ArcGIS API for JavaScript:
Creating Custom Layers and Layer Views

Yann Cabon, Johannes Schmid

Agenda

- **Introduction to layers in 4.x**
- **Building a custom dynamic layer**
- **Building a custom tile layer**
- **Layer loading and attribution**
- **Custom Elevation Layer**
- **Using web workers**
- **Future work**

Introduction to Layers

Yann Cabon

Introduction to Layers extensibility in 4.x

- They don't have a visual representation, they are a data access

- Create a custom layer
  - To connect to a service not supported (or not yet) by the API
  - To work on the data client-side before it's being displayed
  - Because you can ☺

- We will cover:
  - Dynamic layer
  - Tile
  - Elevation

- 4.4 with introduce new layer classes designed for extensibility

Custom Dynamic Layers

Yann Cabon

Dynamic Layer

- displays an image that covers the view. At the end of a user interaction, a new image is requested.
  - 1 export in MapView
  - 2 exports in SceneView
- Pro: One export on the service
- Con: Export is different each time, so cannot be cached in the browser cache

Dynamic Layer

- **Extensibility hooks:**
  - **BaseDynamicLayer.getImageUrl**
    - **Easiest method that creates a url for a extent and size**
  - **BaseDynamicLayer.fetchImage**
    - **Method that do the actual image fetching**
    - **To extend if you need to transform data.**
    - **Default implementation fetch the image at the URL returned by getImageUrl**
- **Example:**
  - **WMS Layer from GeoBretagne.fr**
    - *http://geobretagne.fr/mapfishapp/*
    - *http://geobretagne.fr/geonetwork/apps/georchestra/?uuid=048622c5-b333-4c2b-94ec-40a41608dc06*

Custom Tile Layers

Yann Cabon

Tile Layer

- **Displays adjacent images stitched together to covers the view.**
- **New tiles are requested as the user interacts with the view**
- **Pro: really efficient as tiles are cacheable resources on the server and web browser**
- **Con: a tile resource has to less dynamic to have an efficient layer**

Tile Layer

- **Extensibility hooks:**
  - **BaseTileLayer.getTileUrl**
    - **Easiest method that creates a url for a tile id: level / row / col**
  - **BaseTileLayer.fetchImage**
    - **Method that do the actual image fetching**
    - **To extend if you need to transform data.**
    - **Default implementation fetch the image at the URL returned by getTileUrl**
- **Example:**
  - **Tiled WMS Layer from GeoBretagne.fr**
  - **Tint Stamen tiles**

loading and integration

Yann Cabon

Loading and integration

- **Layers are loadable resources**
- **They need to fetch some information from a service before being ready**
- **Is the service exists and valid?**
- **What are the properties of the layer?**
  - **Extent**
  - **Title**
  - **Attribution**
  - **Others…**

Custom Elevation Layers
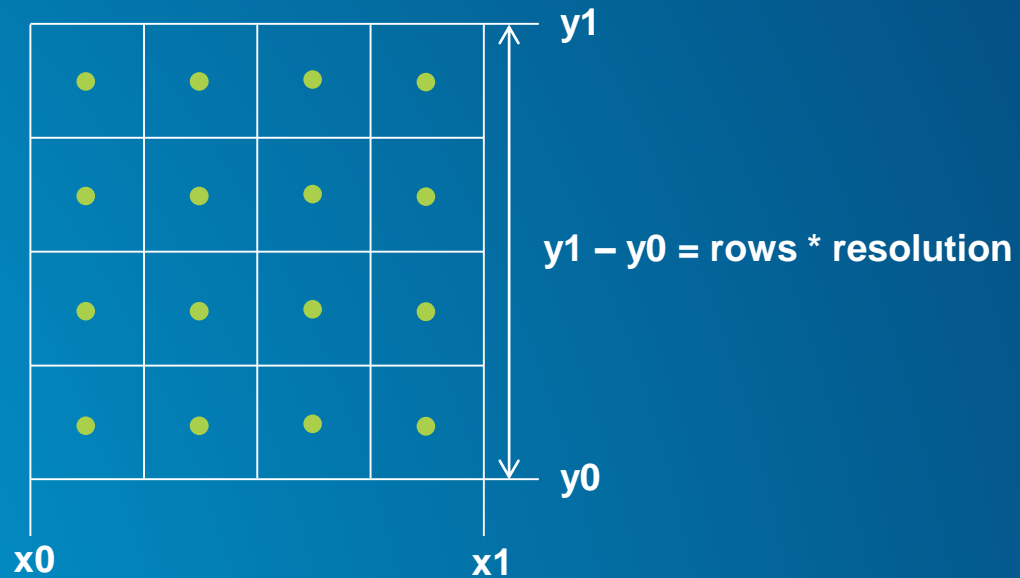
Johannes Schmid

Custom Elevation Layers

- **Derive from BaseElevationLayer and override fetchTile()**
- **Return a Promise that resolves to**

```
{
    values: Array | Float32Array;      // height values in meters
    width: number;                     // this.tileSize[0] + 1 (e.g. 257)
    height: number;                    // this.tileSize[1] + 1
    noDataValue: number;               // Must be equal to options.noDataValue
}
```
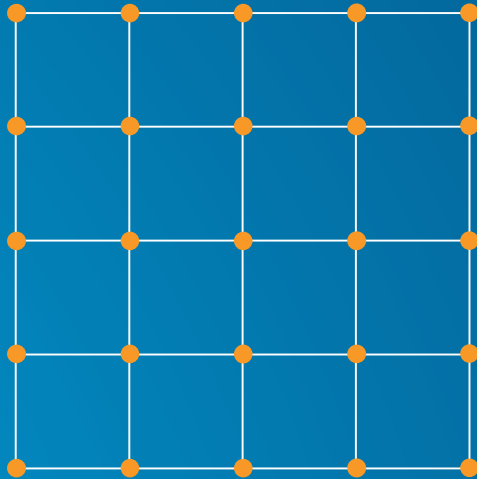
Custom Elevation Layers

- **Map tiles**
  - `[x0, y0, x1, y1] = this.getTileBounds(level, row, col);`

y1

y1 – y0 = rows * resolution

y0

x0   x1

  - **Number of values in one row or column: e.g. 256**
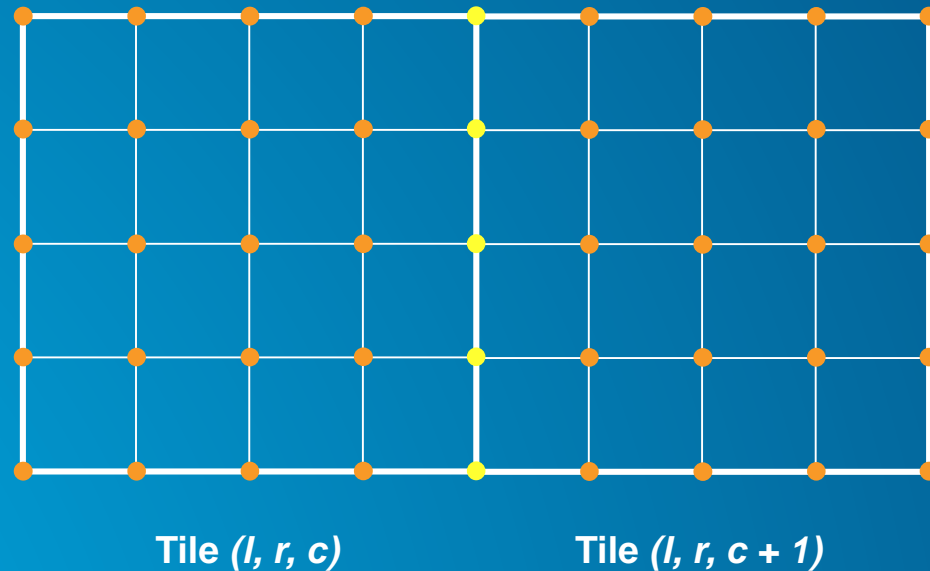
Custom Elevation Layers

- **Elevation tiles**
  - **Pixels/samples are located at the corners of cell grid**

  - **Number of values in one row or column: e.g. 257**
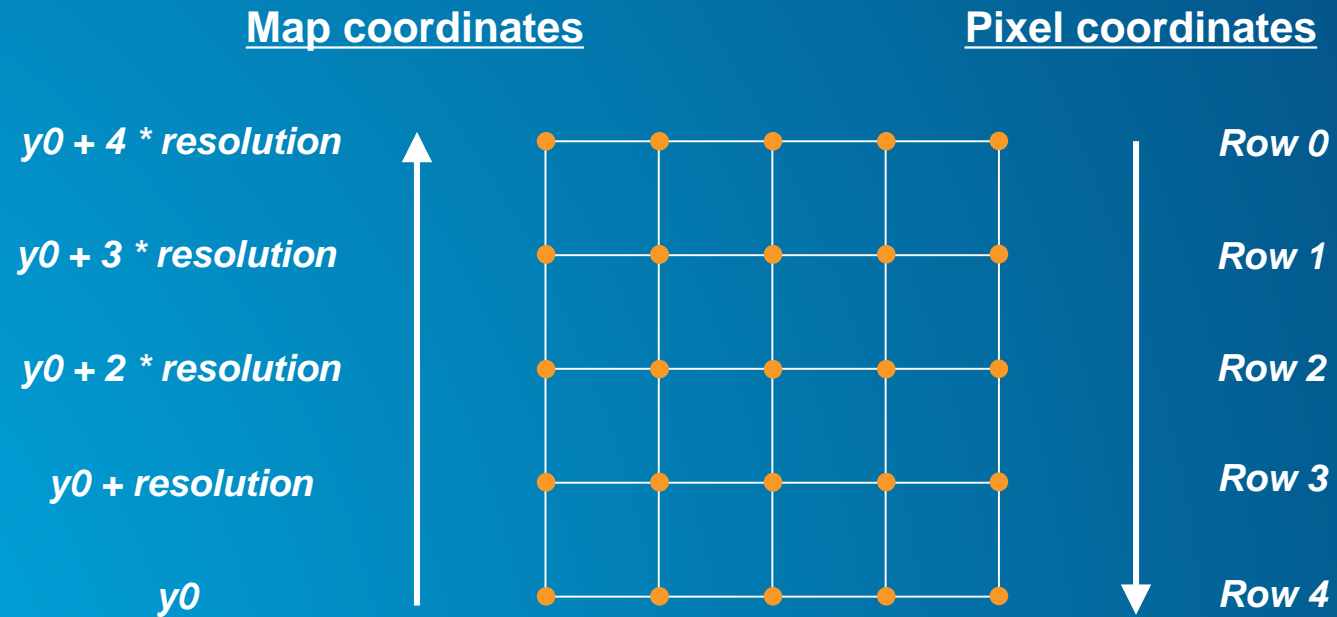 - **But: tileInfo.tileSize[*] = 256 !**

Custom Elevation Layers

- **Elevation tiles**
  - Neighboring tiles have one row of overlapping samples

Tile *(l, r, c)*   Tile *(l, r, c + 1)*

Custom Elevation Layers

- **For more information on Esri Tiled Elevation Services:**

  https://developers.arcgis.com/documentation/tiled-elevation-service/

Using Web Workers

Yann Cabon

Using Web Workers

- **Use worker to execute expensive work on data**
  - **Avoid blocking the main thread**

- **We developed a worker framework**
  - *https://developers.arcgis.com/javascript/latest/api-reference/esri-core-workers.html*

- **Example: Generating contour lines from elevation data**
  - *http://blog.bruce-hill.com/meandering-triangles/*

Future work

Future work

- **Building useful examples**
  - *Loading a TPK and other file types*
  - *Drawing vector data on canvas*
  - Which ones would you like to see?
- **Data refresh / update**
- **Custom LayerViews**