



Map Caching in ArcGIS for Server

Sterling Quinn

Tom Brenneman

Schedule

- Planning and building a cache
- Architectures for caching
- Caching in the cloud

Please!
Turn **OFF** cell phones
and paging devices



- We will hold periodic breaks for questions

Demo of caching with ArcGIS

- **CachingTools geoprocessing service**
- **Defining cache properties**
 - **Build automatically or manually**
 - **Min and max cached scales**
 - **Image format**
- **Calculating cache size**
- **Asynchronous caching**
- **Real-time cache status updates**
- **Viewing job status and fixing problems**
- **Visualizing cache creation over time**

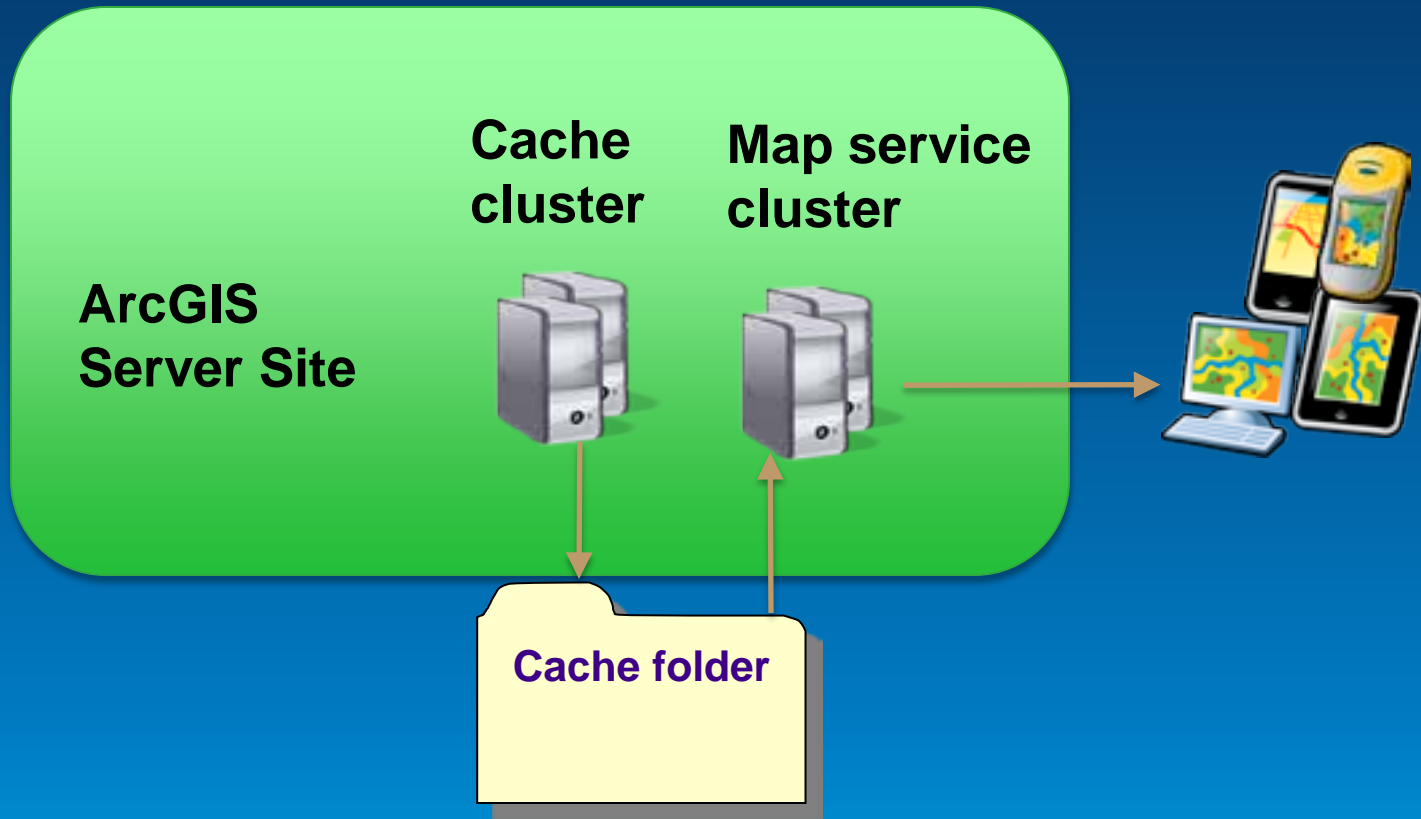
Short break for questions



Architecture for caching and cache updates



Clusters allow you to isolate cache creation



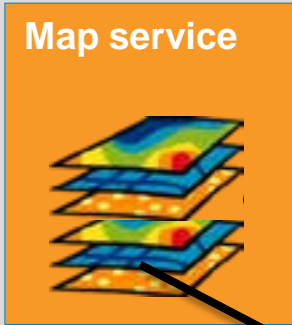
Cache cluster can scale out while the cache job is running

Update a cache using a staging server

Staging ArcGIS Server Instance



Map service



All layers for cartography of map service

Production ArcGIS Server Instance



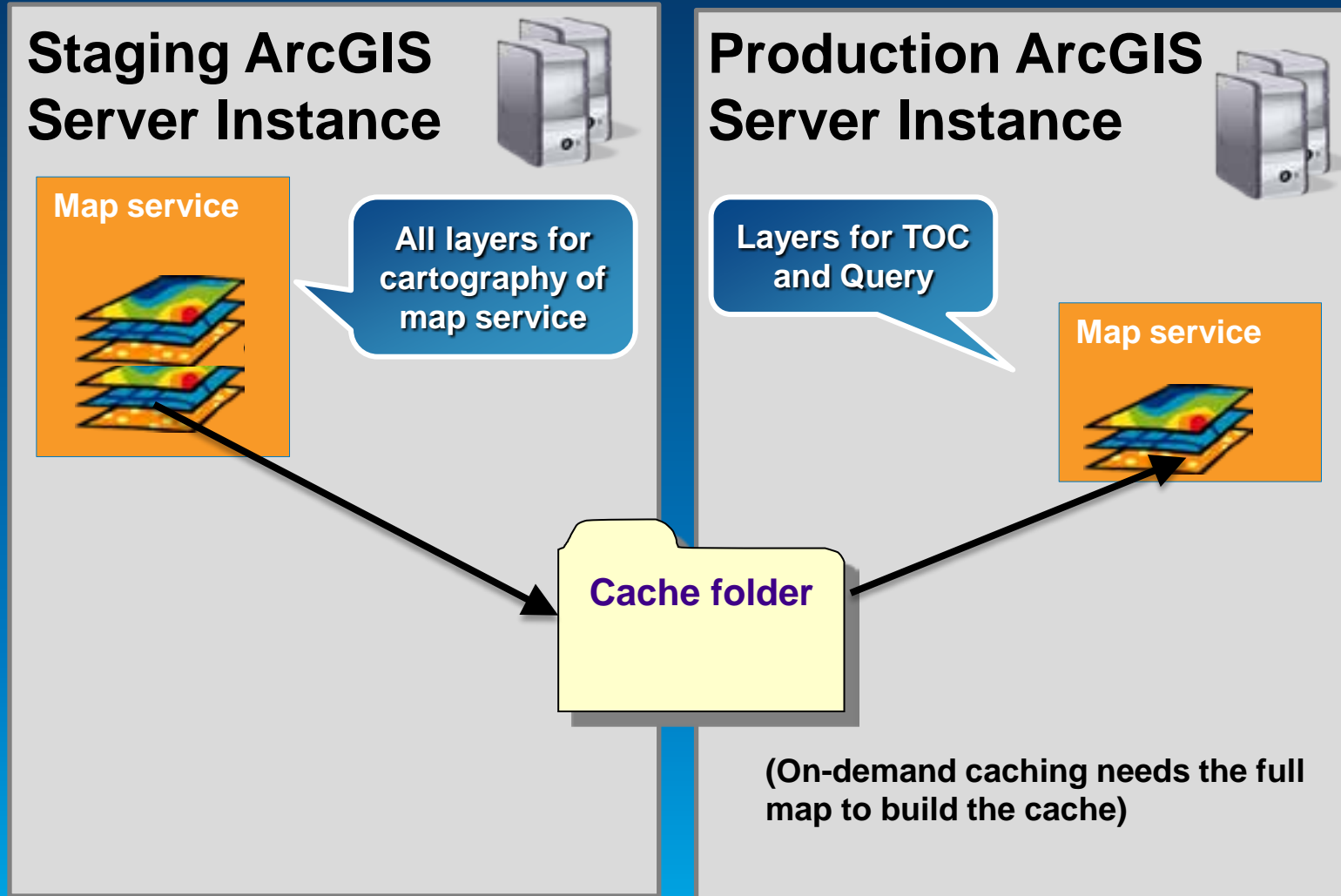
Layers for TOC and Query

Map service



Cache folder

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



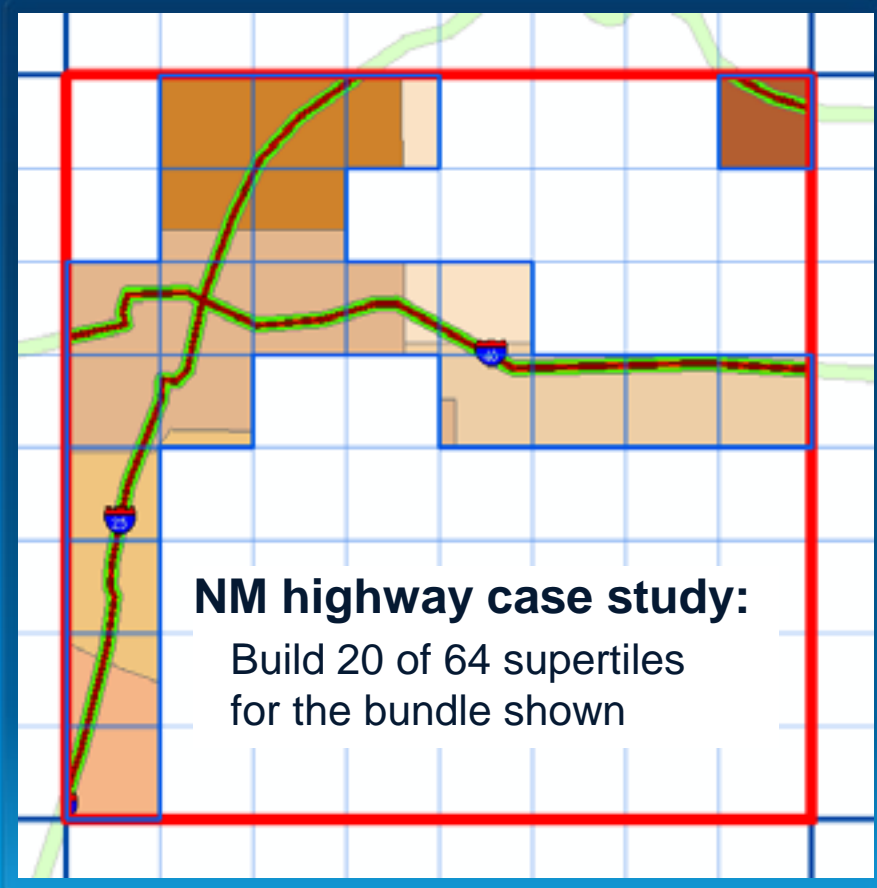
Cache update strategies

- Rebuild the entire cache
 - Size of cache
 - Time to cache
- Rebuild specific tiles
 - Rebuild at specific scales
 - Rebuild areas based on change detection



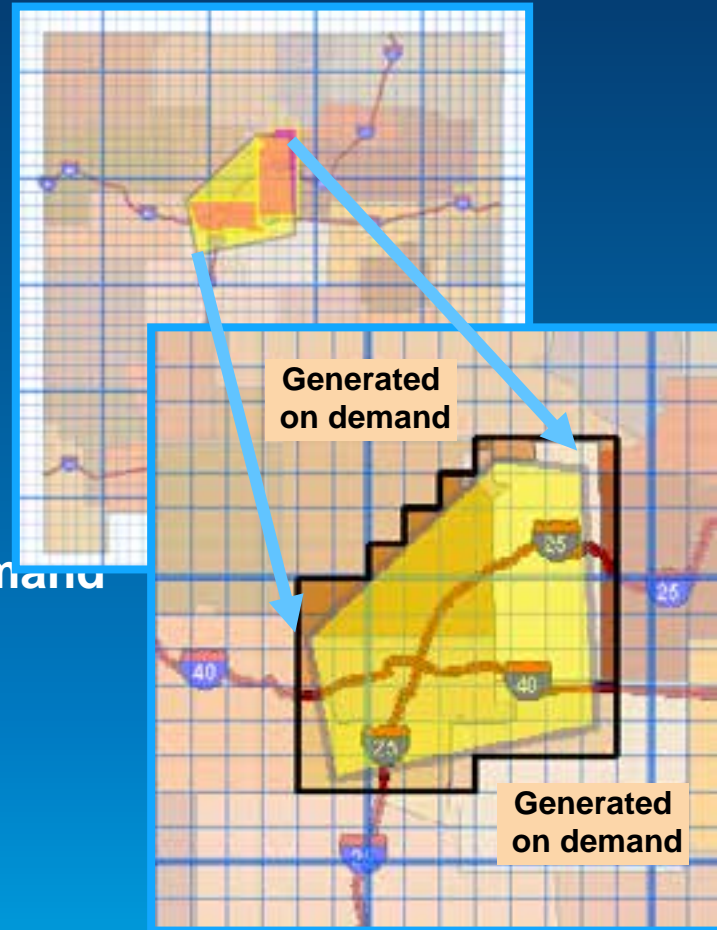
How much of a cache should you generate?

- Cache by feature
 - Geographic elements
 - Generates tiles for intersecting supertiles
- Saves on...
 - Generation time
 - Processor resource
 - Disk usage



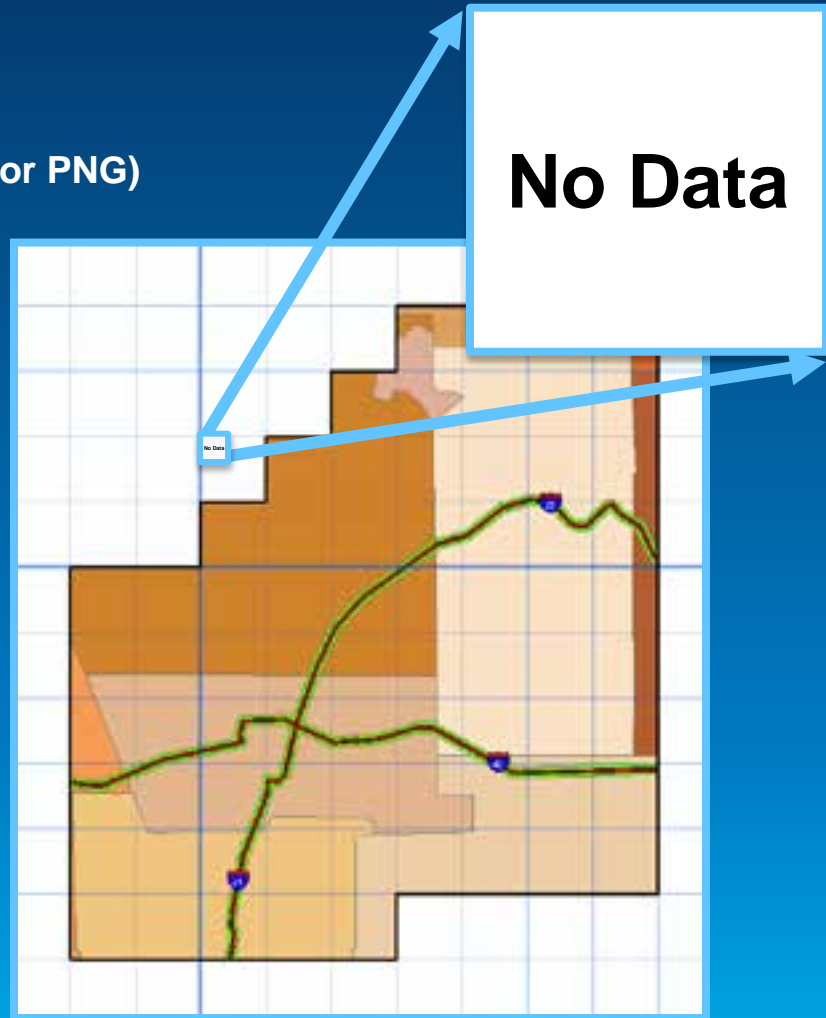
Pre-create coupled with cache on demand

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



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
- Knowledge base article [36939](#) has sample files



Other caching techniques

- Image Services
- ArcGIS Online
- Tile Packages
- Multiple tile servers



What is image service caching

- **Fast access to images as a tiled service**
 - Out performs / scales mosaic dataset and raster dataset
 - Imagery is not processed on the fly
- **Uses image extension**



Image resolution and cache scales

- ArcGIS Server chooses the optimal scales for Imagery
 - ArcGIS will not exceed raster resolution
- Neither should you!
 - ArcGIS Server resamples imagery exceeding raster resolution
 - ArcGIS Desktop zoom to raster resolution
- Scale based on 96 DPI

$$\text{Scale (Ft)} = (x/12) * 96$$

$$\text{Scale (m)} = (x/0.0254) * 96$$

Cell Size(Ft)	Scale 1:X	Cell Size (m)	Scale 1:X
.25	288	0.3	1133.858
.5	576	0.6	2267.717
1	1152	1	3779.528
3	3456	3	11338.58
15	17280	15	56692.91

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



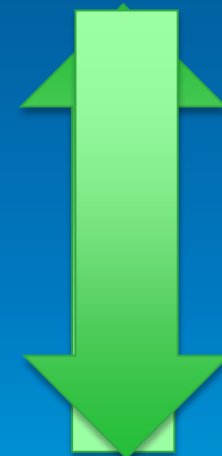
JPEG 96 – 30KB

Why should I cache image services

- Improved performance for basic images
 - Can not modify mosaic methods
- Skip overview generation
 - Tiles generate from large scales to small scales
- Improve performance for slow formats
 - Recommended for highly compressed formats – e.g. JPEG2000, MrSID
- Caching image services is much faster than caching map services with imagery
 - Faster rendering engine for imagery

Image Services

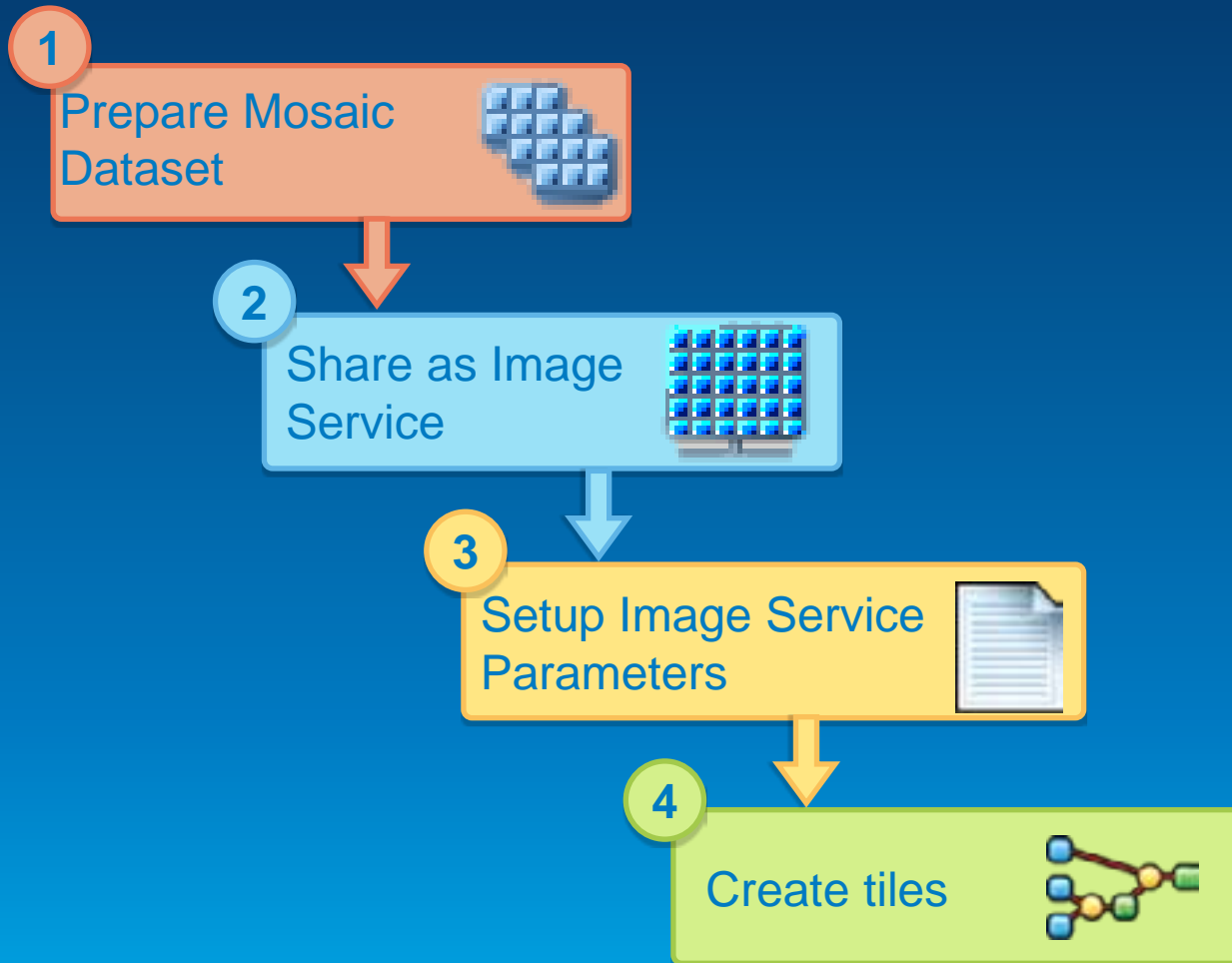
Small Scales



Large Scales



Building an image service cache



Switching to a Dynamic Request

- Supported by Map and Image Services & the Web API's
- Force a Cached Service to Render Dynamic
 - Use `ArcGISDynamicMapServiceLayer(Cached Service URL)`
- Render cached tiles from smaller scales levels 0 – 15
- Render dynamic services at larger scale levels 16 – 19
 - Large scales = less area to draw features = faster

Scale Levels 10 – 17 Cached

Scale Levels 18 – 19 Forced Dynamic Draw



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 premise but store with AGOL
- Understanding credit usage:
<http://www.esri.com/software/arcgis/arcgisonline/credits>

Why create a tile package?

- **Local cache for Desktop and Runtime applications**
- **Transport a map cache**
- **Upload a map cache to ArcGIS Online**

Creating a tile package

- **ArcMap Options > Sharing > Enable ArcGIS Runtime tools**
- **Two options for creation**
 - **Create tile package within ArcMap**
 - **Single processor**
 - **File > Share As > Tile Package**
 - **Create cache with server**
 - **Tile Cache > Export Tile Cache**
 - **Uses Parallel Processing Factor Geoprocessing Environment setting**

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**

Thank You

- **Please fill out session survey**
 - <http://bit.ly/cache2013>
- **Offering ID: 262**
- **Questions?**



Understanding our world.