Federal GIS Conference
February 9–10, 2015 | Washington, DC

Designing and Using Cached Map Services

Frank Xia

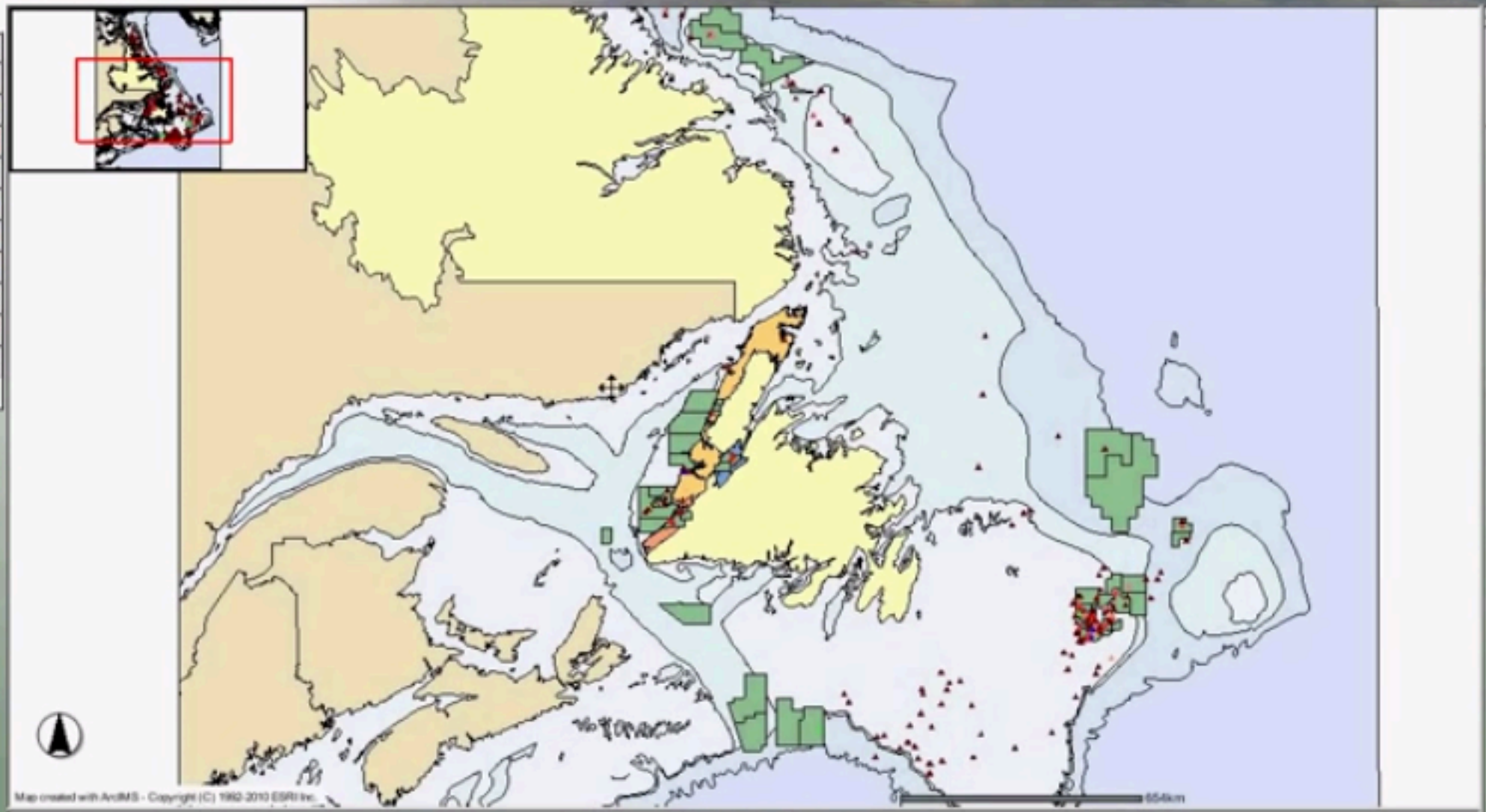
What's covered in this session

- Agenda
  - Why cache maps?
  - Map cache best practices
  - Map cache administration
  - Caching in the cloud
  - 10.3 Performance Improvements

Why Cache Maps

- Understanding caching concepts

Newfoundland & Labrador
Oil and Gas Online Data Source



OffshoreWells is now the Active Layer

Pan

DEMO

- Why cache maps?
  What should you cache?
  How do you make a map cache?

Choosing image formats

- Let the ArcGIS Service Editor choose for you!
  - Vector only caches (few colors)
    - PNG (auto selects bit depth)
  - Vectors only caches (many colors)
    - PNG (auto selects bit depth)
  - Imagery
    - MIXED with 55 quality
  - Vectors or labels + Imagery
    - MIXED with 90 quality

Does Compression really make a difference?

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

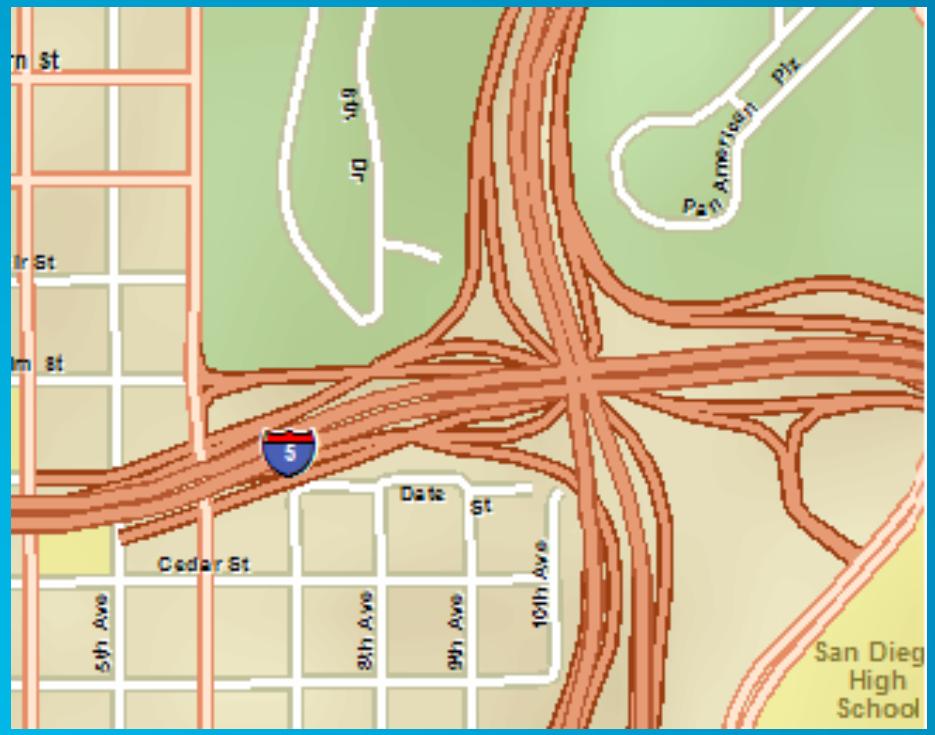
Which one looks better?

JPEG 96 – 25KB

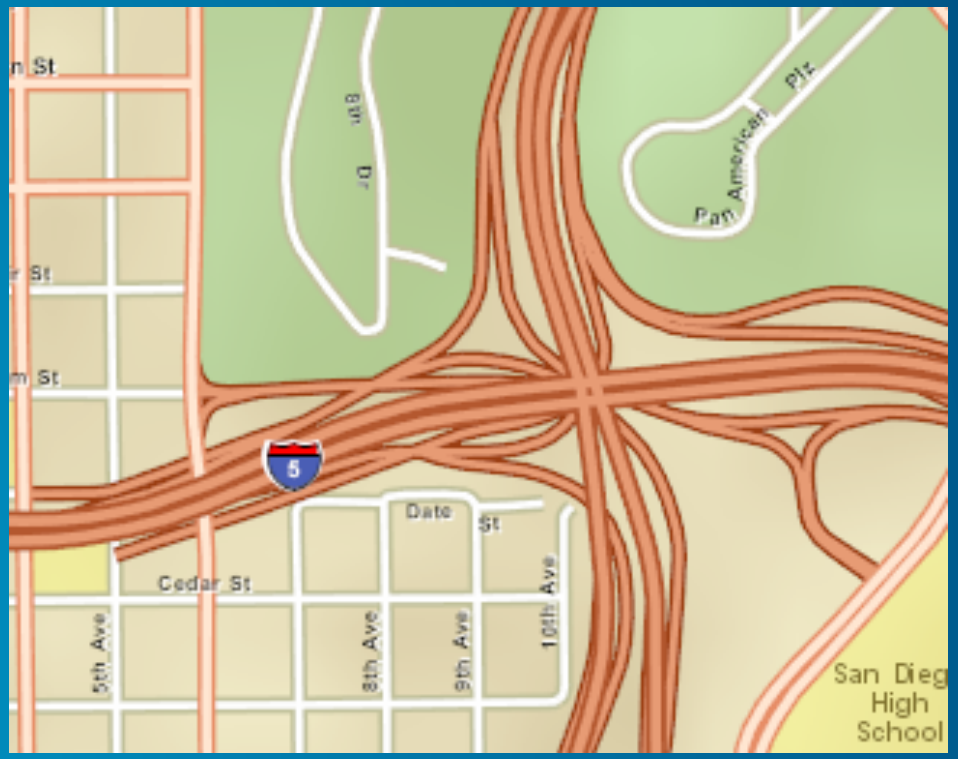
JPEG 96 – 30KB

Does antialiasing make a difference?

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



No antialiasing



Best antialiasing

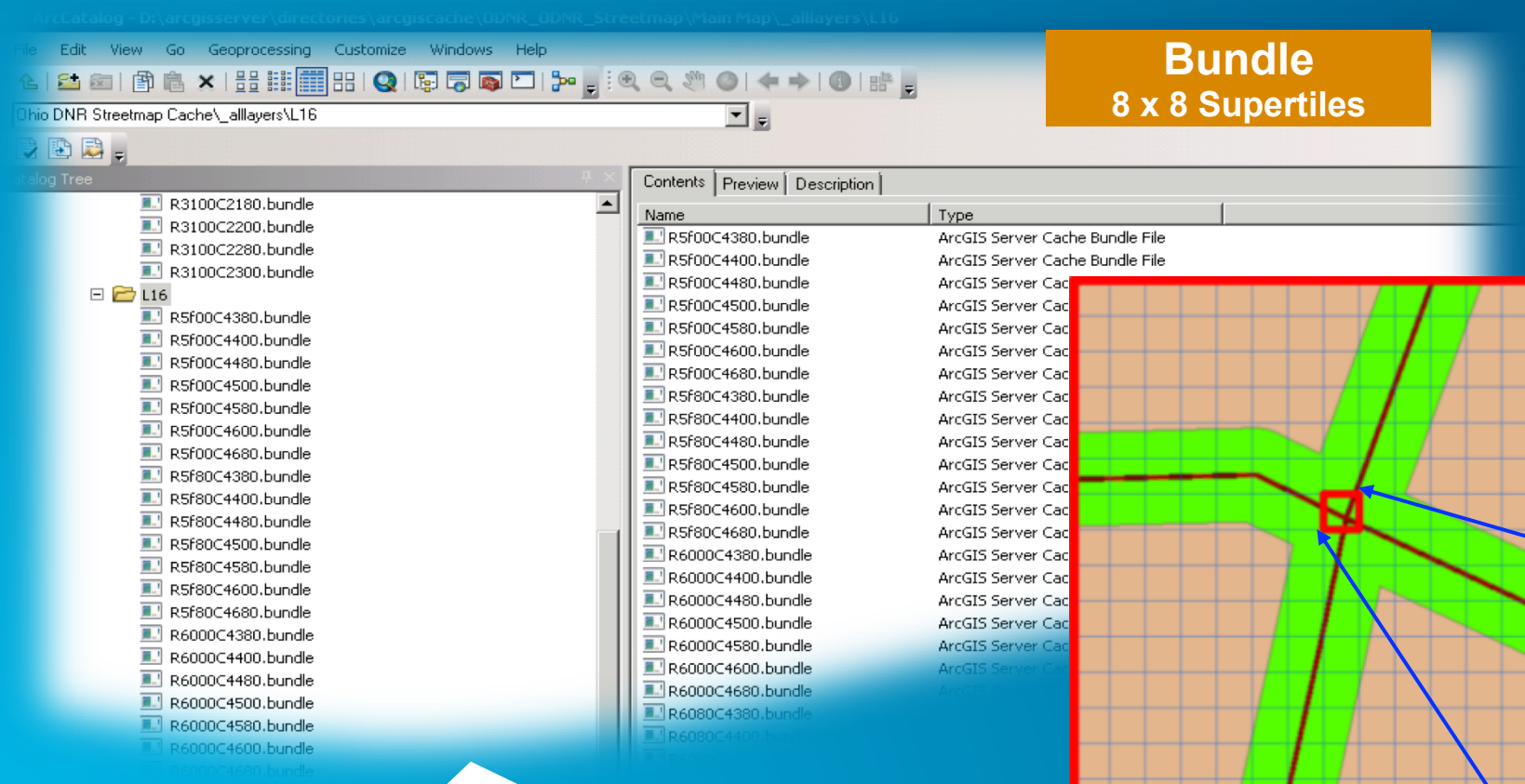
Demo Publish

- With ArcMap

What you should cache?

- Base maps
- Data don’t change very often
- Small scale maps

Understanding cache structure



Bundle
8 x 8 Supertiles

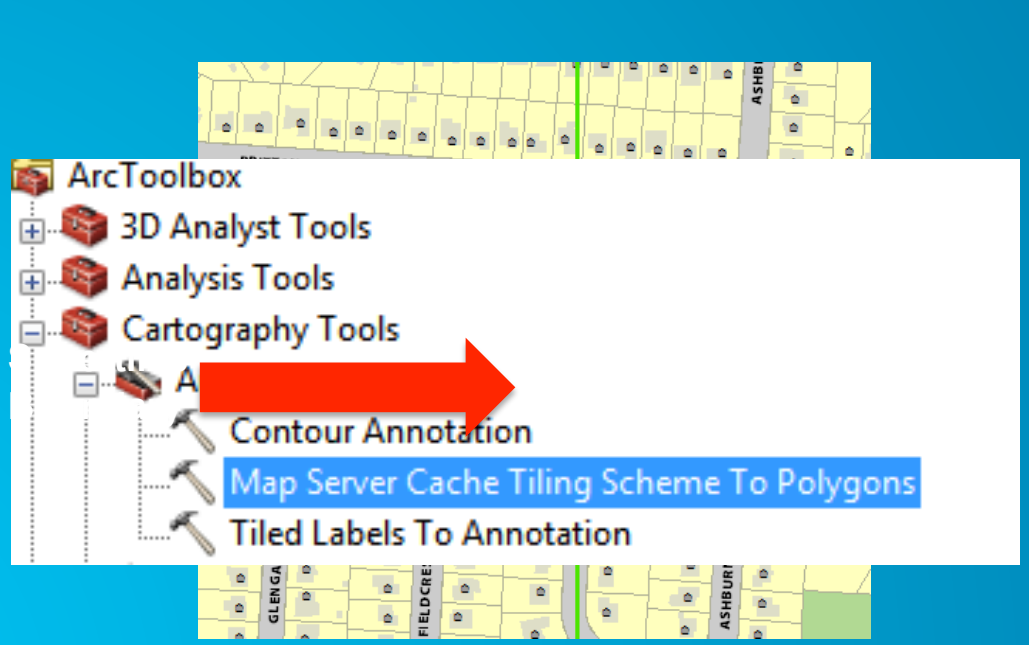
Supertile
16 x 16 tiles

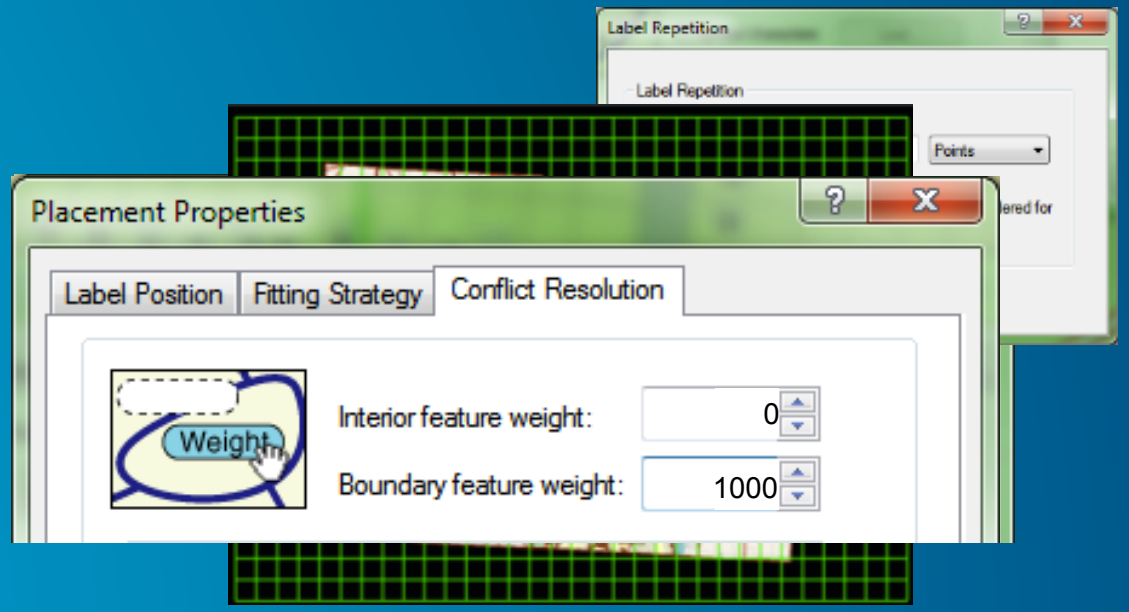
Standard tile
256 x 256 pixels

16, 384 tiles

Supertiles and Labeling

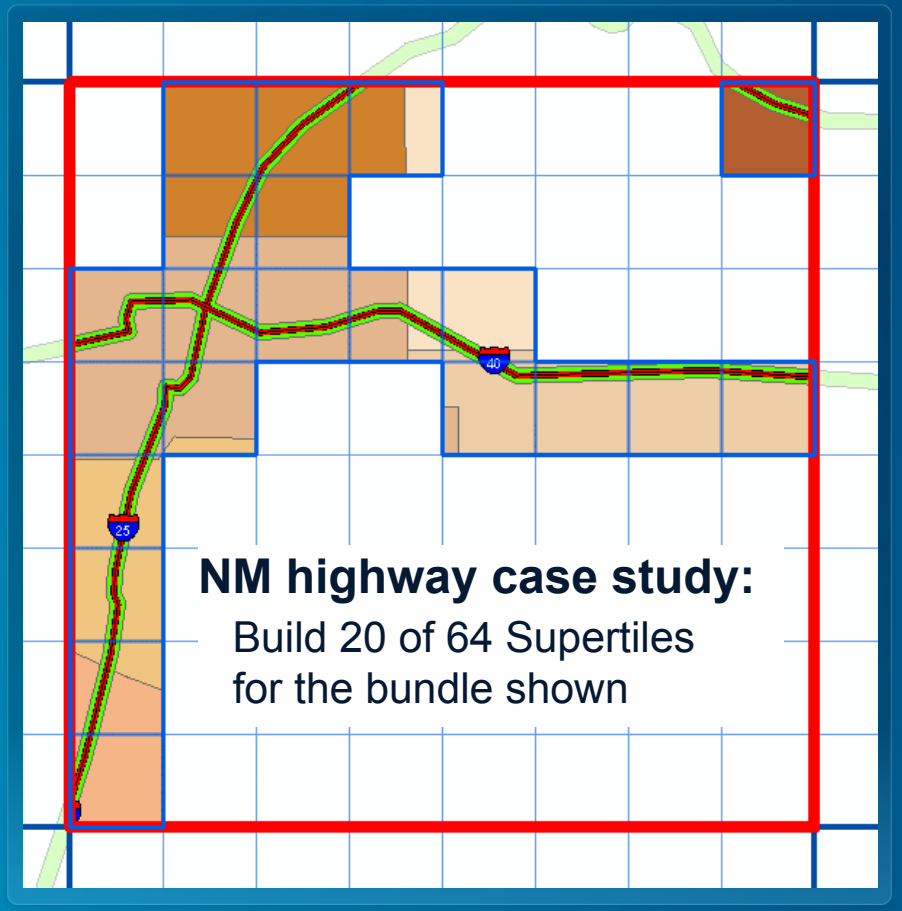
- ArcGIS Server Draws Large Areas
  - Reduces duplicate labels
- Duplication May Occur
  - Use Annotation or MapPlex Labels with Rules
  - Use Map Server Cache Tiling Scheme To Polygons





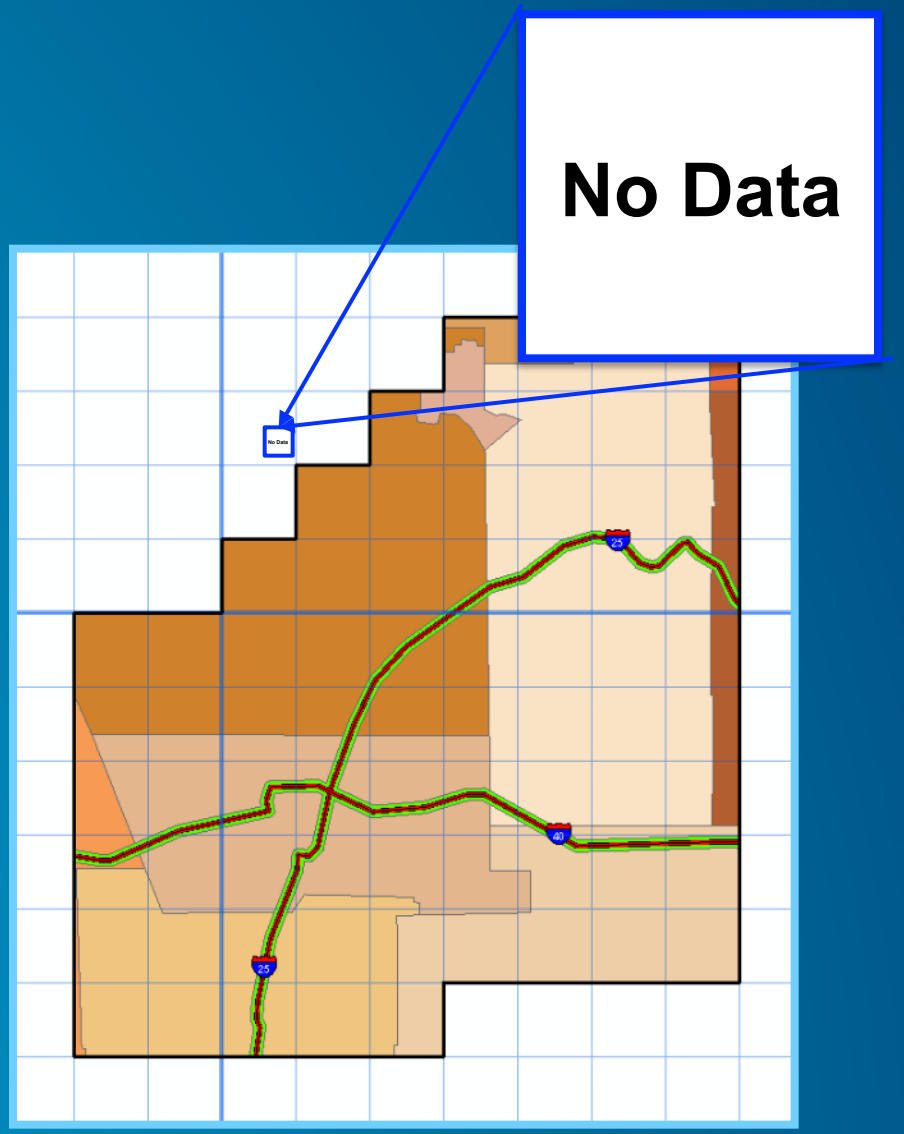
You don't need to generate everything

- Cache by feature
  - Polygon features
  - Generates all tiles for intersecting supertitles
- Saves on…
  - Generation time
  - Processor resource
  - Disk usage



Handling tiles you do NOT create

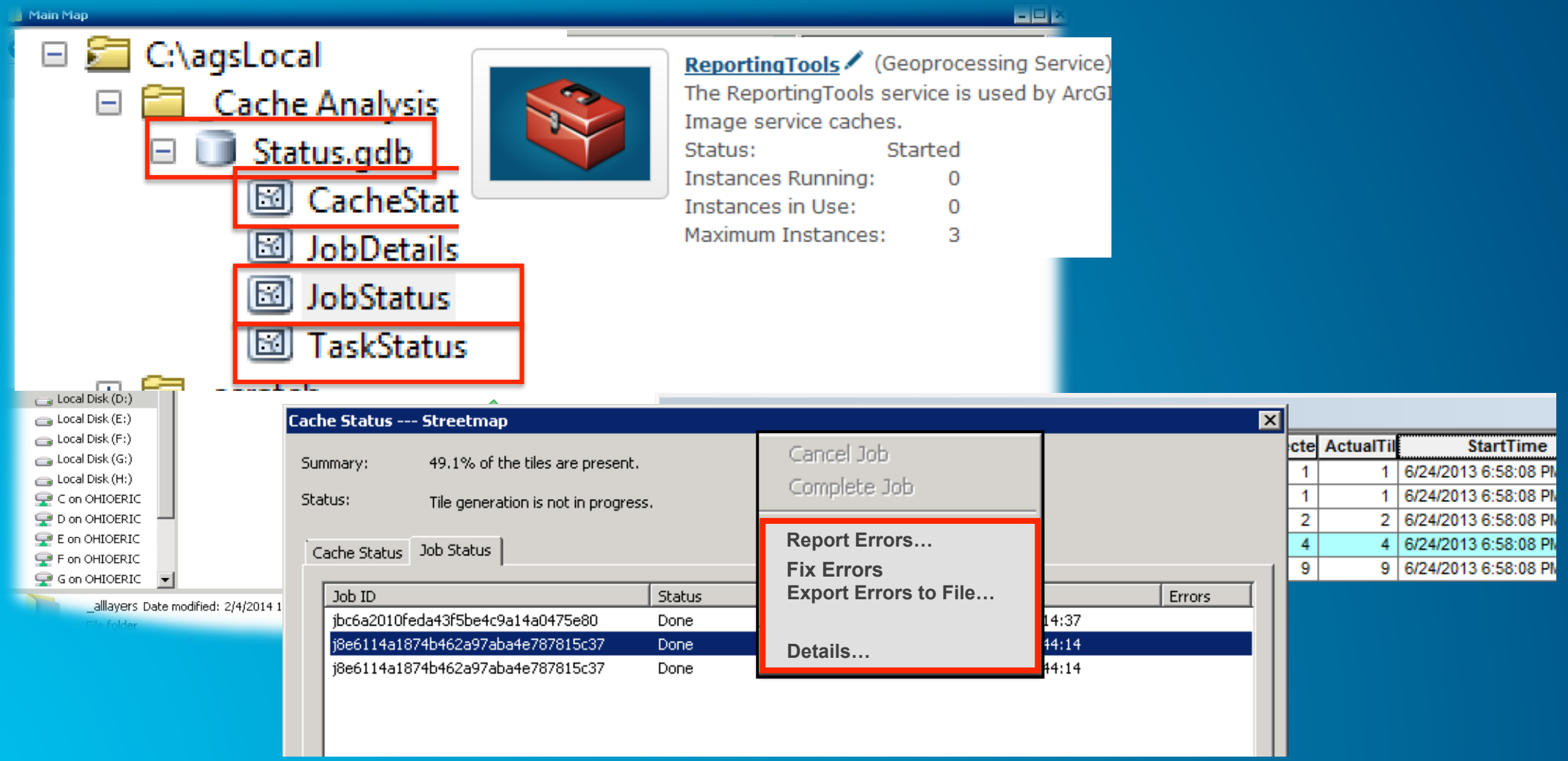
- Create “No Data” tile
  - Same image format (JPG or PNG)
  - Same size (256 x 256)
  - Save in cache folder
    …\<dataframe>_alllayers
- How to
  - Knowledge base article 36939 has sample files



Build a test cache and note the following

- Creation time
- Appearance
- Client performance
- Cache size validation

Tracking cache status & fixing errors

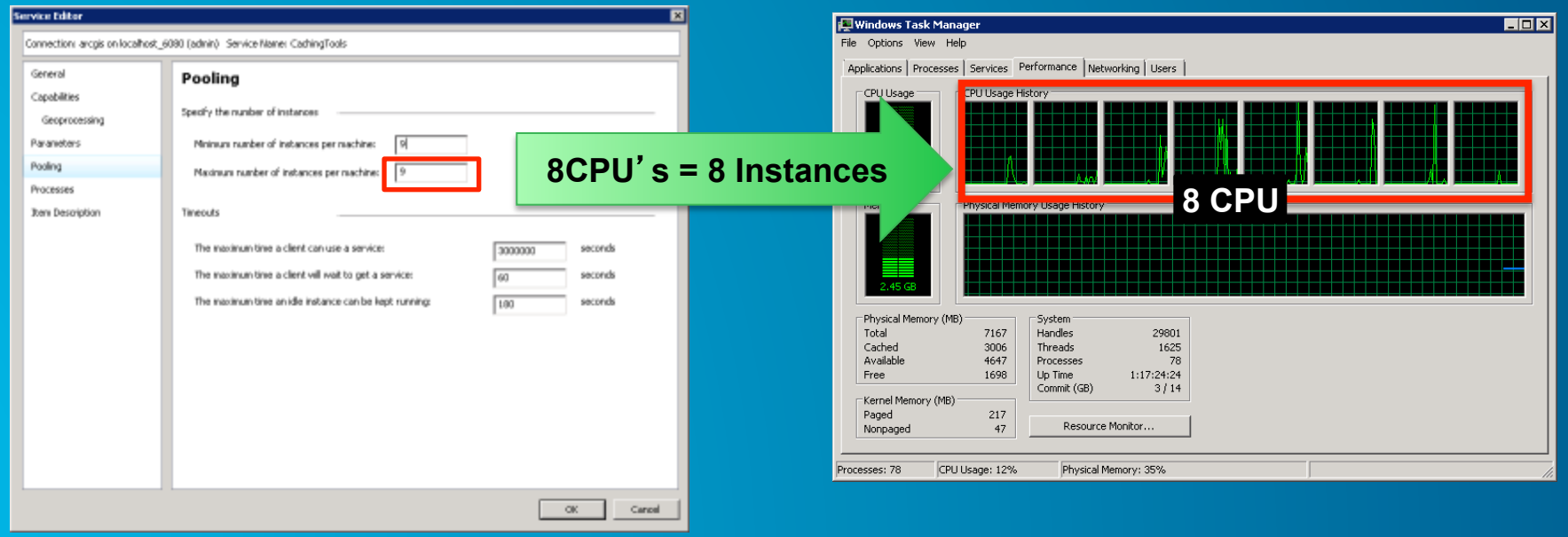


Map cache administration

- 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

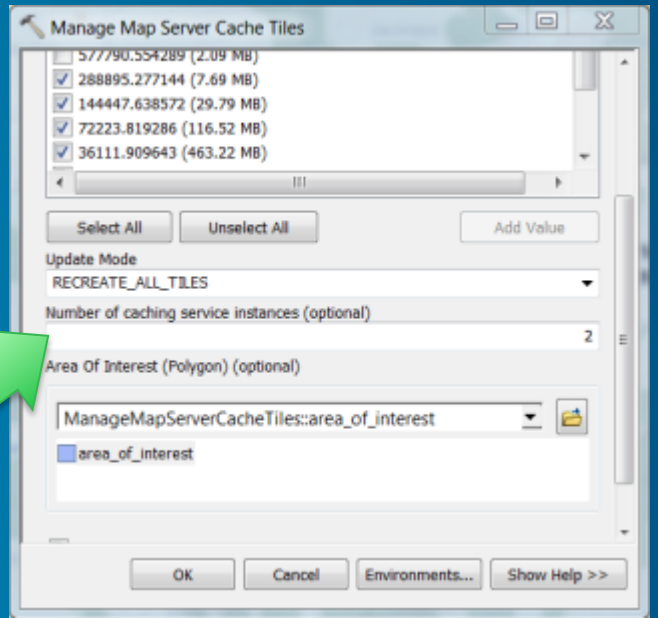


System caching services

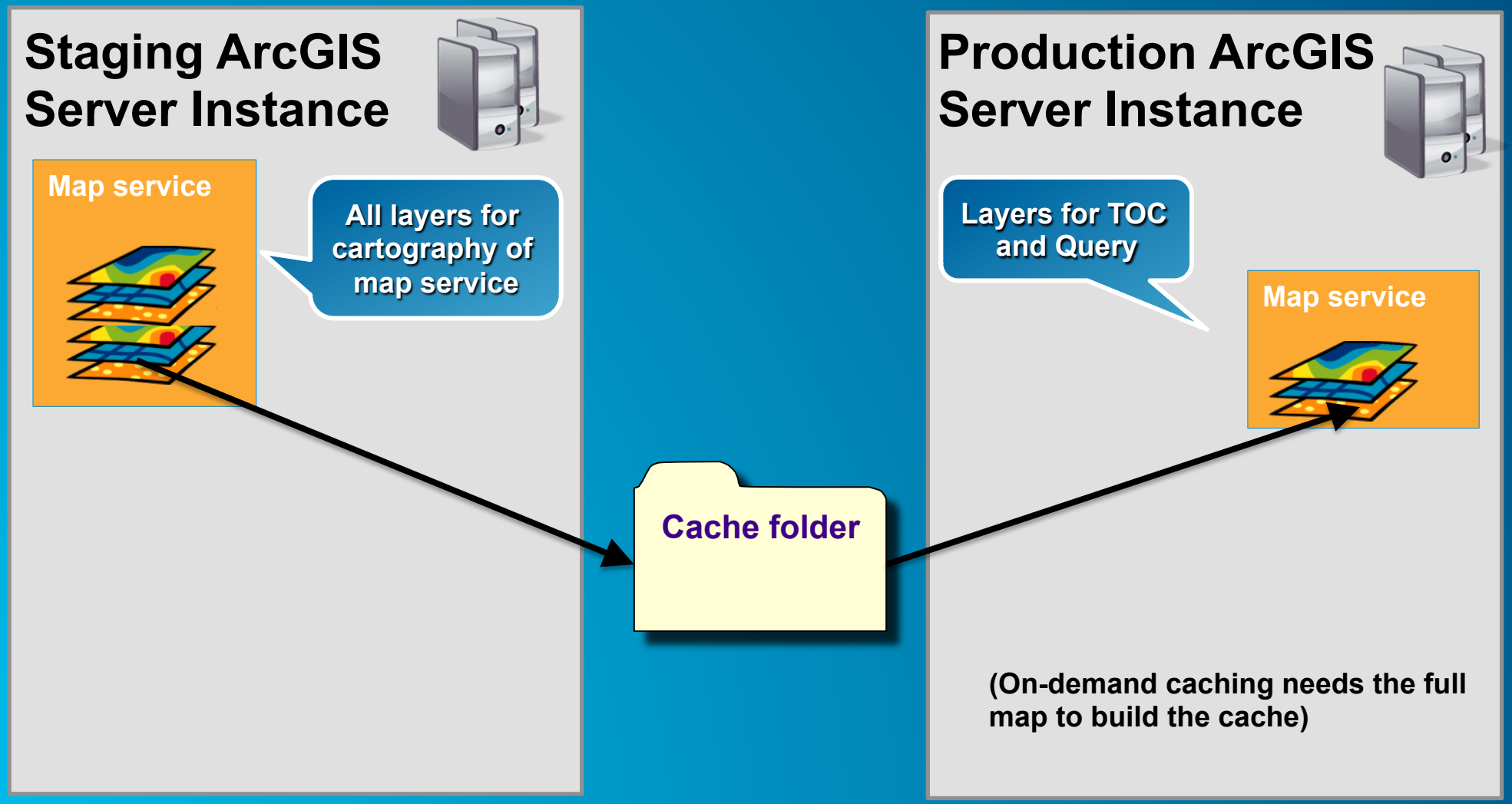
- System services
  - Caching Tools: Sets caching instance per machine
  - Caching Controllers: Assign cache jobs to instances
- Manage Map server Cache Tiles
  - Controls instances per job
  - Set to -1 to use all instances

Per machine

Per job

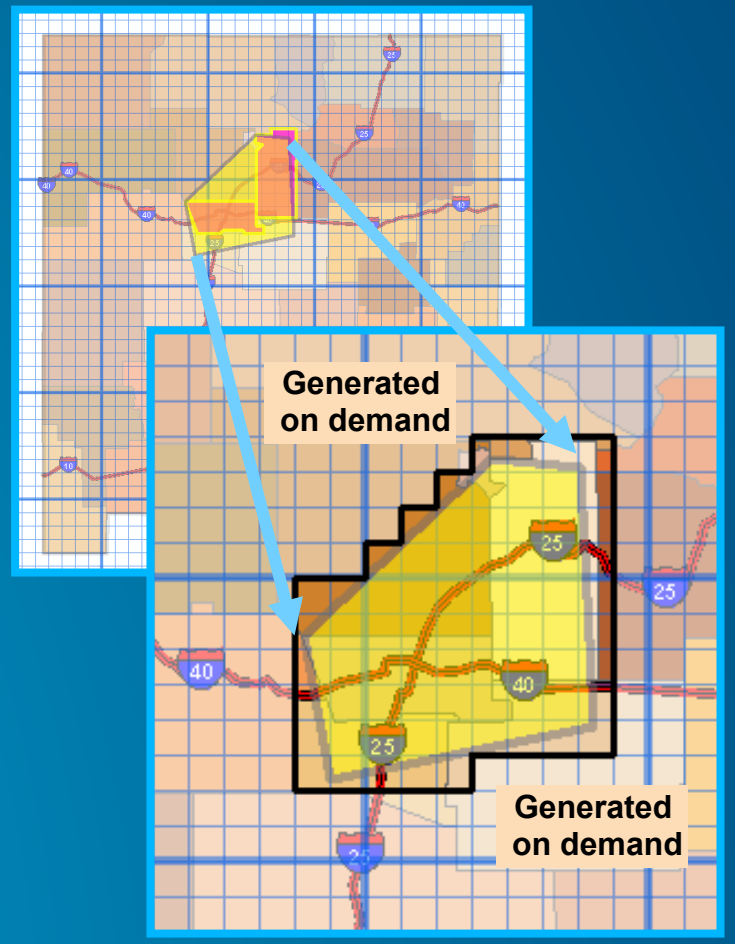


Update a cache using a staging server



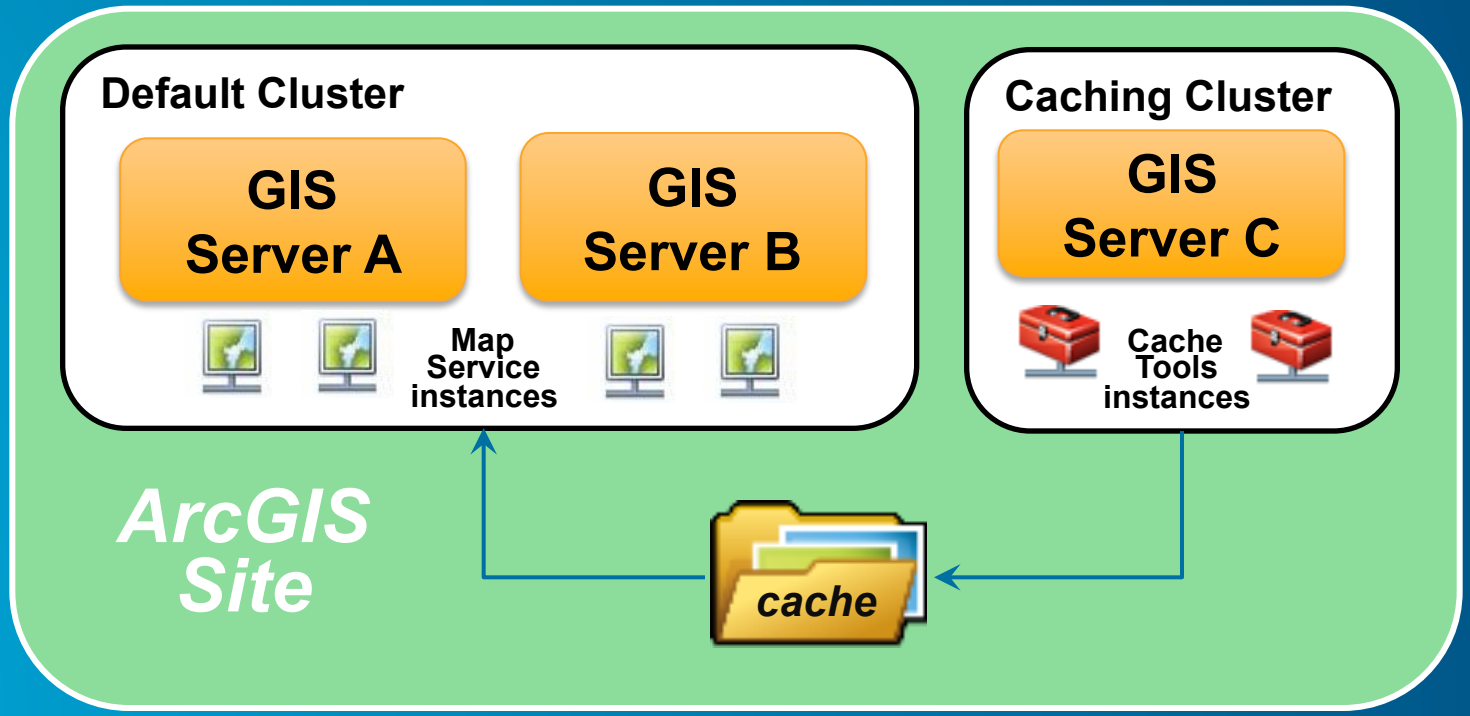
Pre-create coupled with cache on demand

- Pre-create high use areas
  - Population centers
  - Parks, roads, attractions
- Features
  - Cover popular extent
  - Generate key tiles
  - All others generated on demand



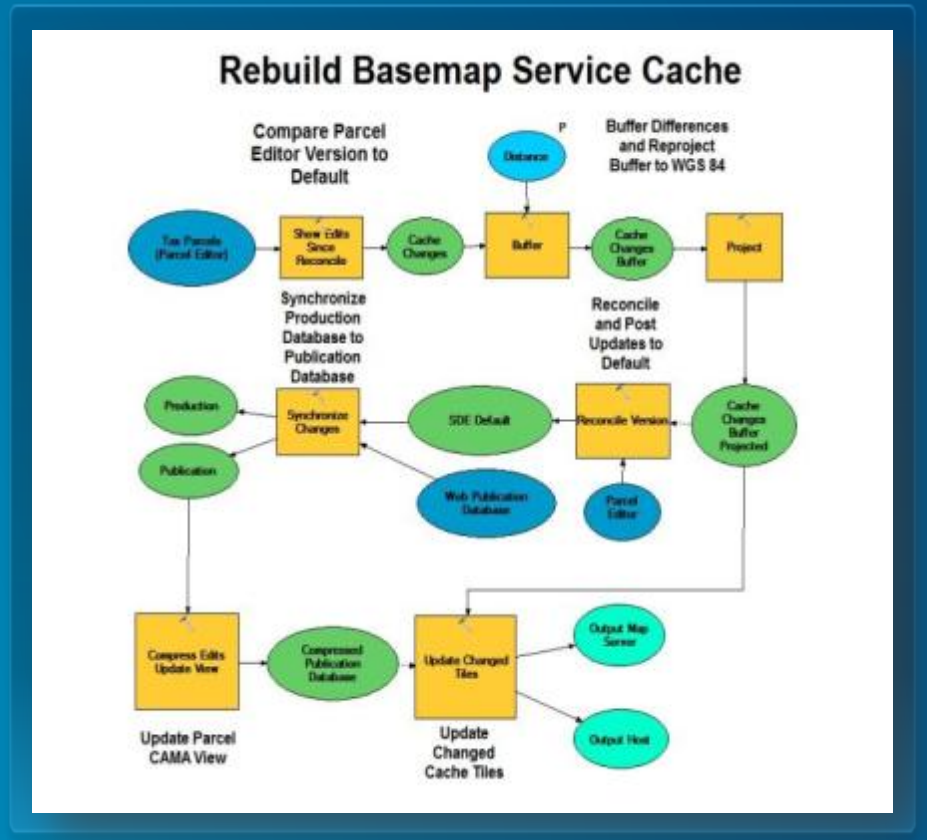
Isolate caching to certain servers

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



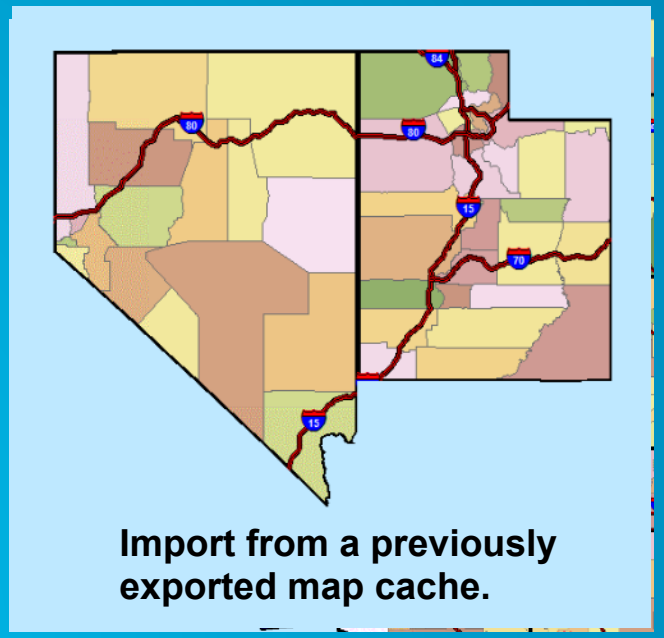
Cache update automation

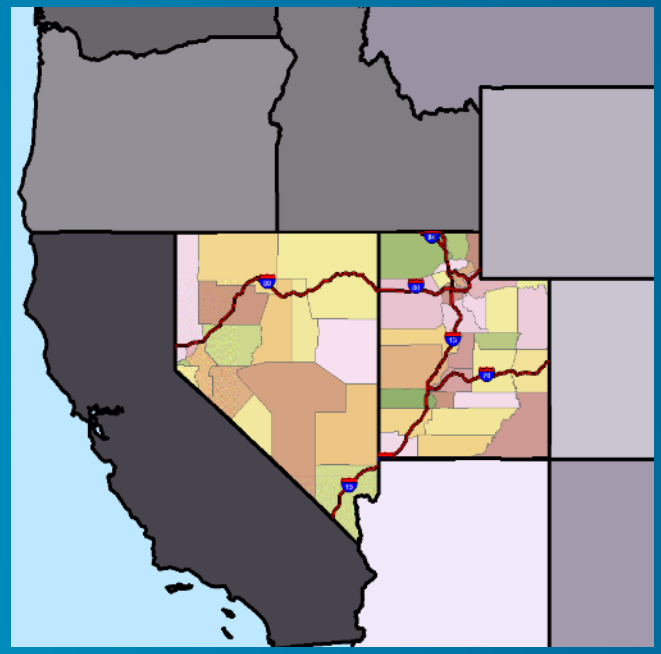
- Use Model Builder to script update automation
  - Rebuild Specific Tiles
  - Export to Python
  - Schedule Run Time
- Useful update tools
  - Compare feature classes
  - Show edits since reconcile
- Sample available on GitHub
  - https://github.com/Erodenberg/UpdateTileCache



Cache export & import tools

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





Caching in the Cloud

- ArcGIS Online Map Caching

Caching in ArcGIS Online

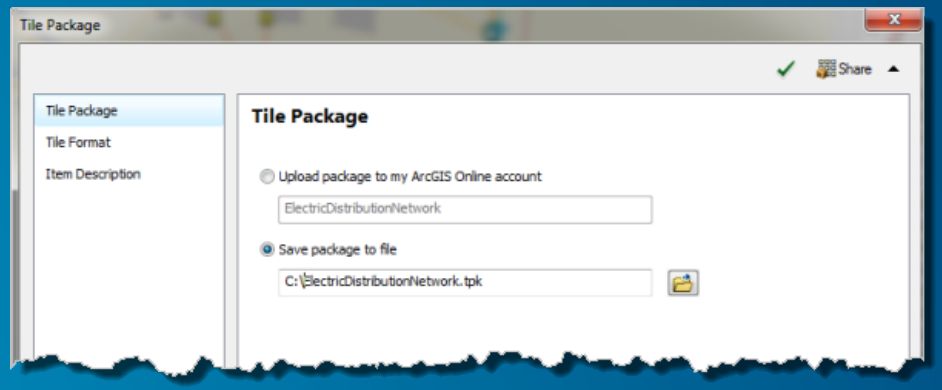
- ArcGIS Online subscription allows for caching
- No need to worry about capacity
- Charged by tile creation and storage
- Two approaches
  - Upload data to AGOL
    - Build and store cache with AGOL
  - Upload tile package to AGOL
    - Build cache on premises (ArcMap) but store with AGOL
- Understanding credit usage: http://www.esri.com/software/arcgis/arcgisonline/credits

Why create a tile package?

- Local cache for Desktop, Runtime and productivity applications (Collector)
- Transport a map cache
- Upload a map cache to ArcGIS Online

Creating a tile package

- ArcMap Options > Sharing > Enable ArcGIS Runtime tools
- Three options for creation
  - Create tile package within ArcMap
    - Single processor
    - File > Share As > Tile Package
  - Create Cache with ArcMap & Geoprocessing
    - Parallel Processing
    - Data Management > Tile Cache Toolset
      - Manage Tile Cache
      - Export Tile Cache
  - Create cache with ArcGIS Server
    - Tile Cache > Export Tile Cache
      - Uses Parallel Processing Factor Geoprocessing Environment setting



Cache Consumption Performance Improvement in 10.3

In ArcGIS 10.3 Server

- Reengineered the compact cache storage format
  - Provides faster performance when consuming cached map/ image services
  - Reduces the number of files created in the cache directory.
- These performance optimizations are supported with
  - New caches created using 10.3
  - Existing (compact/exploded) caches that are upgraded using the “Upgrade Map Server Cache Storage format” tool
- Existing caches will continue to work in 10.3
- Cache format change does not impact clients compatibility to consume cache served out by the Server.

Faster Performance when consuming caches

- 45% better than 10.2.2 over UNC
- 40% better than 10.2.2 on local drive
- Secured services with GIS tier security at par with anonymous services.
  10 times better than 10.2.2
- WMTS is more than 5 times better than 10.2.2

Should I upgrade my existing cache?

- If your caches meet your existing demands, No.
- Benefits only for very large caches. Consider the trade-offs.
  - Upgrade takes 20% of the time taken to generate the cache.
  - QA/QC your new caches/subject to failures

Note:

- Performance improvements of secured cache services & WMTS are independent of version of compact storage format

Questions?

Example Credit Usage

- ArcGIS Online Tile storage = 1.2 credits per 1 GB per month
  - 1 Credit is 10 cents or less
- This cache = 100 MB
- Credits per month
  - 100 MB / 1024 MB * 1.2 Credits = 0.117
  - 0.117 Credits * 10 = 1.17 Cents
- 1.17 cents per month * 12 months = 14.06 cents per year

- … In 20 years this cache will cost you less than 3 dollars

Understanding our world.