Agenda

1. Stream Services and Stream Layers
2. Publishing Stream Services
3. Visualizing Real-Time Data in Web Maps
4. Application Templates & Web App Builder
5. Using the Stream Layer in Custom Web Applications
This is a beginner-intermediate level technical workshop
Leveraging Stream Services

• We expect that you are familiar with:
  - Web maps and feature layers
  - The ArcGIS GeoEvent Extension for Server ("GeoEvent")

• This presentation will demonstrate and discuss:
  - Publishing stream services using the GeoEvent Manager
  - Discovering stream services in the ArcGIS REST Services Directory
  - Advantages stream services provide vs. traditional feature services
  - Viewing stream layers in maps and apps
1 Overview
Real-Time Capability
ArcGIS 10.4

- Ingest high velocity real-time data into ArcGIS
- Perform continuous analytics on events as they are received
- Store observations in a Spatiotemporal Big Data Store
- Visualize high velocity and volume data:
  - as an aggregation
  - as discrete features
- Notify those who need to know about patterns of interest in real-time
2 Stream Services and Stream Layers
Stream services vs traditional feature services

Two patterns, two important differences

- **Feature layers pull from feature services**
  - Layers poll to get periodic updates
  - Must be backed by an enterprise geodatabase (EGDB) or Spatiotemporal Big Data Store

- **Stream layers subscribe to stream services**
  - Service pushes data to layer as soon as it is received
  - Data is not stored in database
**Stream Layer**

*What is it?*

- A layer in the **Javascript API**
  - Available since version 3.6
- Inherits from Feature Layer

---

**Diagram:**

- **GET** to **HTTP**
- **Feature service**
- **ArcGIS Server**
- **RESPONSE**
- **PUSH** from **Web Socket**

---

**ArcGIS Server**

- **esri/layers/FeatureLayer**
- **esri/layers/StreamLayer**
- **GeoEvent Extension**

---

**Stream service**

- **esri/layers/GraphicsLayer**
- **esri/layers/FeatureLayer**
- **esri/layers/StreamLayer**
Stream Layer

Advantages when using real-time data

- More **responsive** and more **efficient** than feature layers
- Stream layers display **immediately** and refresh **automatically**
- Data is only sent to the client **once**
- Messages are sent **without extra headers**
Stream Layer
Requirements

• ArcGIS GeoEvent Extension for Server
  - Stream services are published as GeoEvent output connectors

• Web Browser that supports Web Sockets
  - http://caniuse.com/websockets

• Network support for the Web Socket protocol
  - ws:// wss://

• No custom plug-in required: Standard JavaScript implementation
Stream Services

*What can I use to consume stream services?*

- ArcGIS Online and Portal for ArcGIS **Web Maps**
- ArcGIS Online and Portal for ArcGIS web application **templates**
- Web applications built using **Web App Builder**
- **Your own web apps** that use the ArcGIS API for JavaScript
Demo
Stream Services and Stream Layers
Viewing Real-Time Data in Web Maps
Real-time Data in a Web Map

- Symbology
- Filtering
4 Application Templates and Web App Builder
Web Application Templates

“Pre-packaged” Web Apps

- “Configurable Apps” tab
- Many to choose from
- Many are theme-focused
  - Compare
  - Elevation Profile
  - Impact Summary
- Can publish, download or preview
Web AppBuilder Apps
Share in ArcGIS Online / Portal

- Also via the “Share” button
- Also use “Create a Web App”
- “Web AppBuilder” tab
- Enter a title, add some tags, enter a summary
- “Get Started”
Demo
Web Maps
Templates and Web App Builder
Custom Web Applications
Real-time data in Your Own Web App

Very little code!!

- Dojo “require”
- Construct and add to map

```javascript
require([
  "esri/map",
  "esri/layers/StreamLayer",
  "dojo/domReady!"
]

var url = "https://geoeventsample3.esri.com:6443/arcgis/rest/services/AsdiTracks/StreamServer";
var streamLayer = new StreamLayer ( url );
var map = new Map( "mapDiv", {
  basemap: "topo"
});
map.addLayer ( streamLayer );
```
Stream Layer
Get rid of unneeded features

- **purgeOptions**
  - `displayCount`: Maximum number of features to display
  - `age`: Maximum age of features (in minutes). Defaults to no maximum.

- **maximumTrackPoints**: Maximum features per trackId to display. Defaults to 1

- **purgeInterval**: The purge method is automatically called at this interval (in minutes). Defaults to 0 so purging performed when new message is received.

Note: GeoEvent definition “TIME_END” field is honored

```javascript
var streamLayer = new StreamLayer(url, {
  purgeOptions: {
    displayCount: 1000,
    age: 20
  }
});
```
Stream Layer

Setting Filters on Data

- **definitionExpression**: The where clause used to filter data using attributes.

- **geometryDefinition**: The Extent used as a spatial filter. Only Extent is allowed.

```javascript
var streamLayer = new StreamLayer(url, {
  definitionExpression: "AltitudeFeet > 18000",
  geometryDefinition: new Extent({
    xmin: -120,
    ymin: 38,
    xmax: -115,
    ymax: 42,
    spatialReference: {
      wkid: 4326
    }
  })
});
```
Stream Layer
Setting Filters on Data after layer is created

- After layer is created use `setDefinitionExpression` and `setGeometryDefinition`.
- Setting the properties to `null` clears an existing filter.
- Use `filter-changed` event to determine if filter was set successfully.

```javascript
streamLayer.on( "filter-change", function(evt) {
    if (evt.error) {
        console.log ( evt.error );
    } else {
        console.log ( evt.filter );
    }
});
```
Stream Layer

Events to use

- **error**: Use to determine why layer could not be created.
- **connect**: Use to listen for web socket connection opened
- **disconnect**: Use to listen for web socket connection closed
- **attempt-reconnect**: Use to determine when a reconnection attempt is made and how many have been made
- **message**: Use to get message sent to web socket without having to listen to map events
DEMO

Custom Applications
Conclusion
Summary

Your Session Title Goes Here

• Conclusion
  - Summary 1
  - Summary 2
  - Summary 3

• To learn more see the tutorials.
Real-time: Leveraging Stream Services

Helpful links

• StreamLayer API help:
  https://developers.arcgis.com/javascript/jsapi/streamlayer-amd.html

• Stream Layer Sample:
  https://developers.arcgis.com/javascript/jssamples/layers_streamlayer.html

• Sample Applications on Github:
  https://github.com/jdelgadillo/UC2016-StreamServices/

• Sample Stream Services with Simulated Data:
  https://geoeventsample3.esri.com:6443/arcgis/rest/
Please take our Survey
Your feedback allows us to help maintain high standards and to help presenters

Find your event in the Esri Events App

Find the session you want to review

Scroll down to the bottom of the session

Answer survey questions and submit
Questions / Feedback?

Suzanne Foss
Product Engineer
Real-Time GIS Team
sfoss@esri.com

Javier Delgadillo
Real-time Engineer
Geoevent Team
jdelgadillo@esri.com