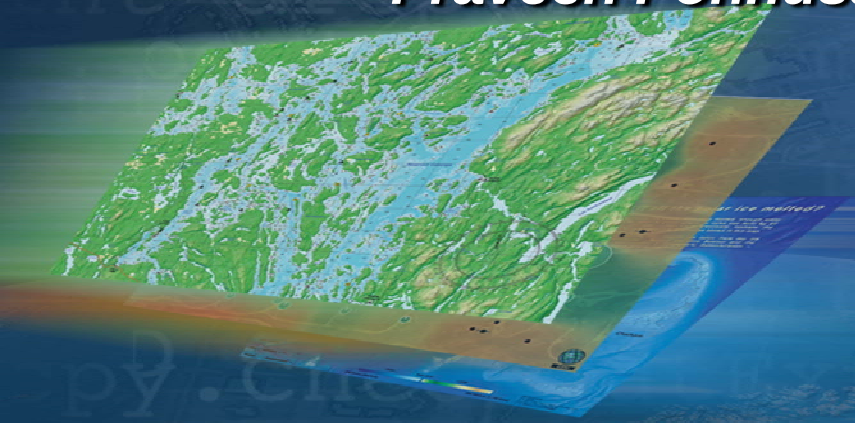
ESRI Developer Summit

March 22–25, 2010
Palm Springs, CA

Working with the ArcGIS API for JavaScript 2.0

Jeremy Bartley
Kelly Hutchins
Praveen Ponnusamy



ESRI

What is new in the ArcGIS API for JavaScript at 2.0

- Client side FeatureLayer
- Integrating Time into your applications
- Building Editing applications
- Enhanced ImageServer layer and Network Analyst task
- Making the API easier for you
- Leveraging the platform – HTML5

esri.layers.FeatureLayer

FeatureLayer – A new type of Operational Layer

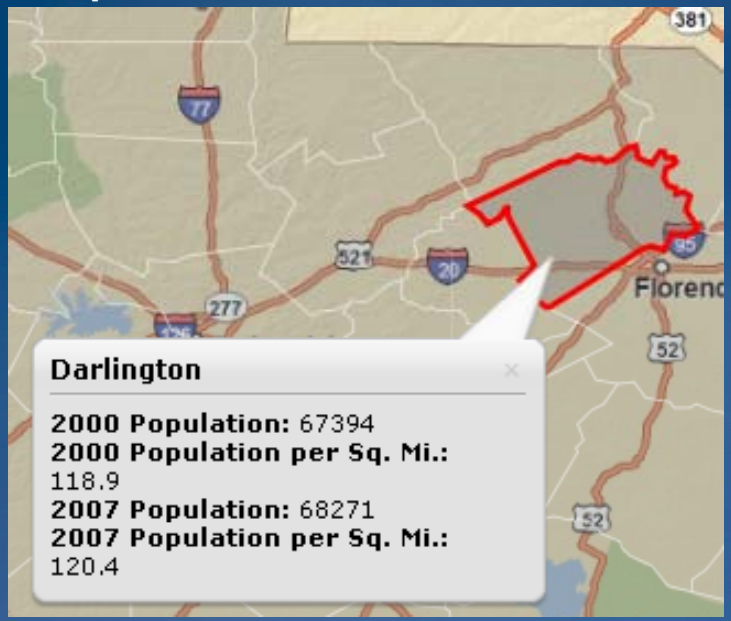
- Extends GraphicsLayer
- Works against a Layer / Table in a
  - Map Service
  - Feature Service
- Supports
  - Selection
  - DefinitionExpression
  - Query
  - QueryRelated
  - Edits (when working with a FeatureService)

FeatureLayer – A new type of Operational Layer

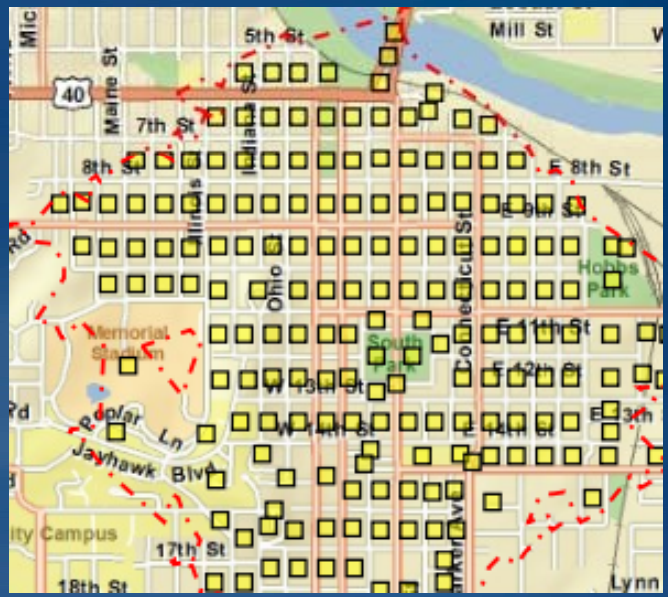
- Honors
  - Scale dependency
  - Renderer (with ArcGIS Server 10 services)
  - Labeler definition (not in public beta)
- Advertises
  - Field level domains
  - Attachments
  - Templates (editing)
  - Relationships (with ArcGIS Server 10 services)
  - Types and Templates (when working with ArcGIS 10 FeatureService)
- Mode
  - How are the features delivered to the client?

FeatureLayer View Mode

Snapshot – All Features on Client



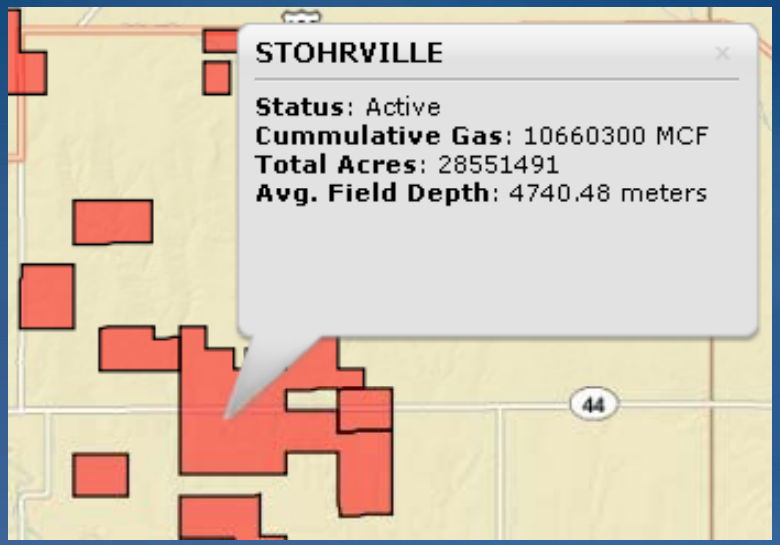
Selection Only – Only Selected Features



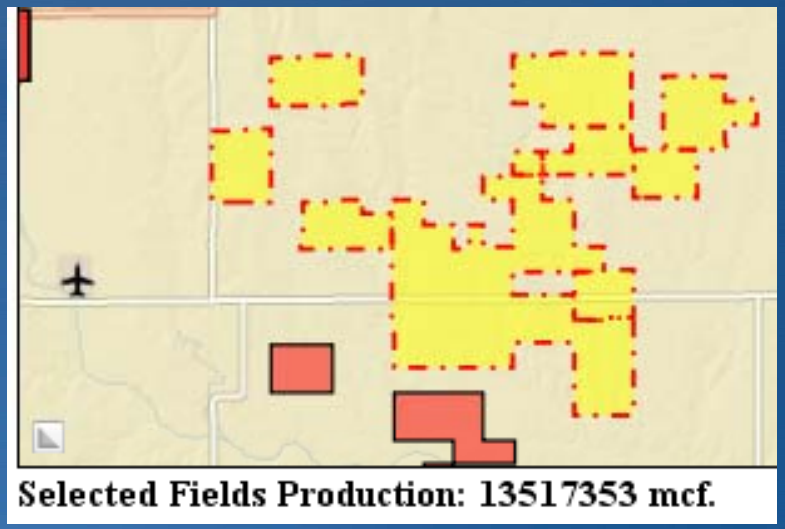
OnDemand – Request Features as map moves

FeatureLayer – Layer definition and Selection

Layer definition applied by FeatureLayer



FeatureLayer supports selection

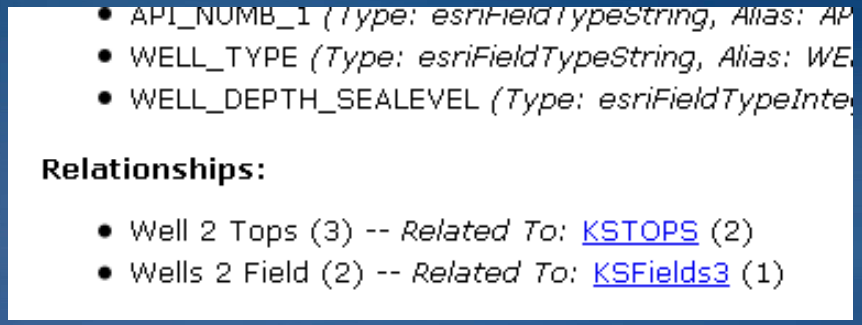


FeatureLayer QueryFeatures and SelectFeatures Details

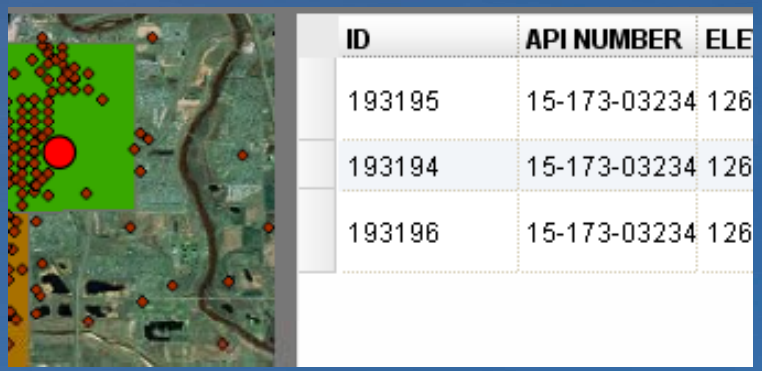
- Both take esri.tasks.Query() as input
- Selected Features are always held on the client
- Mode dictates where query is executed.
- queryFeatures and selectFeatures happen on the server except for the following cases:
  - Snapshot: Queries by TimeExtent, ObjectIds, and Extent based Geometry with intersection relationship are done on the client
  - On-Demand: Queries by extent based Geometry with intersection relationship are done on the client if the extent is within the current map extent
- Selection Only Mode: Always executed on the server

FeatureLayer QueryRelatedFeatures

- **FeatureLayer advertises Relationships as defined in the GDB and exposed via the REST API**

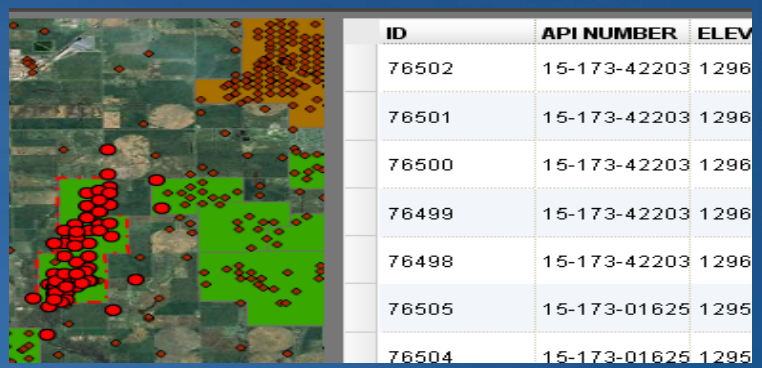


Query related features from 1 feature



ID	API NUMBER	ELE
193195	15-173-03234	126
193194	15-173-03234	126
193196	15-173-03234	126

Query related features from multiple features



ID	API NUMBER	ELEV
76502	15-173-42203	1296
76501	15-173-42203	1296
76500	15-173-42203	1296
76499	15-173-42203	1296
76498	15-173-42203	1296
76505	15-173-01625	1295
76504	15-173-01625	1295

Integrating Time into your applications

Arroyo Field Production by Time - Mozilla Firefox

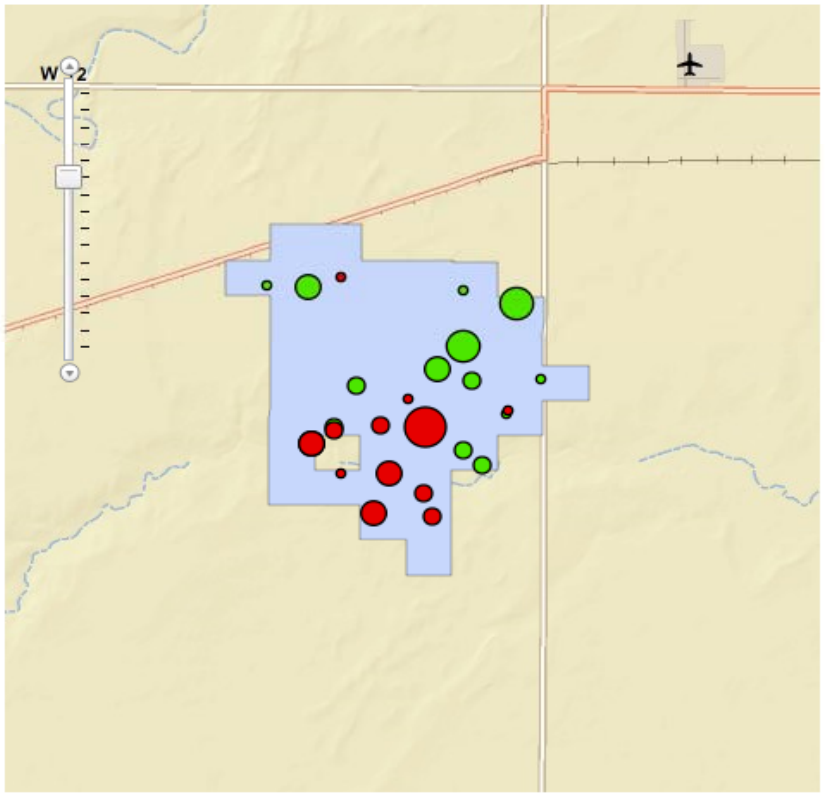
File Edit View History Bookmarks Tools Help

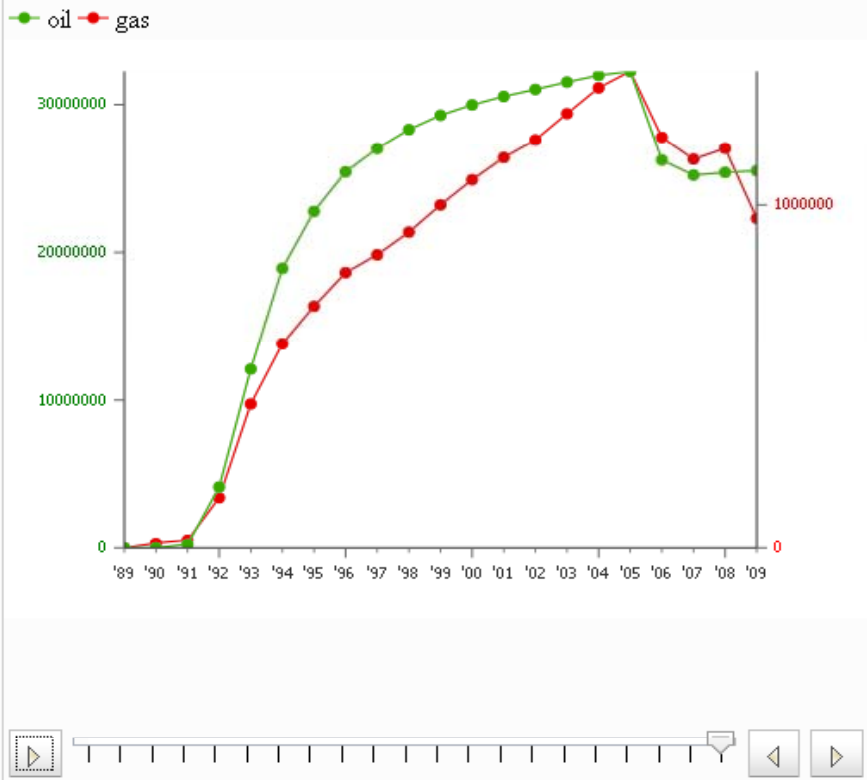
http://json/jsapidemos/2.0/Time/chartsample.html

prepetual time

ArcGIS Server Blog | Basic Service Area | Editable Points of In... | Feature Layer: Poin... | JavaScript API Refe... | Arroyo Field Pro...

Arroyo Field Production in *2009* Stanton County Gas Leases produced 25522524 Mcf with 17 wells. Stanton County Oil Leases produced 961128 bbls with 12 wells.





Find: ontime Next Previous Highlight all Match case

Waiting for serverapi.arcgisonline.com...

Map is time aware!

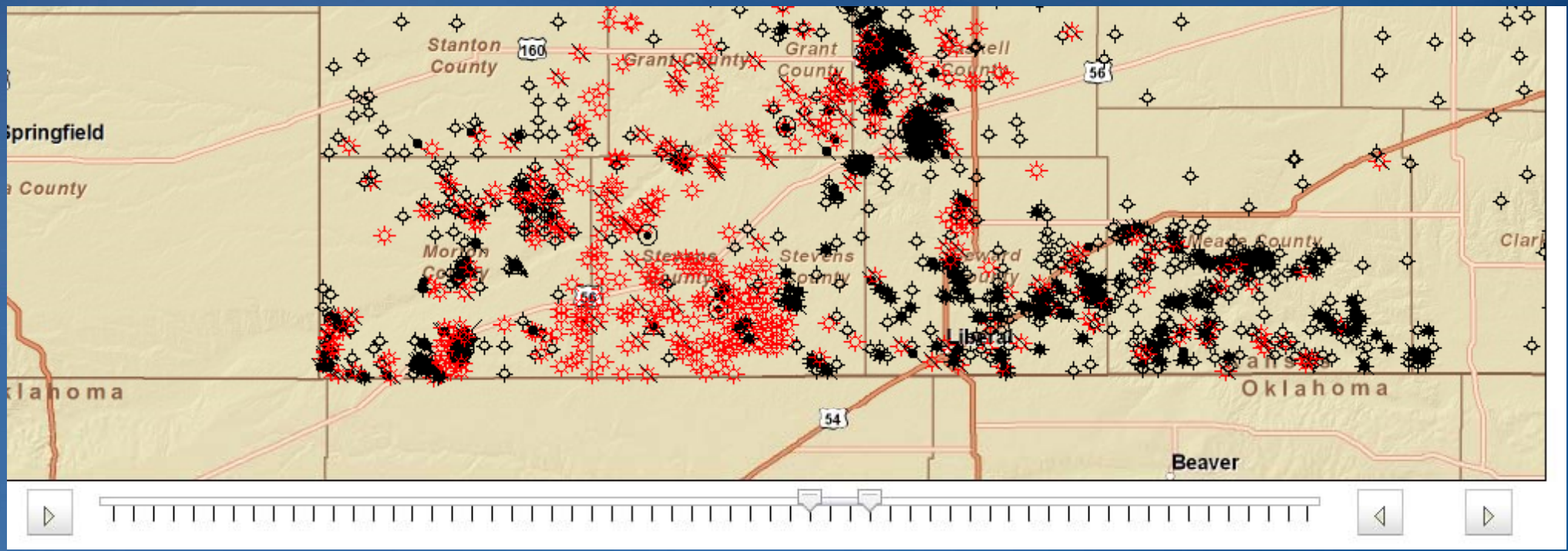
- Map includes a setTimeExtent() method
  - Control time like you control spatial extent
- Map can be driven by a TimeSlider
- Map drives time aware layers
- TimeSlider is an out of the box Dijit that can be placed anywhere
- Map and TimeSlider fire onTimeExtentChange event

esri.TimeExtent()

- TimeExtent has startTime and endTime properties
- If startTime and endTime are equal then timeExtent represents a time instant.
- startTime and endTime are Date objects
- var endTime = new Date(); //represents now
- var startTime = new Date(2001); //represents 1/1/2001 UTC

More about TimeSlider

- One thumb to represent cumulative or time instant
- Two thumbs for end user controllable start and end time
- End users can control time slider manually or by animating over the time span



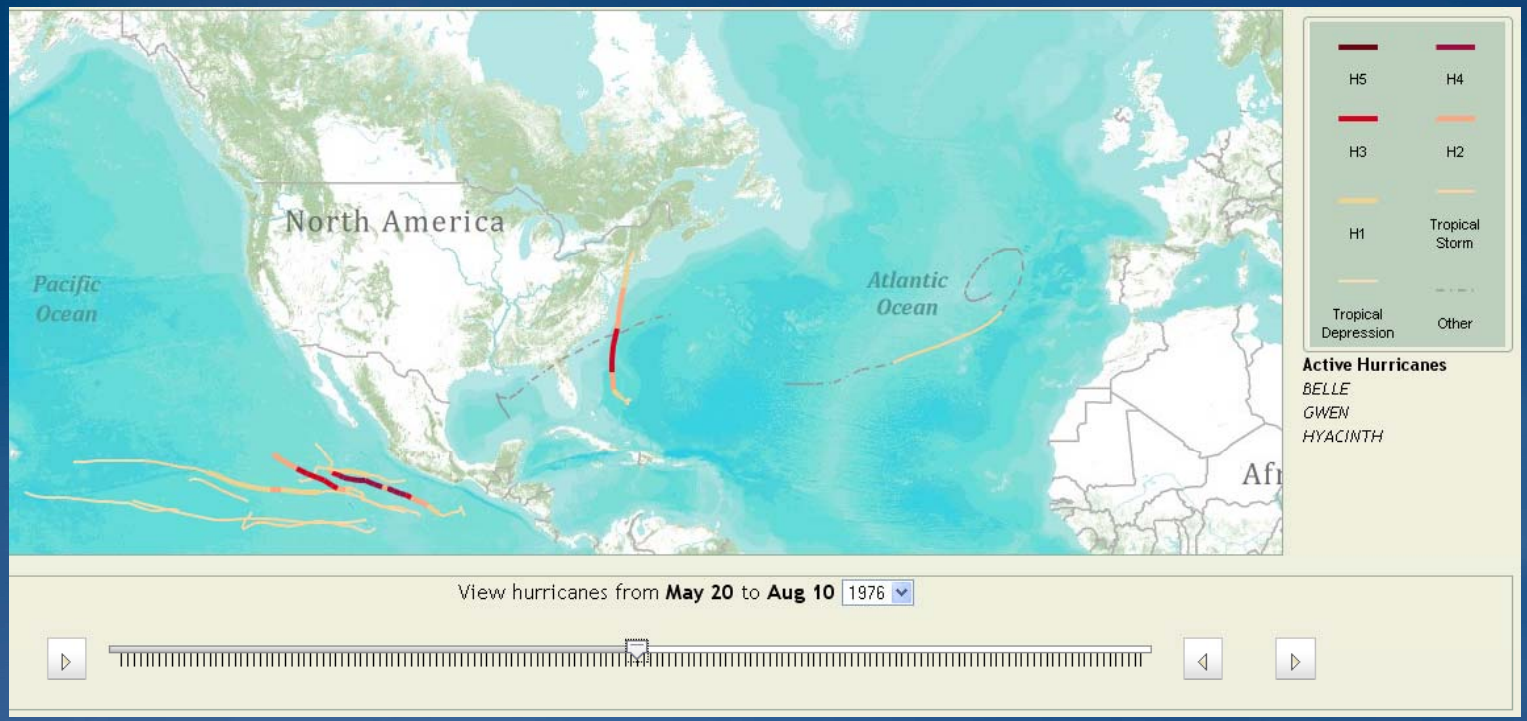
Time-aware ArcGISDynamicMapServiceLayer and ArcGISImageServiceLayer

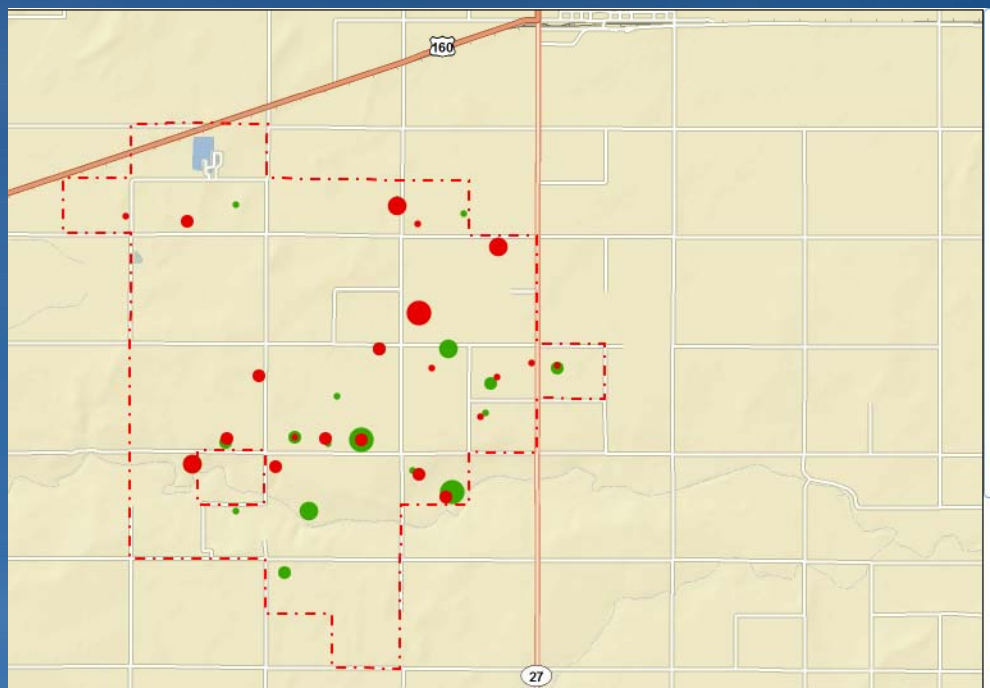
- Created as an image on the server
- Each time step interval is a new image request
- Use ArcGISDynamicMapServiceLayer if you have too many features that cannot be drawn as a FeatureLayer

Time-aware FeatureLayer

- Client-side rendering with a Symbol
- Temporal Renderer
- Event handling
- InfoWindow
- Client side queries for time
  - If FeatureLayer Snapshot mode
- Use if you can handle all features on the client (snapshot)

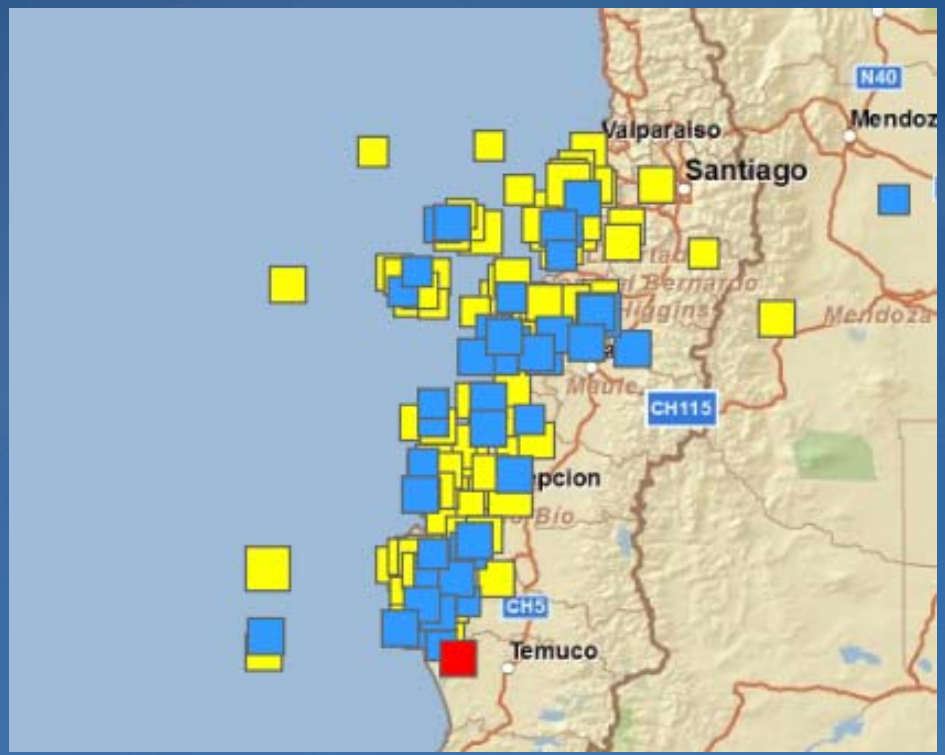
Time aware FeatureLayer

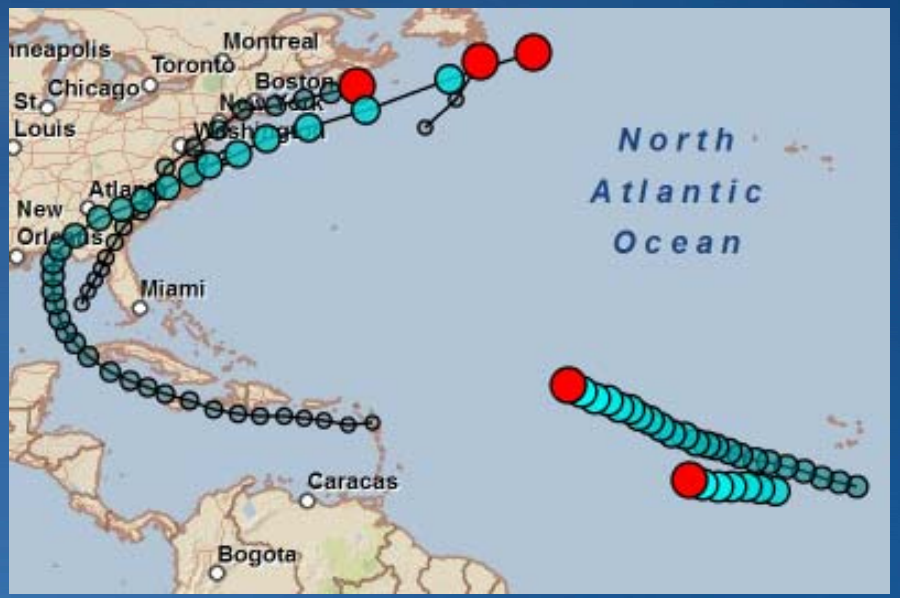




Temporal Renderer

- Observation renderer
- Last Observation renderer
- Track renderer
- Time Class Breaks Ager





Time Tips and tricks

- Leverage Dojo for common time calculations
  - Compare two dates: dojo.date.compare(currentDate,minDate)
  - Difference between two dates: dojo.date.difference(timeStart, currentTime, time units)
  - Add to an existing date: dojo.date.add(date, interval, amount)
  - Format Dates based on locale: dojo.date.locale.format

```
var formattedDate = dojo.date.locale.format(date, {
            selector: 'date',
            datePattern: 'MMM dd, yyyy' }
```

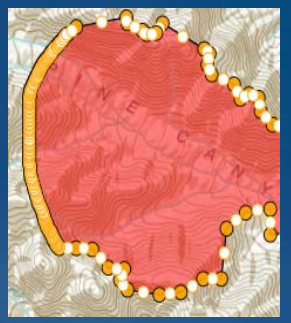
Editing

Editing Basics -- Server

- Works against ArcGIS Server Feature Service
- Template style editing
  - Type symbol and template (default values) are returned by service
- Last in wins
- Domains and subtypes exposed and supported
- You can edit stand alone tables and attachments
- All edits to FeatureService are through HTTP POST
  - If your app is not on the same domain as your GIS server make sure you set up your proxy!

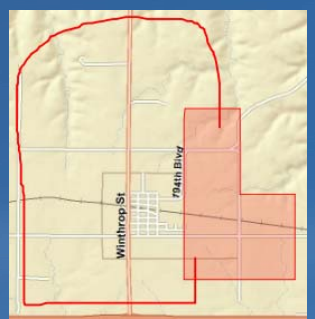
Strong low level support for Editing in API

- FeatureLayer
  - Constructed with a layer in a Feature Service: FeatureLayer.applyEdits(adds, updates, deletes)
- Edit Toolbar
  - API for manipulating/transforming geometry of existing graphics
- Generate symbol shapes on custom DOM node
  - esri.Symbol.getShapeDescriptors()
- Enhanced GeometryService task to support editing operations
  - Auto Complete, Reshape, Cut, Union, Intersect, Difference, Trim Extend, Convex Hull
  - DEMO



FeatureLayer + Edit Toolbar

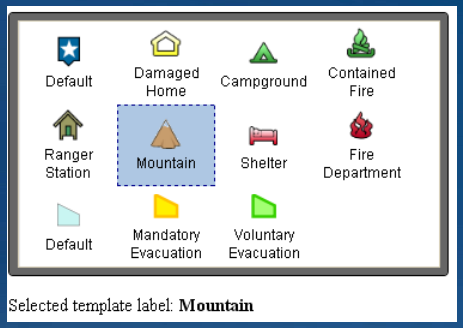
FeatureLayer + Geometry Service Reshape



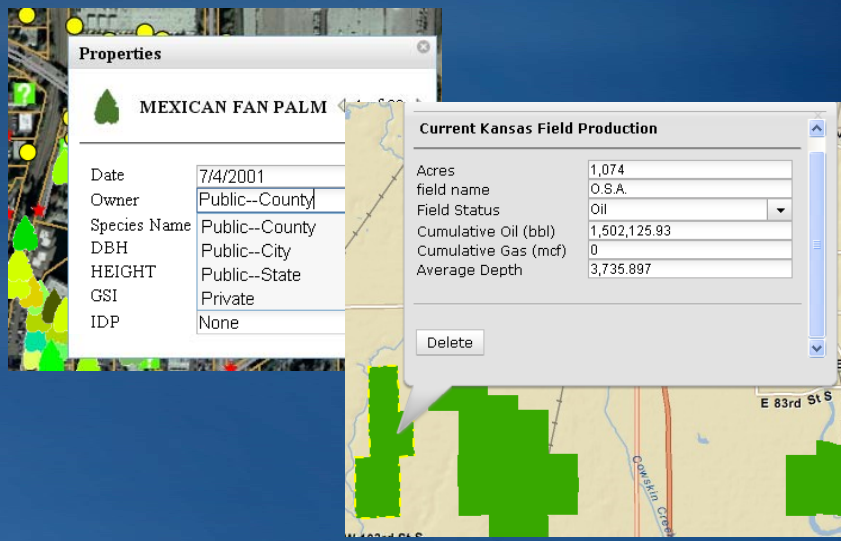
*GS task now works with geometries!

JSAPI includes a set of easy to use Widgets

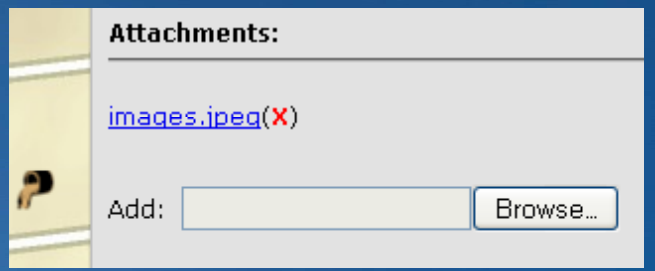
- **Template Picker**
  - Client side widget to display symbols as defined in FeatureLayer
  - Can also be used as a legend



- **Attribute Inspector**
  - Widget to support modifying attributes and deleting features
  - Honors domains set in FeatureLayer
  - Fires onAttributeChange event
  - Can be placed in any div
  - Updates after FL.onSelectionComplete and FL.onEditsComplete



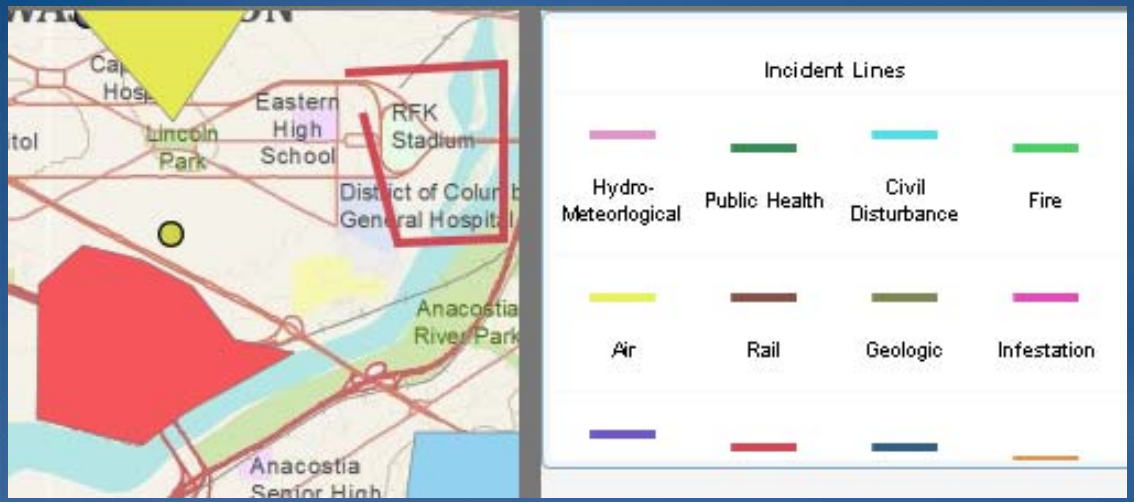
- **Attachment Editor**
  - Widget to support viewing, creating, and deleting attachments



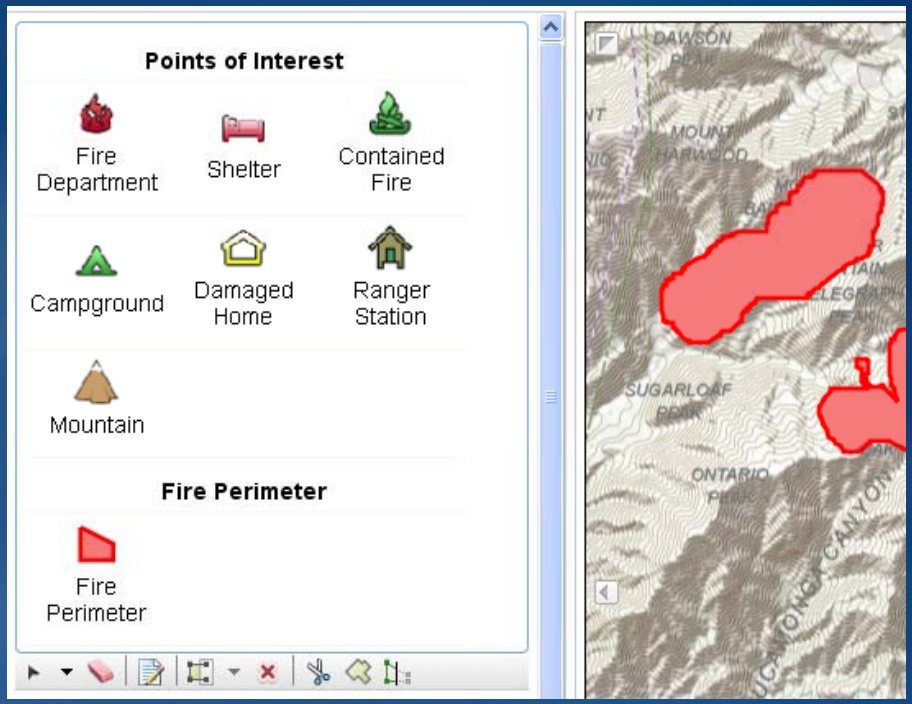
Editor Widget

- Web APIs support creating focused editing applications built using the underlying building blocks of the API
- Web APIs also support a complete, configurable Editing Widget
- Editing Widget is similar to ADF Editor Task, but works with Feature Service Templates
- Editing Widget can be configured to add/remove functionality
- New ArcGIS.com WebMap will leverage the Editing Widget

Editor Widget



Simple Editor Widget – Template based editing



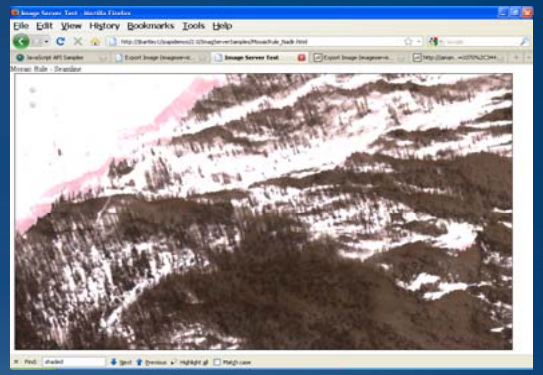
Editor Widget configured with advanced tools

Editing work left for 2.0

- Snapping
- Undo/Redo

Enhanced ImageServer layer and Network Analyst task

Enhanced ArcGISImageServiceLayer



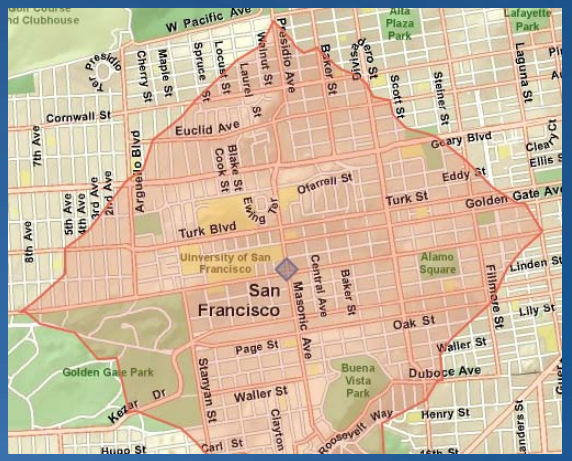
- Control mosaic definition
  - LockRaster, Nadir, Seamline, Center, and others
- Query raster catalog
  - Query footprints and get access to raw image
- Identify on ImageService
  - Get access to the pixel values against default service or by custom MoasicRule
- Perform Raster functions
  - Slope, Hillshade, NDVI, Colormap, Stretch

Network Analyst Task updates

- **RouteTask**
  - Now supports polyline and polygon barriers

- **ClosestFacilityTask**
  - Route to the closest facilities

- **ServiceAreaTask**
  - Drive time



Making the API easier for you

Making the API easier: Helper methods and overall enhancements

- Support for Well Known Text with ArcGIS server 10
  - Watch out for URL length!
- A new way to control layer visibility
  - layer.setVisible(true/false);
- Get access to the Layer that contains the Graphic.
  - var layer = graphic.getLayer();
- Convert arrays of Graphics to arrays of Geometries
  - var geoms = esri.getGeometries(graphics);
  - Array index order is the same

Making the API easier: Helper methods and overall enhancements

- Client side spatial query
  - if (extent.intersects(geometry)) { //geometry intersects extent }
- Rename the dojo namespace that is returned by the JSAPI
  - Change dojo.connect() to esriDojo.connect() New at 1.6
- Add multiple layers to the map at one time.
  - dojo.connect(map,"onLayersAddResult",layersAddedFunc);
  - map.addLayers([layer1,layer2]); //array order defines order on map
- Layer definition support for Identify and Find
- Support for out spatial reference in locator

Smarter API

- Team is working on optimizing the delivery of the API
- Smaller download
- Device Recognition (iPhone versus IE on Windows)
- Device specific event support

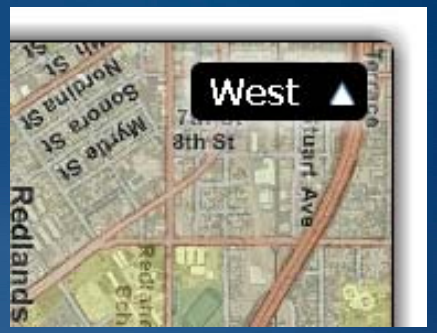
Leveraging the HTML 5 platform

HTML 5

- Not a standard yet, but a collection of functionality for developing rich internet applications without a plugin.
- You can leverage it now
  - Local Storage
  - Geolocation
  - Timed Media playback (movies/audio)
  - Native Rich Text Editor
- Not all browsers are equal. Only use if you can guarantee browser control (Phone for example) or your app can gracefully fallback.

Leveraging HTML5 enhancements in the JSAPI*

- CSS Transitions with CSS Level 3
  - Smooth animation between zoom levels and pans
  - Ease while panning
  - CSS Transforms



- Web Workers
  - Browser has 1 thread.
  - Web workers allow you to spawn several threads to do things like: calculations, information retrieval, and parsing

- Canvas
  - Immeadiate Draw mode, may support in Graphics Layer for devices that only support Canvas
  - No Native events

*Still experimenting. No promises at this time.

Questions?