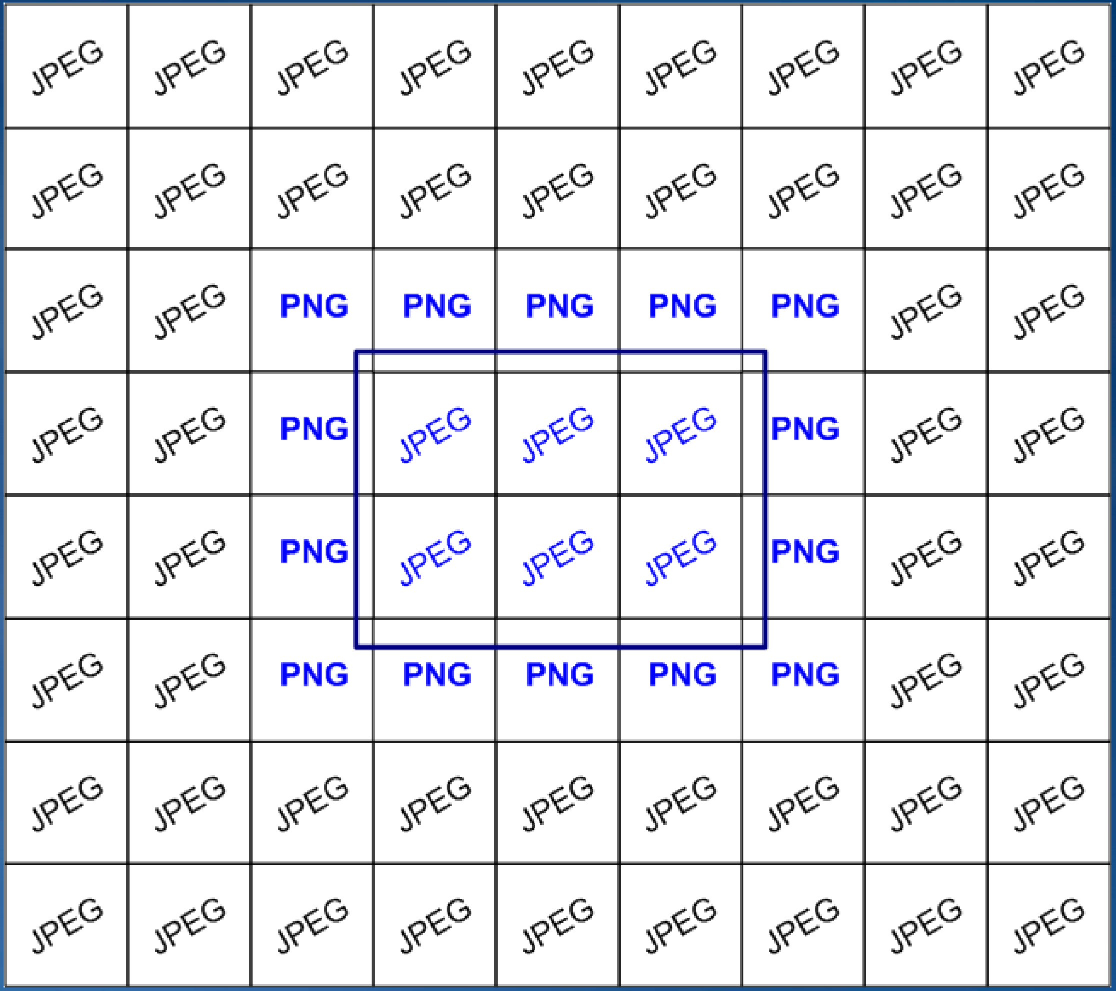
Helpful scripts

- Geoprocessing resource center code gallery
  - Cache validation
  - Create gridded feature class
  - Compare feature classes
- ArcGIS Server > Web Applications > .Net ADF resource center code gallery
  - Show edits since reconcile

WHAT’S NEW IN ARCGIS 10

Mixed mode image format

- Creates JPEG tiles unless transparent pixels detected
- If transparent pixels detected, creates PNG32
- Enormous space savings for large caches
- Demo



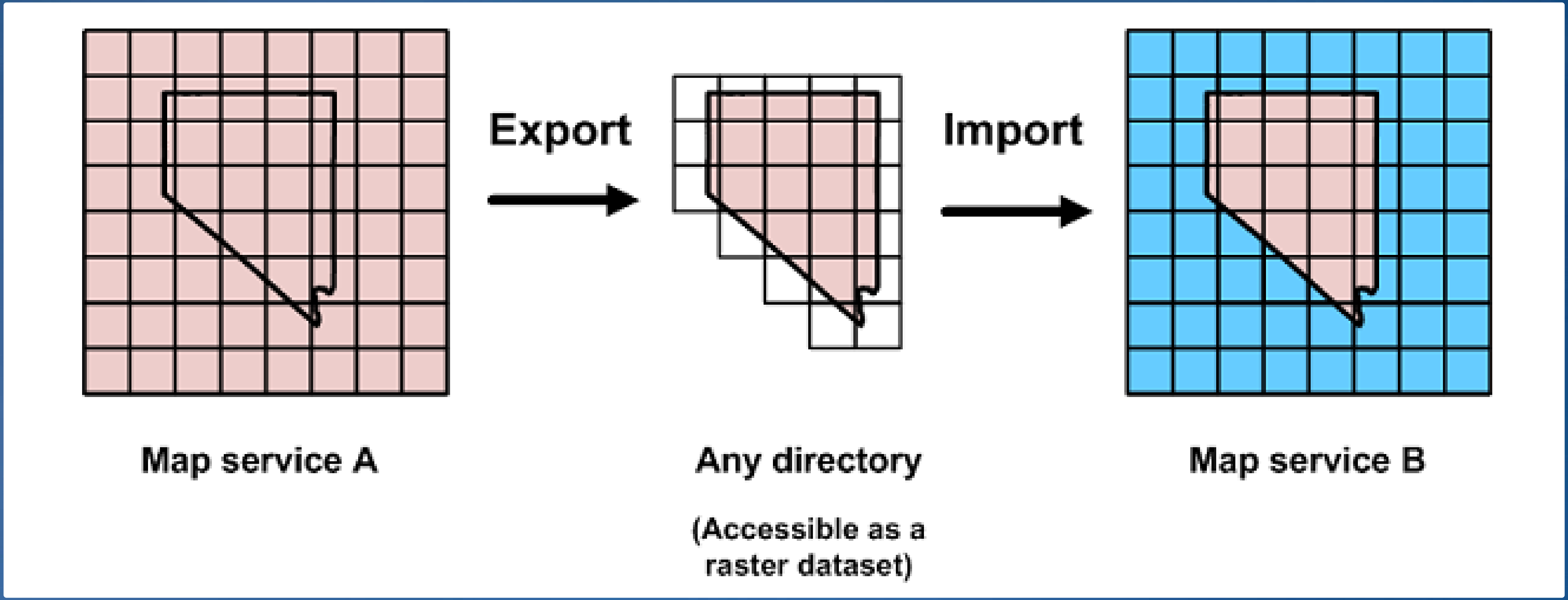
JPEG	JPEG	JPEG	JPEG	JPEG	JPEG	JPEG	JPEG	JPEG
JPEG	JPEG	JPEG	JPEG	JPEG	JPEG	JPEG	JPEG	JPEG
JPEG	JPEG	PNG	PNG	PNG	PNG	PNG	JPEG	JPEG
JPEG	JPEG	PNG	JPEG	JPEG	JPEG	PNG	JPEG	JPEG
JPEG	JPEG	PNG	JPEG	JPEG	JPEG	PNG	JPEG	JPEG
JPEG	JPEG	PNG	PNG	PNG	PNG	PNG	JPEG	JPEG
JPEG	JPEG	JPEG	JPEG	JPEG	JPEG	JPEG	JPEG	JPEG
JPEG	JPEG	JPEG	JPEG	JPEG	JPEG	JPEG	JPEG	JPEG

Compact cache storage format

- Stores tiles in compact, continuous file streams (“bundles”)
  - Maximum ~16,000 tiles per bundle
- Faster copying
- Smaller size on disk
- Generally faster to create
- Less fragmentation

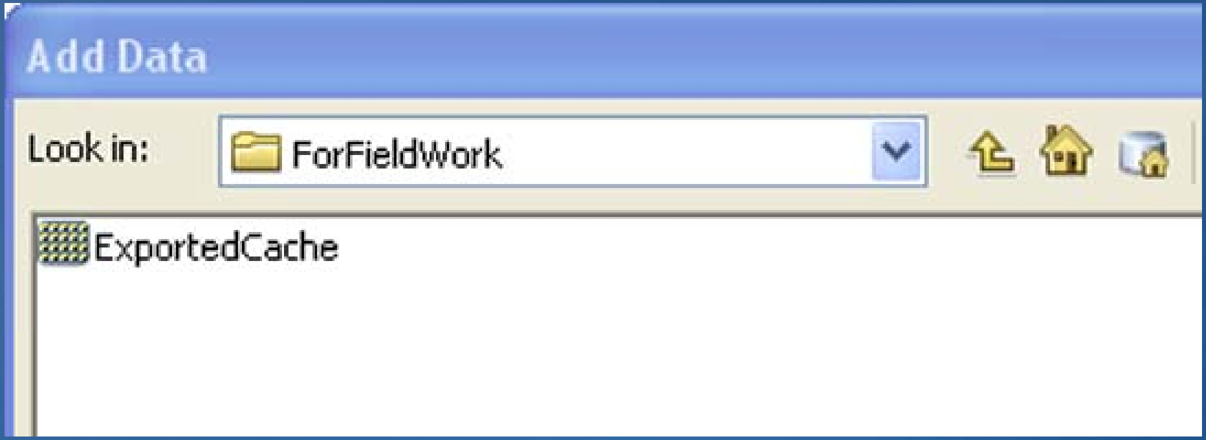
Move spatial subsets of tiles between caches

- New export and import tools allow collaborative and “best available” caching
- Demo



Caches treated as a raster dataset in ArcGIS

- Add Data and browse to cache directory.
  - Looks like any other raster
- Export caches for disconnected field work
  - Export to compact format recommended



Look in: ForFieldWork

ExportedCache

Summary

- Why and what to cache
- Key components of a map cache
- Authoring considerations for cached map services
- Caching strategies
- System architecture