



ArcGIS Runtime SDK for Java: Building Applications

Eric Bader

@ECBader

Esri Developer Summit
Washington, DC

Agenda

- ArcGIS Runtime and the SDK for Java
- How to build / Functionality
 - Maps, Layers and Visualization
 - Geometry Engine
 - Routing / Geocoding
 - Advanced analysis with Geoprocessing
 - Message Processing
- What's next

ArcGIS Runtime



- Family of SDKs for multiple platforms
 - Consistent capabilities
- Native to the platform
 - For building great apps
- Lightweight and fast
- Powerful
- Easy

Part of the ArcGIS platform

- ArcGIS Online / Portal
 - Maps, services, content and Organization branding

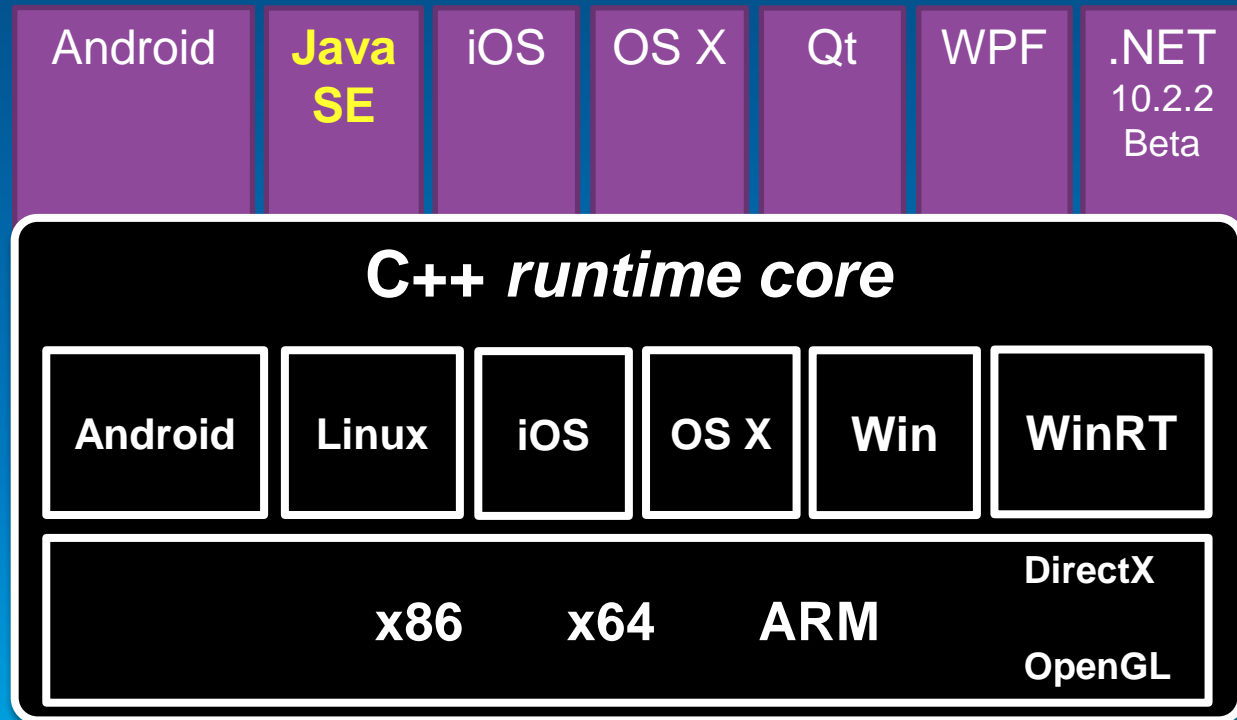


- ArcGIS for Server
 - Services



- ArcGIS for Desktop
 - Packages

Runtime Architecture



ArcGIS Runtime SDK for Java

- Build native apps for Windows and Linux
 - Windows 8.1 / 8 / 7 / 2012 / 2008 / Vista* / XP*
 - Linux: Ubuntu / RedHat
 - 32 and 64 bit
- Java SE API 7u40 / 6u45
- Java Swing
- Eclipse plugin



** 10.2 was the final release of these*

What you get

- The SDK includes:
 - Eclipse plugin
 - Set of jars to code against
 - Open-source toolkit (mainly UI components)
 - Tools to license / deploy / debug
 - Concept documentation, API reference
 - Tons of samples



How you get the SDK

1. Free developer subscription:
<http://developers.arcgis.com/plans>
2. Download:
<http://developers.arcgis.com/java>
3. Register the SDK (free):
<http://developers.arcgis.com/licensing>
4. Build!

Demo

The SDK



Functionality

Geoprocessing	Network Analysis (Routing, ...)	Editing
Geometry Engine	GPS	Search (Geocoding, ...)
Mapping & Visualization (online and local content)		

Mapping & Visualization

- Add data as Layers
- Online
 - ArcGIS Online
 - ArcGIS for Server
 - OGC (WMS)
 - Bing
 - OpenStreetMap
 - KML
- Local
 - Tile packages
 - Map packages
 - Locator packages
 - Geoprocessing packages



Hierarchy of Map Layers



- Layers are added bottom-to-top
- Graphics layers: live / temporary data:
 - Vehicles, people, events
- Operational layers: dynamic features
 - Facilities, buildings, zones, networks
- Basemap layer: spatial context
 - Imagery, topography

Mapping & Visualization

- GPS Layer
 - File or Serial port
 - NMEA
- Graphics Layer
 - Custom data
 - Point, Line, Polygon
 - Symbology

Geometry Engine



Buffer



Difference



Intersect
(equals, within, contains,
crosses, touches)



Union



Project



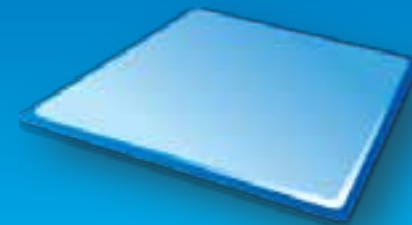
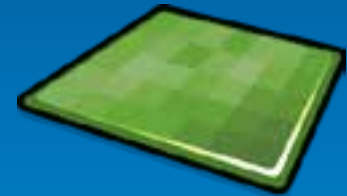
Shortest Distance



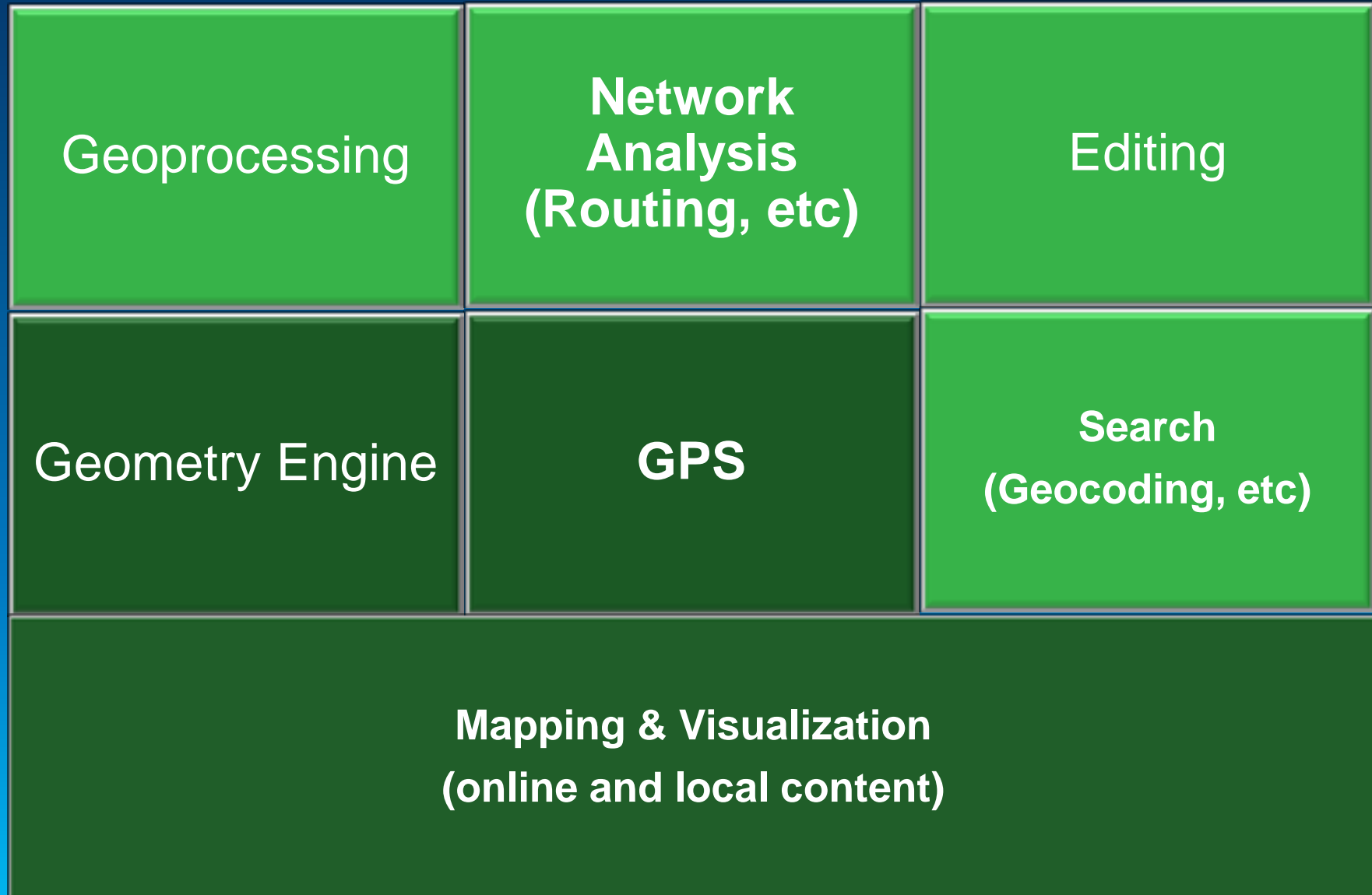
Line Length

Demo

Mapping



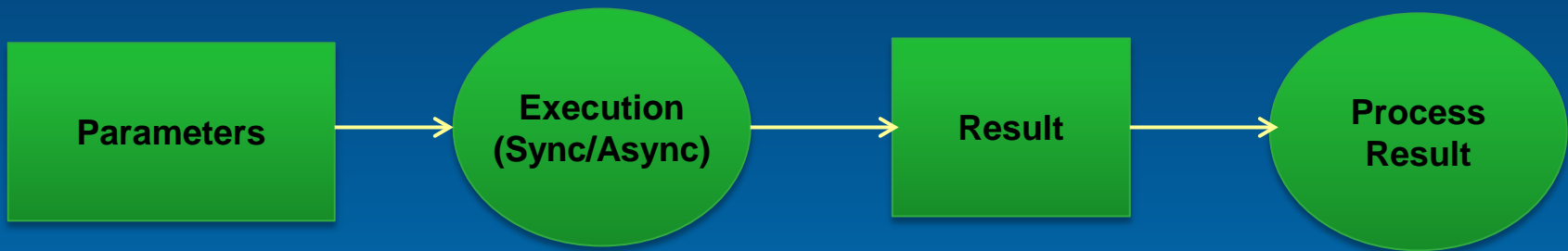
Java SE Runtime Functionality



Search

- Search spatial data points (Features)
- Search tasks
 - Query
 - Find
 - Identify
 - Geocoding
- Synchronous and asynchronous pattern
- Online and offline
- Can accept input in one spatial reference, and output in another

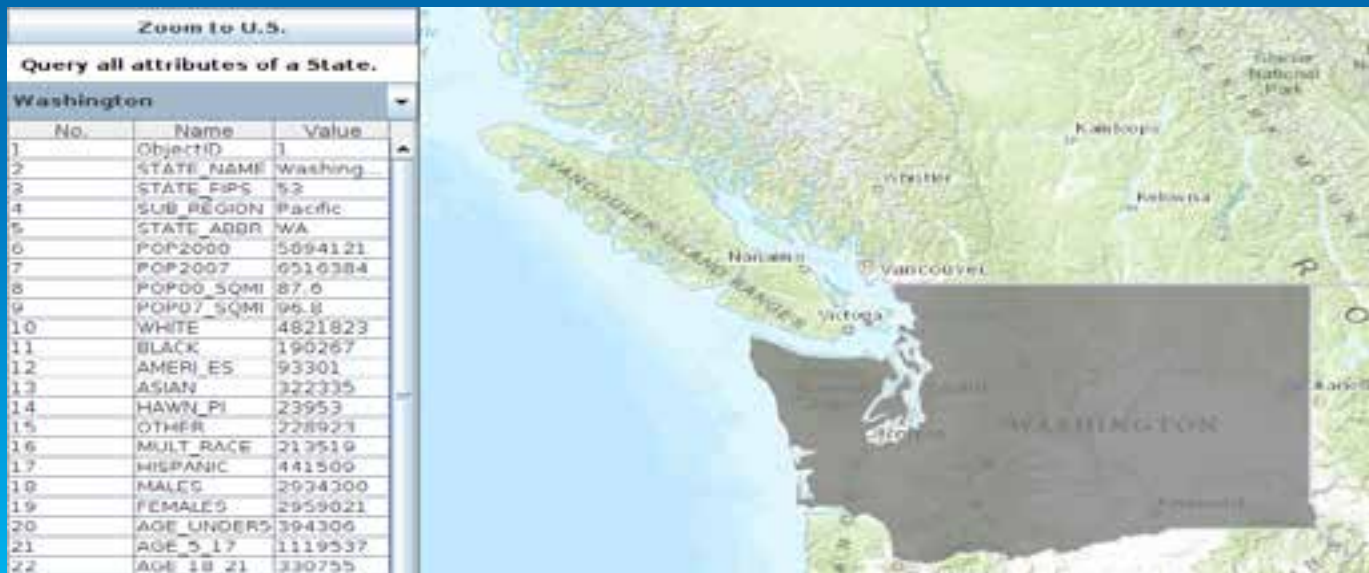
Search – asynchronous processing



- Execute Sync vs. Execute Async
 - Sync is a blocking call
 - Async preferred when called from Swing's UI thread
 - Handle async result using callbacks
 - `Operation.executeAsync(parameters, <callback>);`

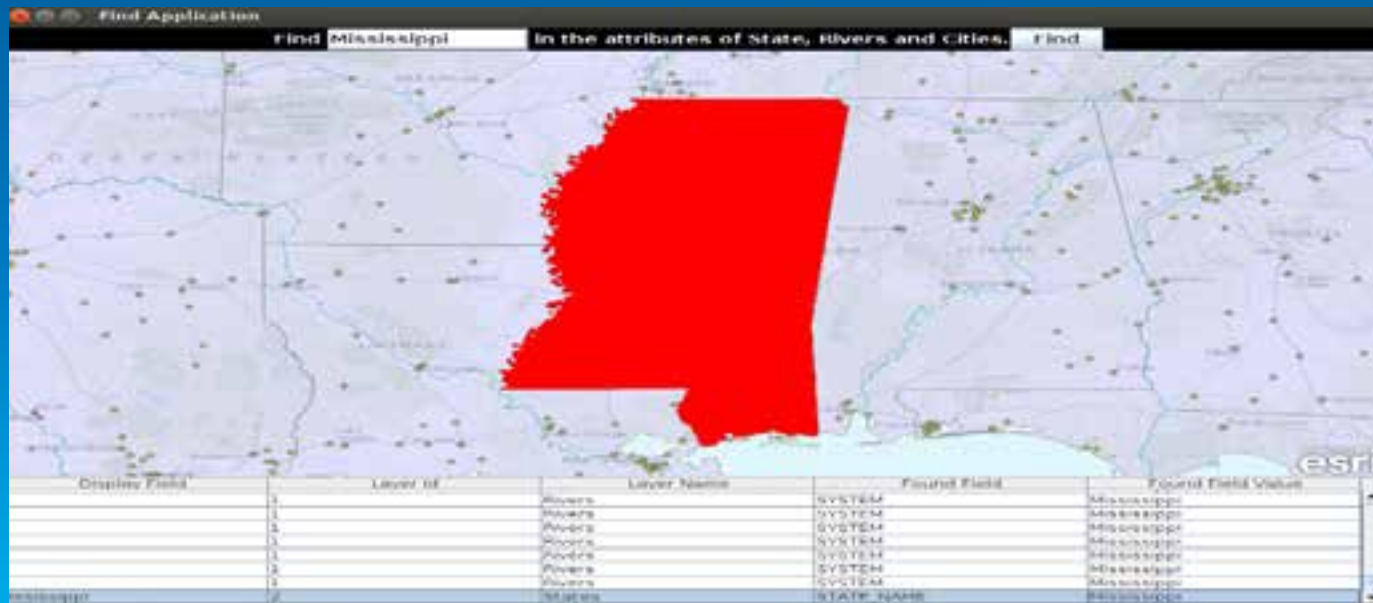
Search - Query

- Search for features that meet some conditions
- SQL-like syntax
- Non-spatial filter
 - E.g. query cities where population > 1000000
- Spatial filter
 - E.g. query cities in selected area



Search - Find

- Searches attributes of features for some given text
- Can search multiple layers
- Useful for free-form search
 - E,g., Find features with “Mississippi”
 - Returns cities in Mississippi, river, state



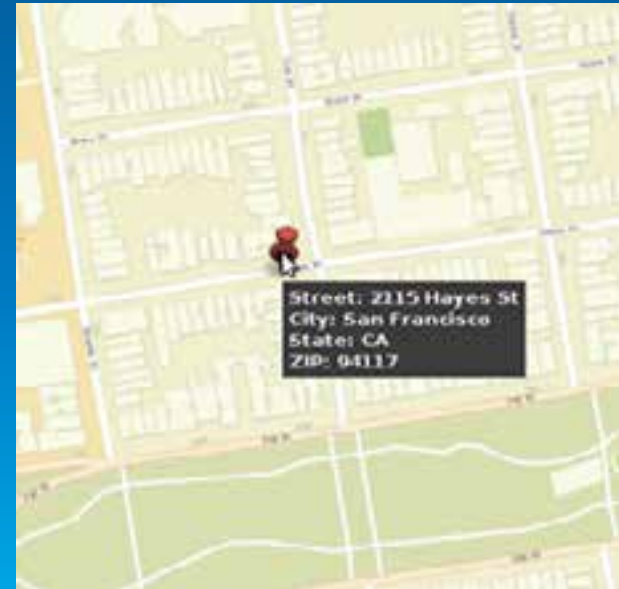
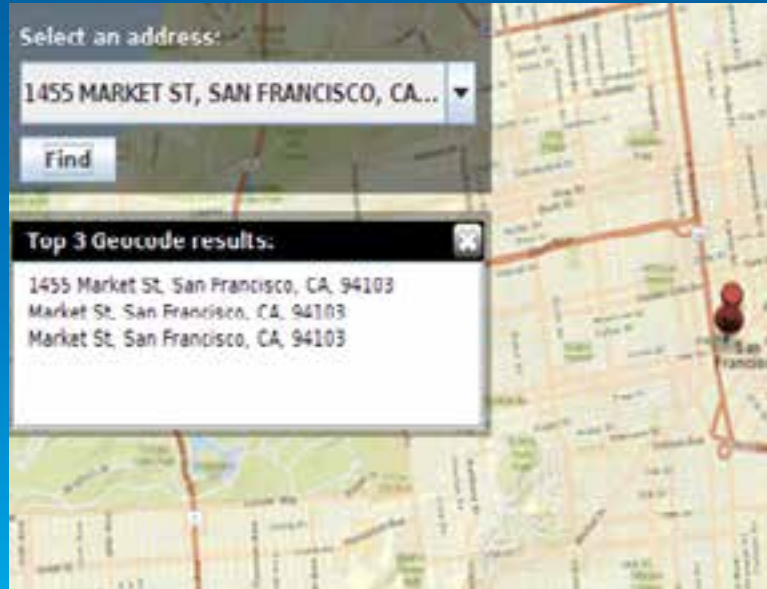
Search - Identify

- Searches all features at a given geometry
- Can identify on multiple layers
- e.g., clicking on Palm Springs can return the city of Palm Springs, and its state California.



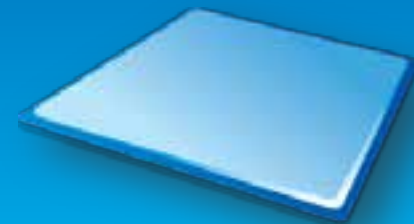
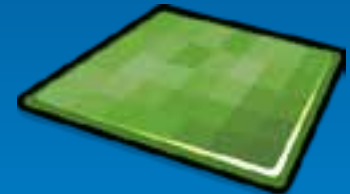
Search - Geocoding

- Works with a geocoding service online or a locator package
- Input single line or address fields
- Batch processing
- Reverse-geocoding



Demo

Geocoding

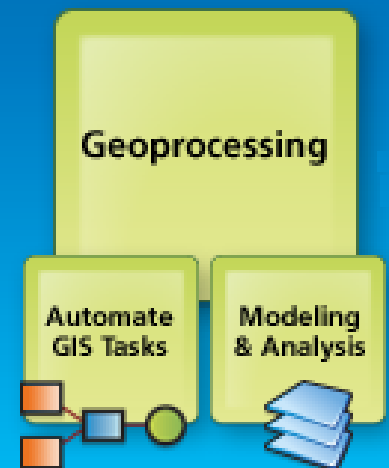


Functionality

Geoprocessing	Network Analysis (Routing, etc)	Editing
Geometry Engine	GPS	Search (Geocoding, etc)
Mapping & Visualization (online and local content)		

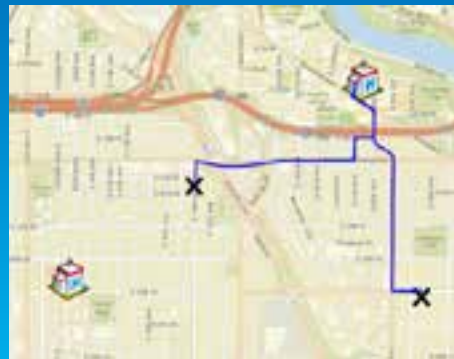
Geoprocessing

- Set of tools included in the ArcGIS System to perform GIS analysis
- Custom script tools implemented using ModelBuilder or Python
- Available
 - Online - hosted as a geoprocessing service by ArcGIS server, or
 - Offline - local geoprocessing packages (GPK)
- Execute the service as a task from the API



Network Analysis

- Routing
 - Uses route service
 - Option to set stops, preserve order, add barriers
- Closest Facility
 - Find routes to closest facility
 - E.g., find routes to closest hospital from an accident
- Service Area
 - Calculate service areas of facilities
 - E.g., calculate area that can be reached in 5 minutes from a hospital



Demo

Geoprocessing & Network Analysis



Functionality

Geoprocessing	Network Analysis (Routing, etc)	Editing
Geometry Engine	GPS	Search (Geocoding, etc)
Mapping & Visualization (online and local content)		

Editing

- Performed on a ArcGIS FeatureLayer
- Create, Edit, Delete features
- Edit attributes and geometry of a feature
- Using API – for fine-grained control
- Using Toolkit – for rapid development
- Online and local

Editing – using toolkit

- Features
 - Template Picker
 - Creates templates of features; can be used to add new features
 - Drawing Overlay
 - Helps in creating different geometries on map
 - Editing Toolbar
 - Provides option to select, edit, delete features



Editing – using API

```
// obtain the clicked point
Point mapPt = map.toMapPoint(mouseEvent.getX(), mouseEvent.getY());
// create a map of attributes (keys must match fields in the feature layer)
Map<String, Object> attributes = new HashMap<String, Object>();
attributes.put("Type", "Park");
attributes.put("Description", "Editing...");
// prepare the Graphic to add
Graphic g = new Graphic(mapPt, new SimpleMarkerSymbol(Color.RED, 10, Style.CIRCLE), attributes, 0);
Graphic[] adds = {g};

if (featureLayer.isEditable()) {
    featureLayer.applyEdits(
        new Graphic[] {},
        new Graphic[] {},
        new CallbackListener<FeatureEditResult[][]>() {

            @Override
            public void onCallback(FeatureEditResult[][] objs) {
                // do something with the feature edit result object
            }

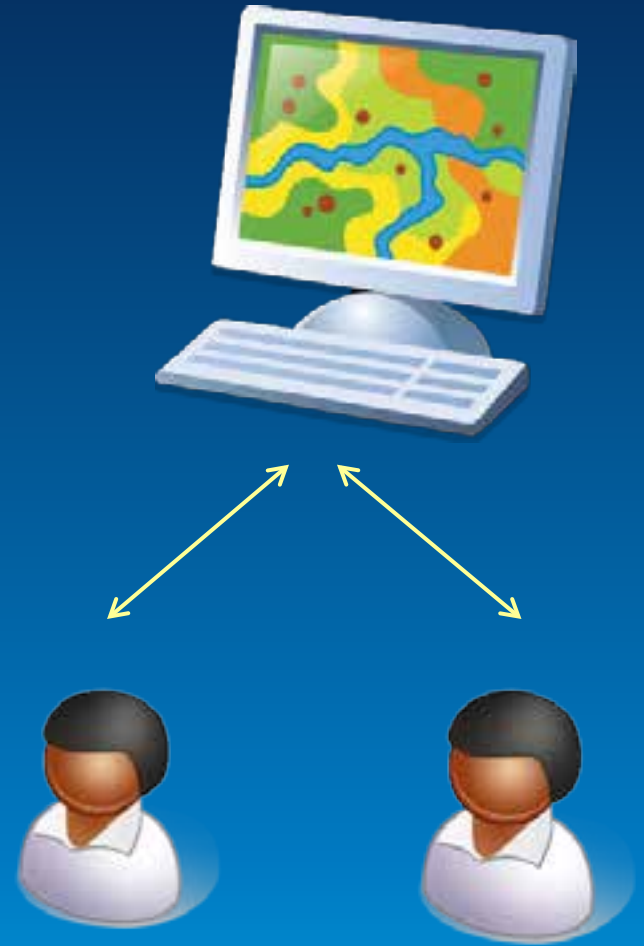
            @Override
            public void onError(Throwable e) {
                // handle the error
            }
        }
    );
}
```

Editor tracking

- ArcGIS FeatureLayer can have editing info
 - Creator
 - Time of creation
 - Last editor
 - Time of last edit
- Track editing info on a feature
- Enforce permissions

Demo

Editing



Functionality

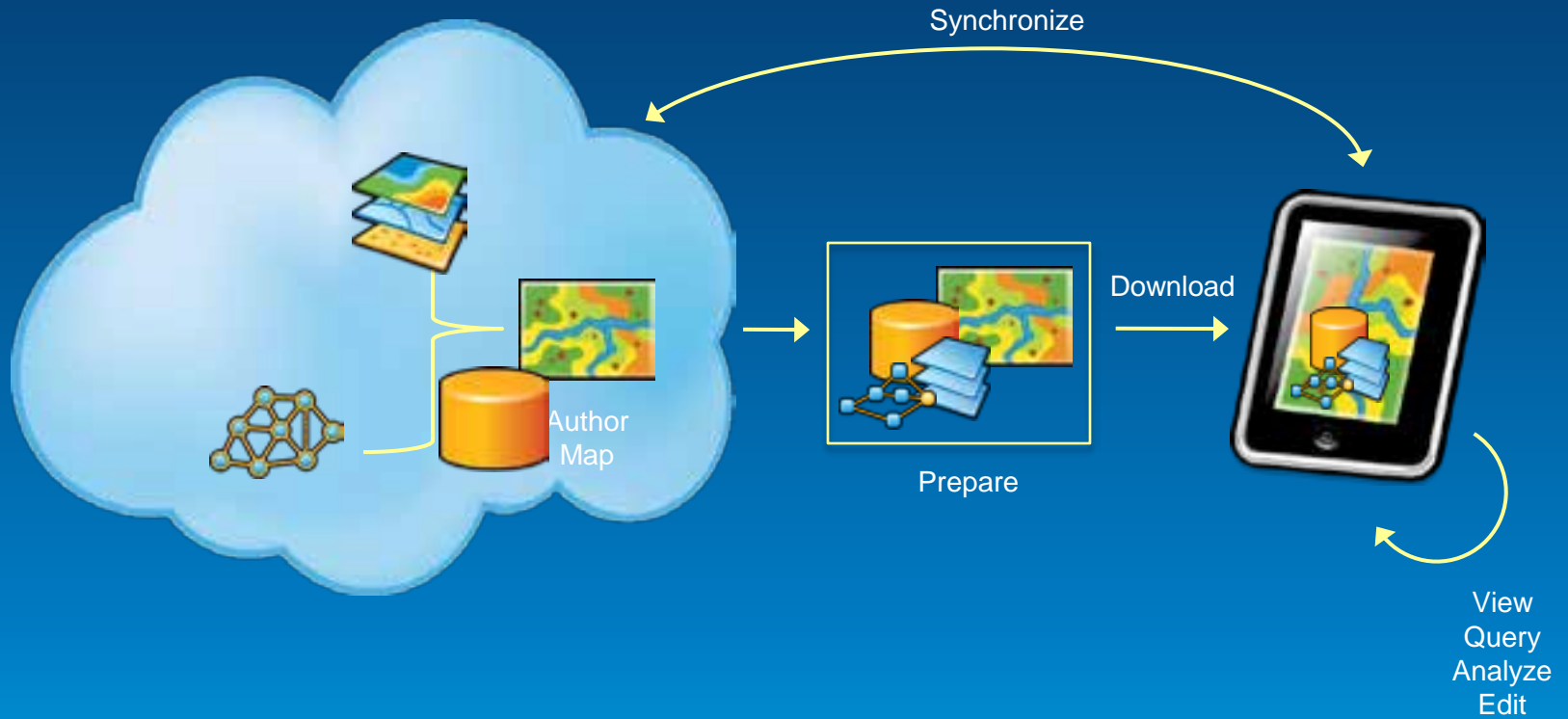
Geoprocessing	Network Analysis (Routing, etc)	Editing
Geometry Engine	GPS	Search (Geocoding, etc)
Mapping & Visualization (online and local content)		

What's next?

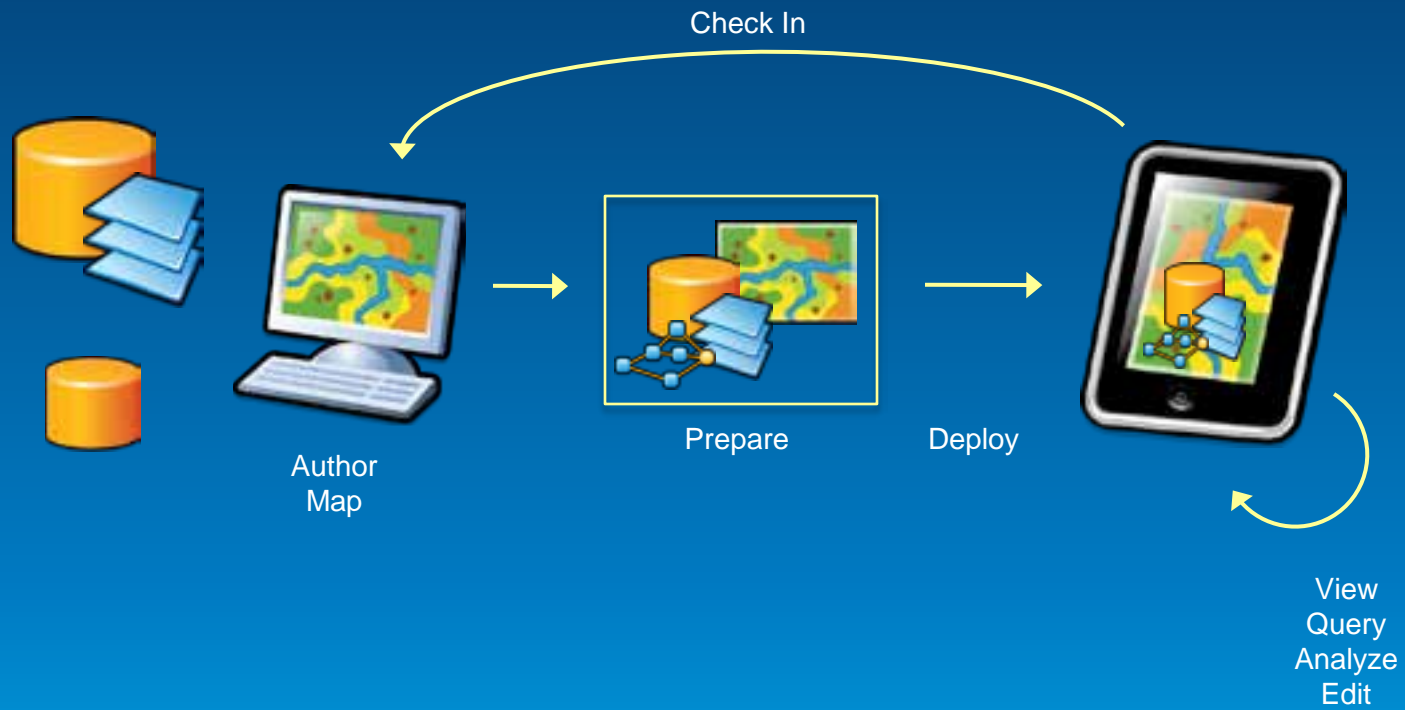
...and what we're working on...

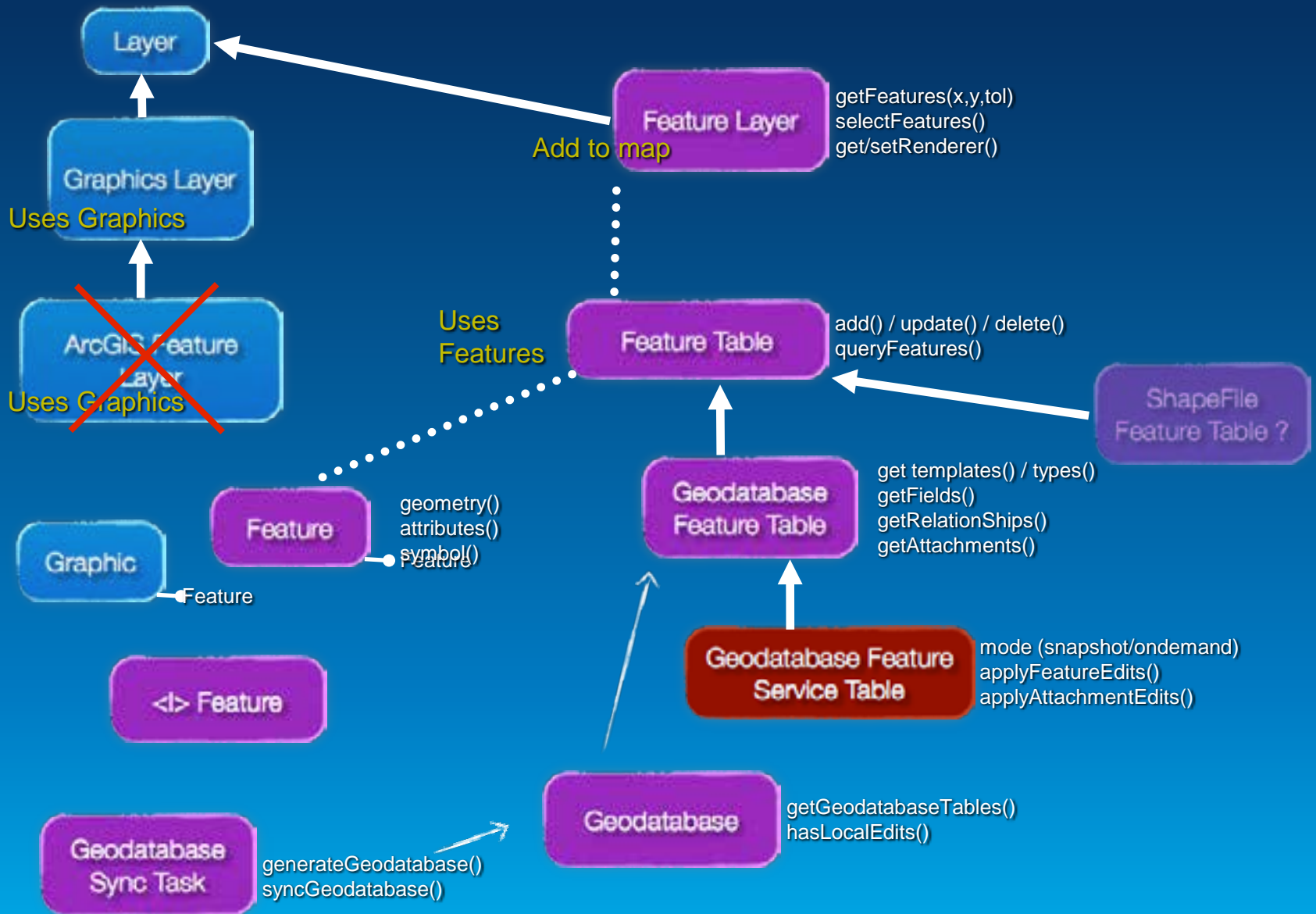
- Disconnected Map Use
- Better License support
- Data sources – direct read of rasters and vectors
- Modular Deployments
- 3D
- Support for additional Geoprocessing tools
 - Linear Referencing, Data Management, Conversion, Spatial Statistics
- Vector basemaps

From ArcGIS Online and Portal

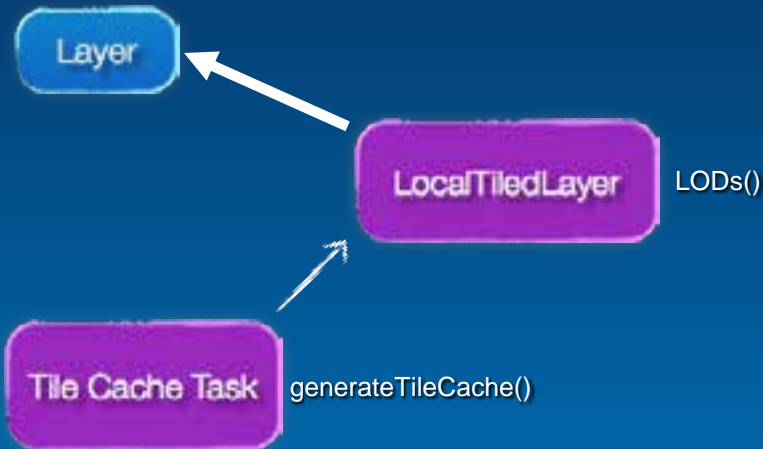


From Desktop

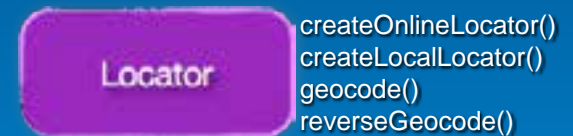




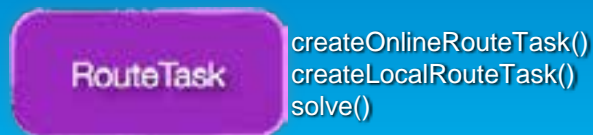
Take basemaps offline



Offline geocoding



Offline routing

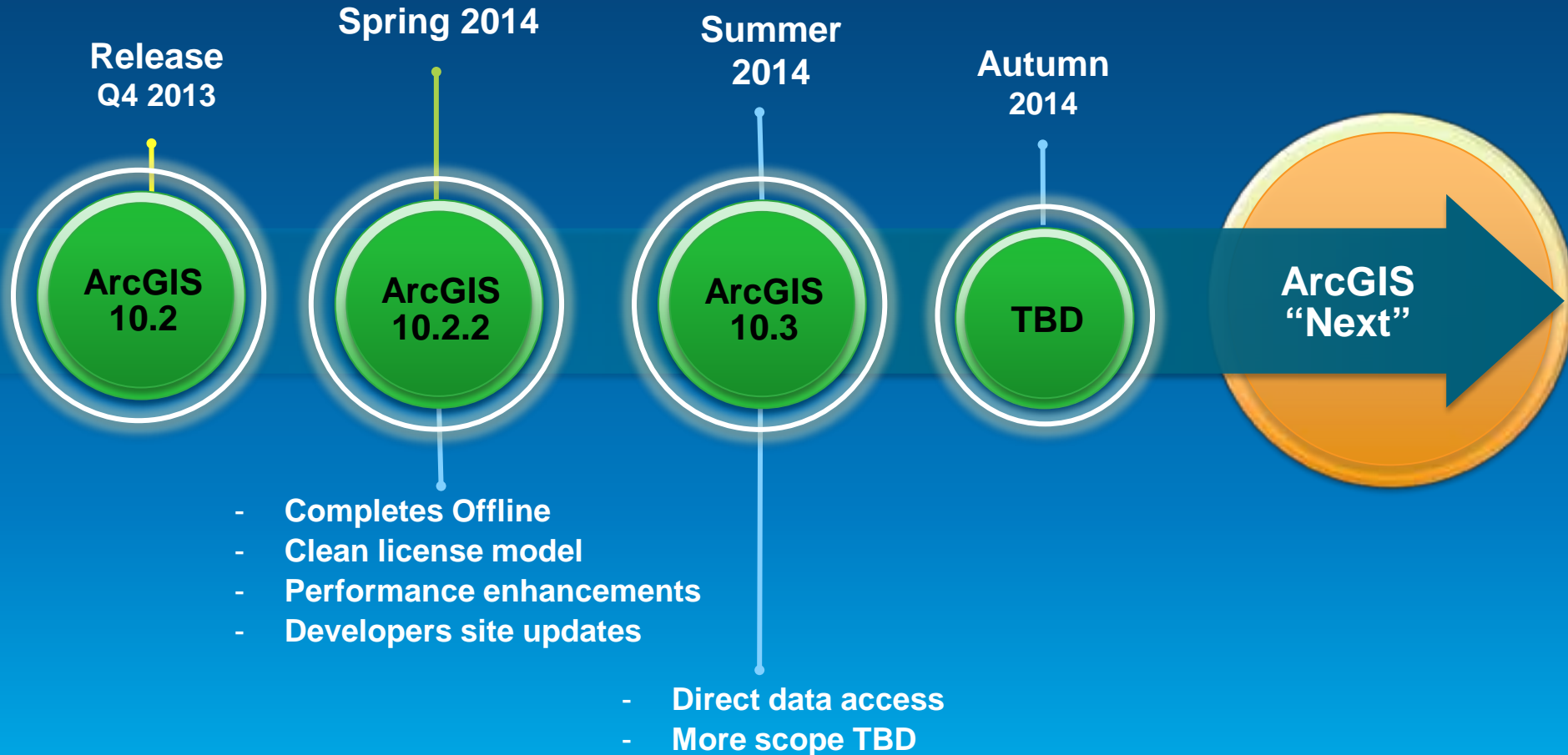


Runtime Licensing @ next release

Simple and consistent...

- Development
 - Download all SDKS - no cost
 - Basic development – no cost
 - Standard development
 - Developer OR EDN subscription
- Deployment
 - Basic – no cost
 - Standard
 - Runtime – per deployment cost
 - or
 - Cloud – subscription based (developer/user)

What's next - Release Schedule





Thank you!

Questions?



Understanding our world.