



# Qt – Developing ArcGIS Runtime Applications

Eric Bader

@ECBader

Esri Developer Summit  
Washington, DC

# Agenda

- Getting Started
- Creating the Map
- Geocoding and Routing
- Geoprocessing
- Message Processing
- Working Offline
- The Next Release
- What's Coming

# 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 enterprise system

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

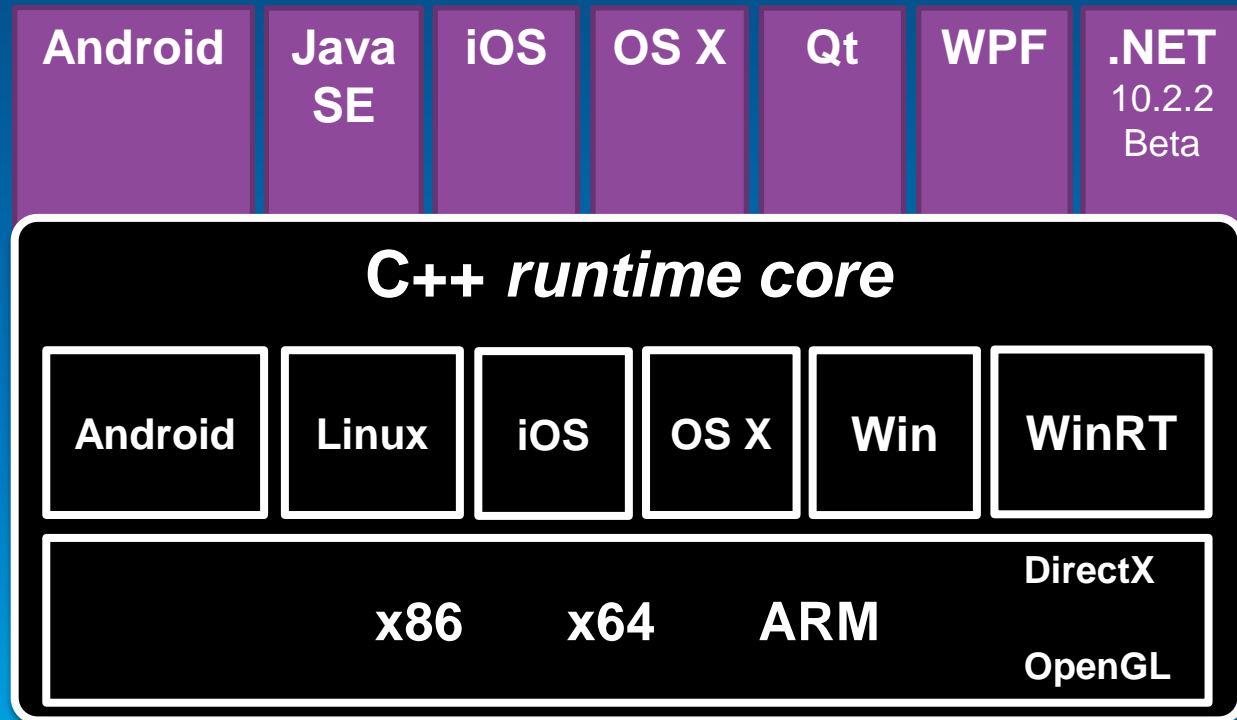


- ArcGIS for Server
  - Services



- ArcGIS for Desktop
  - Packages

# Runtime Architecture



# Qt Enables Cross-Platform Development

Easy

- Cross-platform libraries
- High-level abstractions

Portable

- Write once, run anywhere
- Builds as native C++

Open

- Pre-built platforms
- Source code

# Deep Functionality

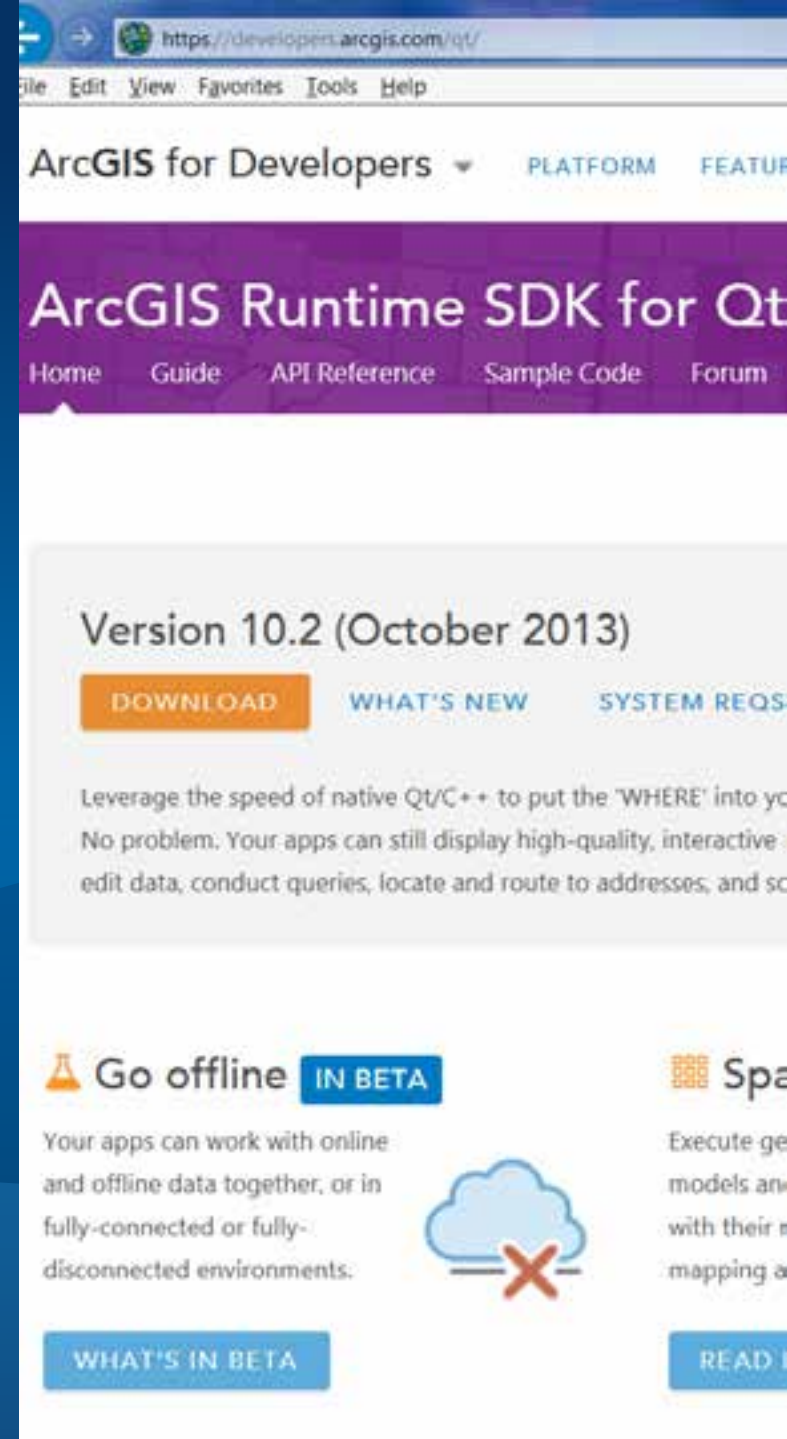


And much, much more...



Demo

# Getting Started



The screenshot shows the ArcGIS Runtime SDK for Qt website. The browser address bar displays the URL <https://developers.arcgis.com/qt/>. The navigation menu includes "ArcGIS for Developers", "PLATFORM", and "FEATURES". The main heading is "ArcGIS Runtime SDK for Qt". Below the heading, there are links for "Home", "Guide", "API Reference", "Sample Code", and "Forum". A prominent section highlights "Version 10.2 (October 2013)" with a "DOWNLOAD" button and links for "WHAT'S NEW" and "SYSTEM REQS". A descriptive paragraph states: "Leverage the speed of native Qt/C++ to put the 'WHERE' into your apps. No problem. Your apps can still display high-quality, interactive maps, edit data, conduct queries, locate and route to addresses, and so on." Below this, there are two featured sections: "Go offline IN BETA" with a flask icon and a "WHAT'S IN BETA" button, and "Spatial Analytics IN BETA" with a grid icon and a "READ MORE" button. A cloud icon with a red 'X' is positioned between the two sections.

https://developers.arcgis.com/qt/

File Edit View Favorites Tools Help

ArcGIS for Developers PLATFORM FEATURES

## ArcGIS Runtime SDK for Qt

Home Guide API Reference Sample Code Forum

### Version 10.2 (October 2013)

[DOWNLOAD](#) [WHAT'S NEW](#) [SYSTEM REQS](#)

Leverage the speed of native Qt/C++ to put the 'WHERE' into your apps. No problem. Your apps can still display high-quality, interactive maps, edit data, conduct queries, locate and route to addresses, and so on.

#### Go offline **IN BETA**


Your apps can work with online and offline data together, or in fully-connected or fully-disconnected environments.

[WHAT'S IN BETA](#)

#### Spatial Analytics **IN BETA**

Execute geospatial models and workflows with their native mapping and visualization capabilities.

[READ MORE](#)





# Platform Support

ArcGIS Runtime SDK for Qt

- Operating Systems (development and deployment)
  - Windows 7, 8, 8.1 Server 2008 and 2012
  - Red Hat 6.2
  - Ubuntu 12.04
- Windows SDK: 64-bit only (32-bit coming 2014)
- Linux : 64-bit and 32-bit

# Getting Started

- Get Qt SDK
  - 5.1.1 minimum.
- Get Qt Creator
  - Version 2.7.x or higher recommended

(VS 2012 also supported)



# Licensing Model @ 10.2

## Development:

- Get a free developer subscription: [developers.arcgis.com](http://developers.arcgis.com)
- Download and install SDK
- Get free SDK authorization code from [developers.arcgis.com](http://developers.arcgis.com)
- Build!

*Notice: EDN subscription is not required*

## Deployment: 2 levels of licensing:

- **Basic** (free) and **Standard** (purchase deployment packs)
- Contact Customer Service for license strings
- Compile license into your app.

# Hierarchy of Map Layers



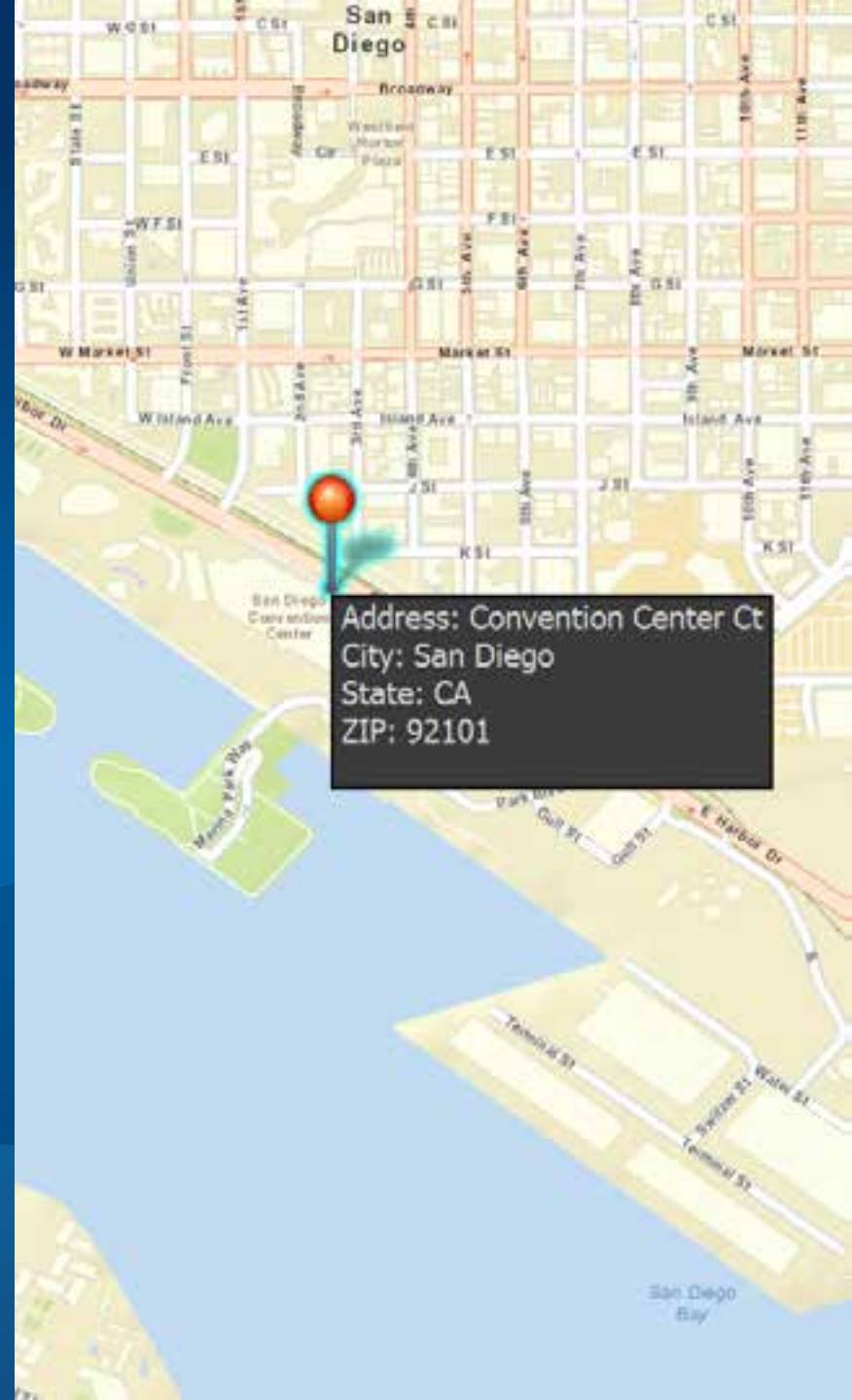
- Layers are added bottom-to-top
- Graphics layers: live / temporary data:
  - Vehicles, people, events
- Operational layers: dynamic features
  - Facilities, buildings, zones, networks
- Basemap layer: spatial context
  - Imagery, topography

# Provisioning Content

- **Online Services** accessed via a URL
  - Tiled layer, Features, Geoprocessing, Geocoding
- **Local Content** authored as packages and installed locally
  - Map Packages: Editing, querying
  - Tile Packages: High performance base layer
  - Geoprocessing Packages: Models and Scripts
  - Geocoding Packages

Demo

# Geocoding



Address: Convention Center Ct  
City: San Diego  
State: CA  
ZIP: 92101

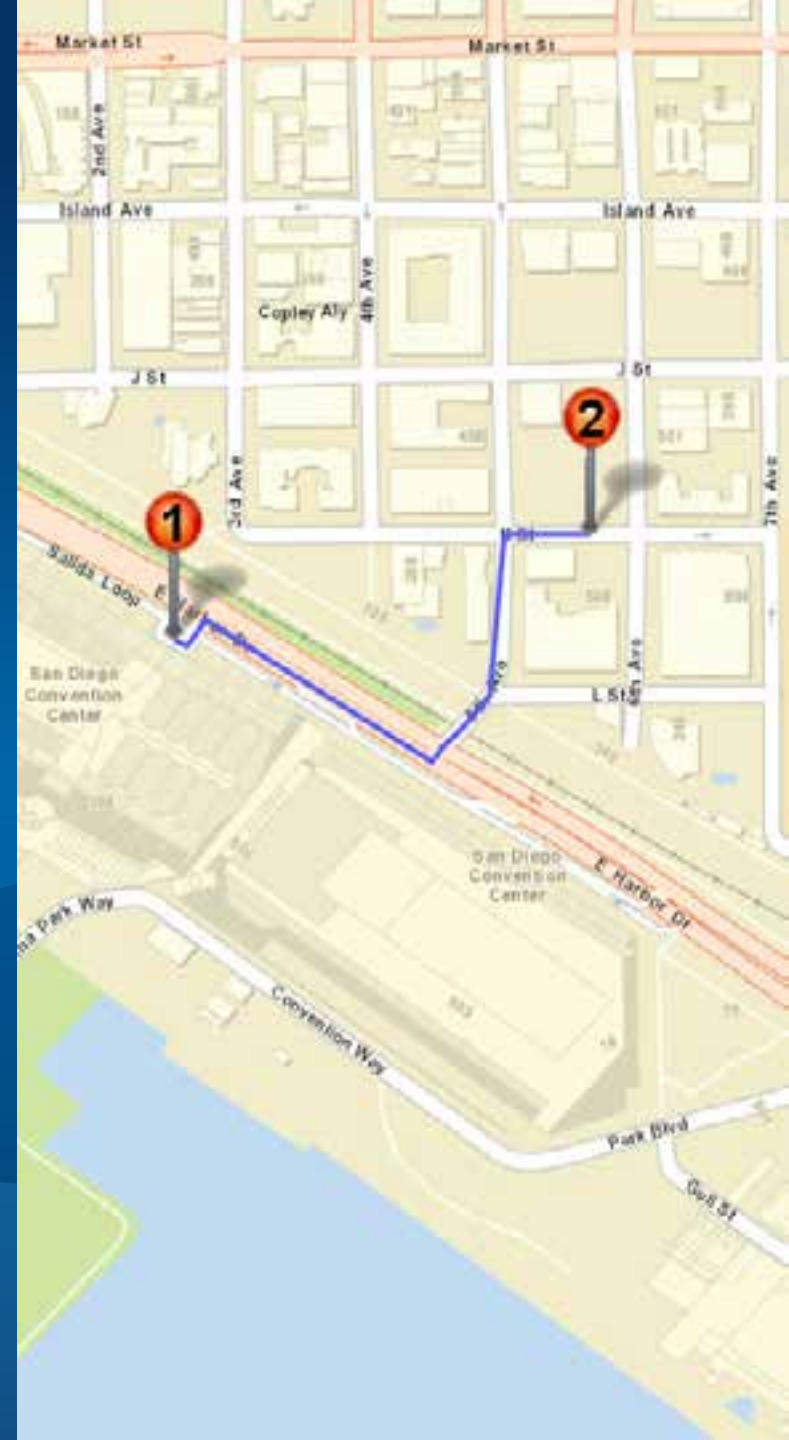
# Geocoding and Routing use a Task-Based Framework

- Workflow
  - Construct a task object
  - Provide parameters
  - Connect slots for completion signals
  - Execute task ( for geocoding, Locator::find() )
  - Grab results in slot when task completes
- Other Runtime tasks
  - Query & identify features
  - Find closest facility or service area (network analysis)

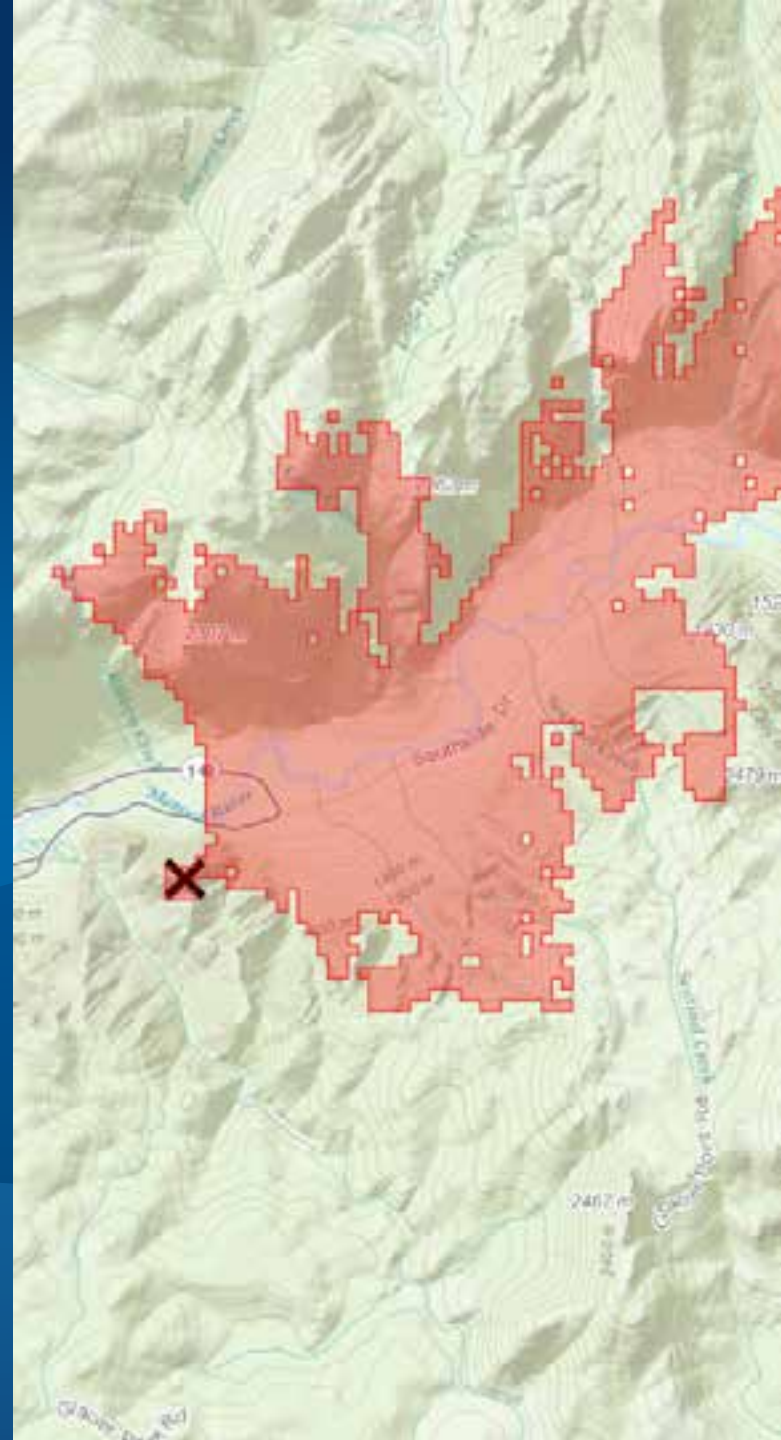


Demo

# Routing



# Geoprocessing

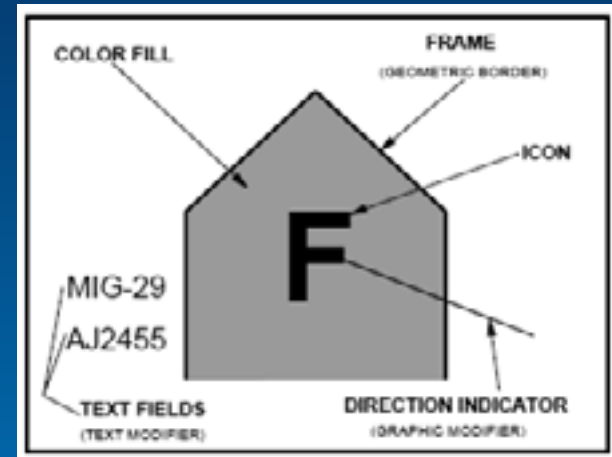


# Geoprocessing

- Geoprocessing is the source of advanced GIS analysis in the ArcGIS Runtime
  - Functionality that goes beyond the Runtime API
- Connected
  - ArcGIS Server's and ArcGIS Online's Published Services
- Disconnected
  - Local geoprocessing from Geoprocessing Packages (GPKs)

# Message Processor

- Displays command and control symbols
  - Message communicates a symbol's type, location
  - MP displays the symbol on a special graphics layer
- Works with standard symbol dictionaries
  - MIL-STD-2525C, App6B
- Makes it easy to display these symbols

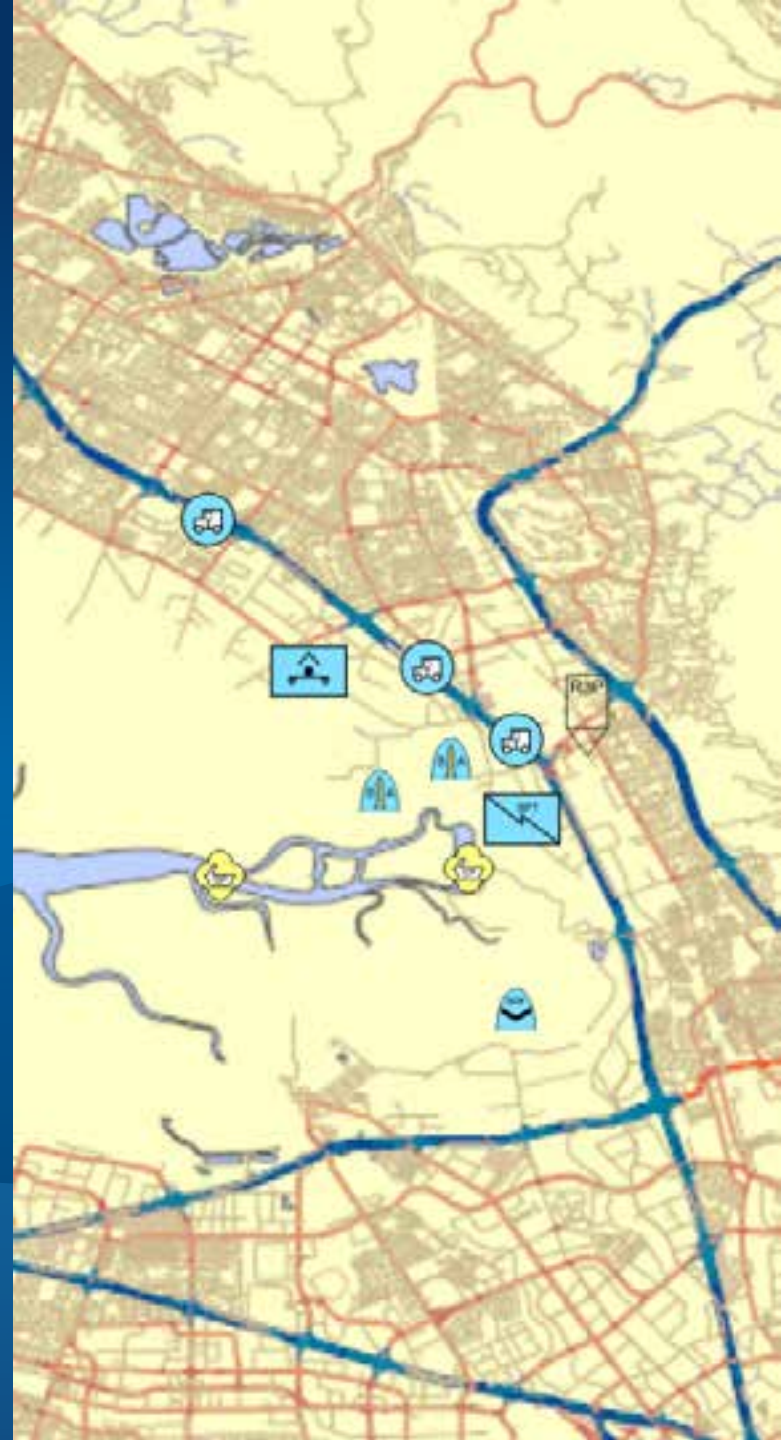


# Using the Message Processor

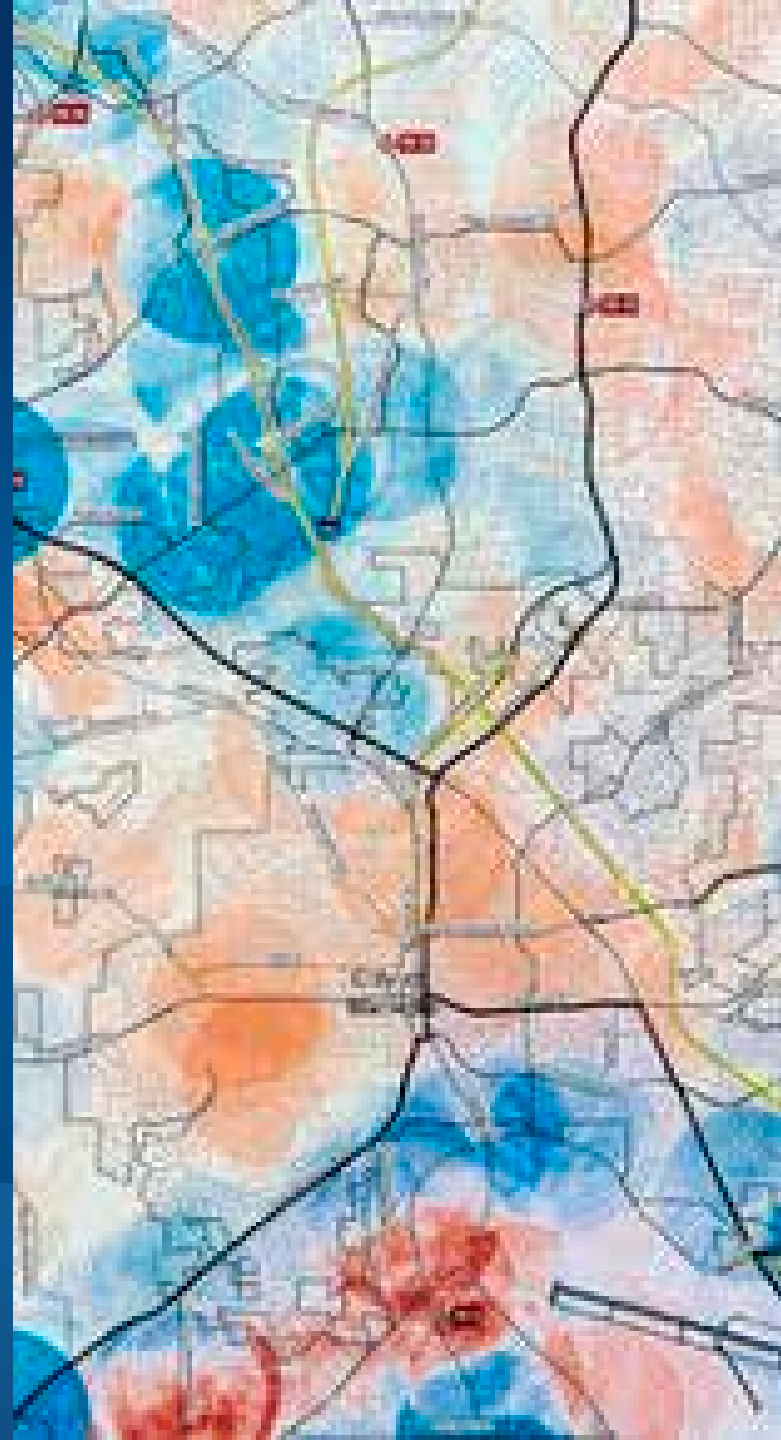
- Create **MessageGroupLayer**, add to map
- Get **MessageProcessor** that was created within the MessageGroupLayer
- For each message received from outside source
  - Construct Message object from message content
  - Pass Message to the MessageProcessor
  - MessageProcessor adds / updates / deletes symbol in MessageGroupLayer

Demo

# Message Processing and Geoprocessing



# Working Offline





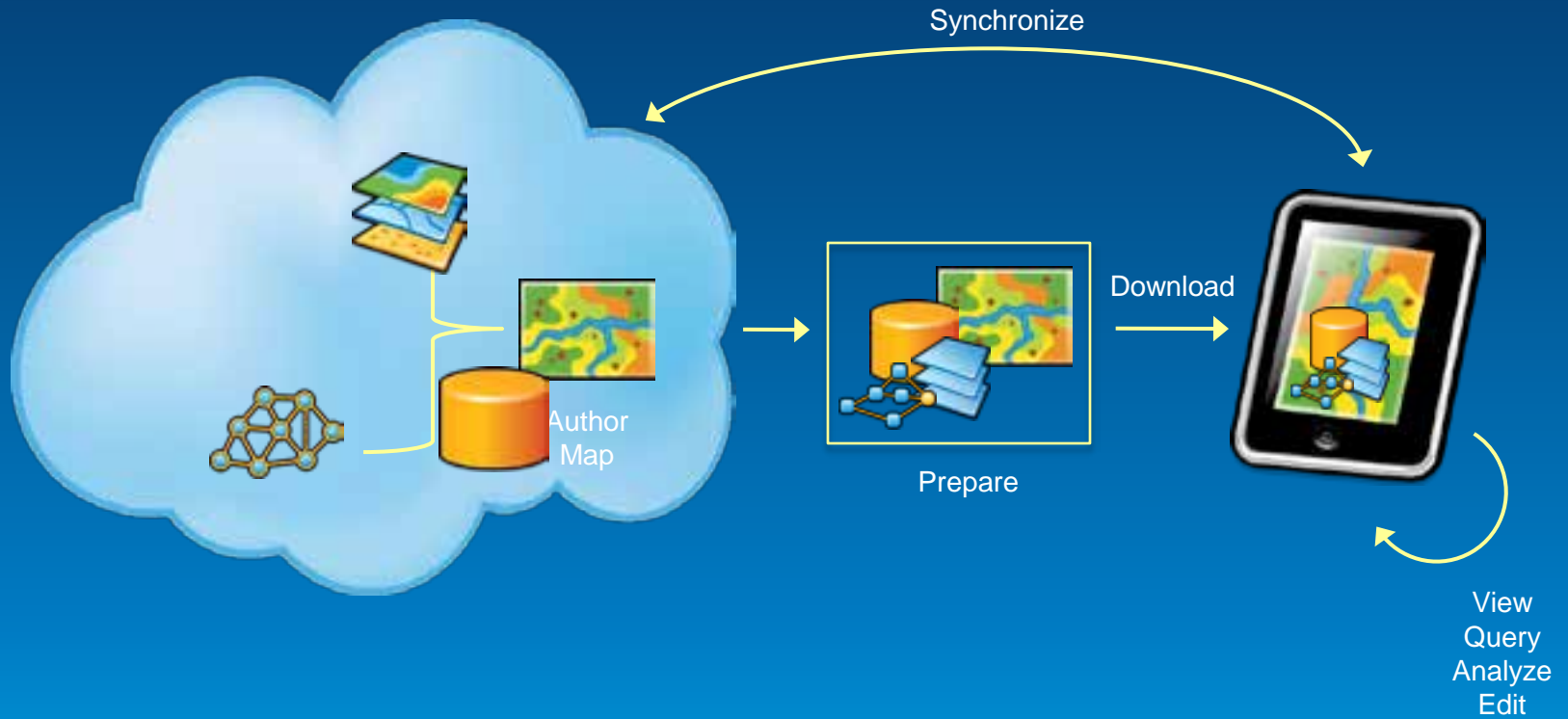
# Offline ArcGIS

- **All** Runtime SDKs
  - Licensing - “Standard” level functionality
- Disconnected use of the ArcGIS platform
  - Viewing/interacting with maps
  - Querying data
  - Find places and locations
  - Get directions
  - Editing features
    - Synchronization with service

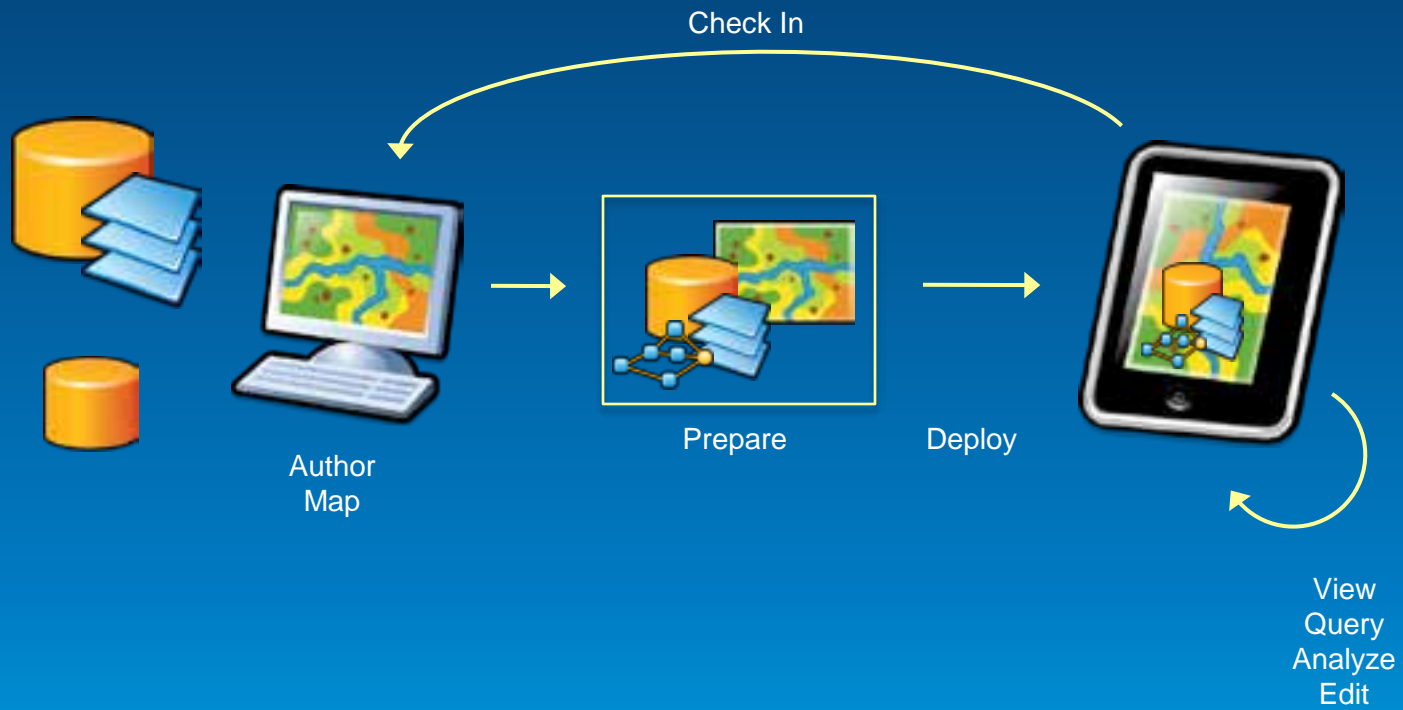
# DEMO...

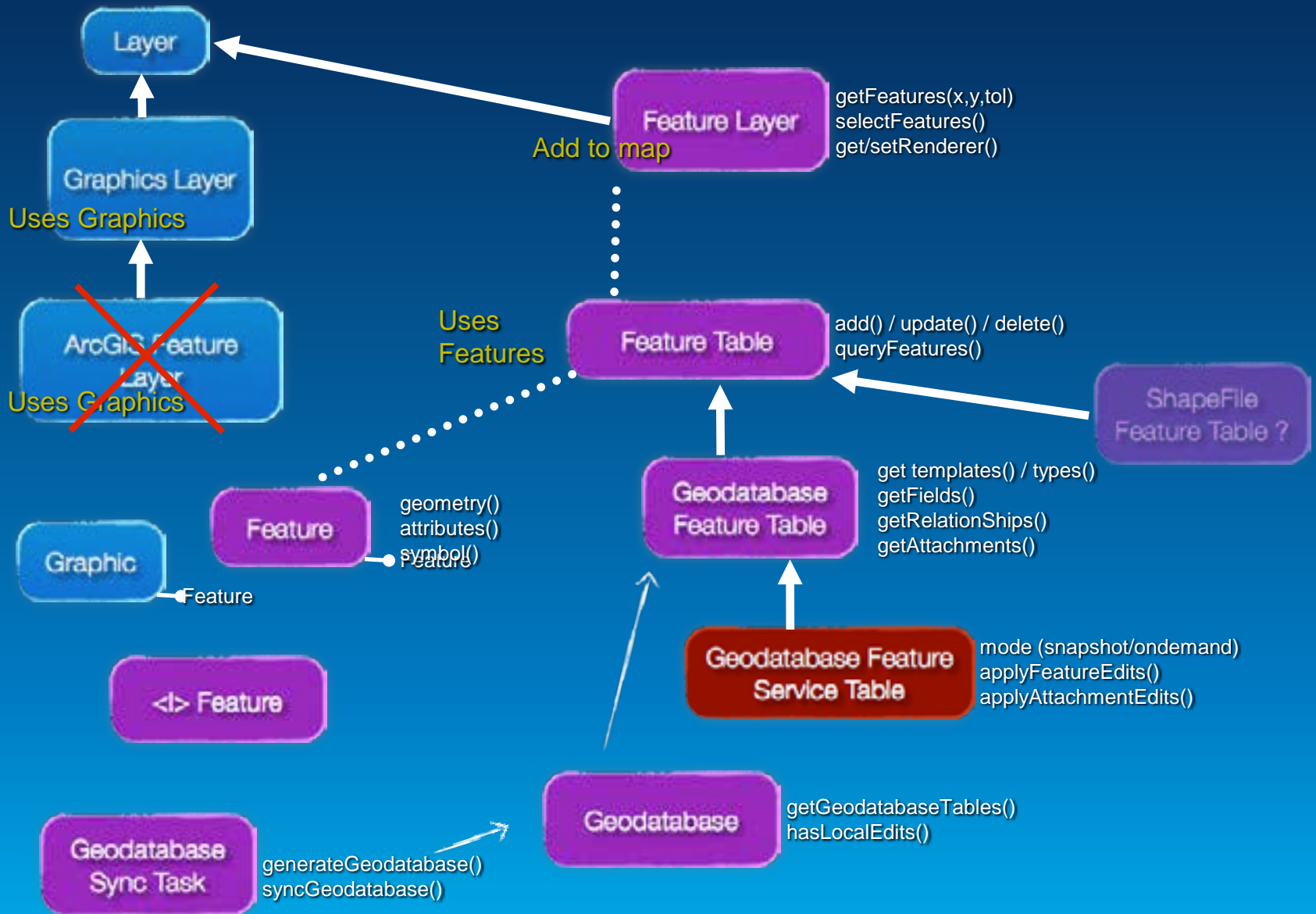
- Editing
- Geocoding
- Routing
- Tile Cache

# From ArcGIS Online and Portal

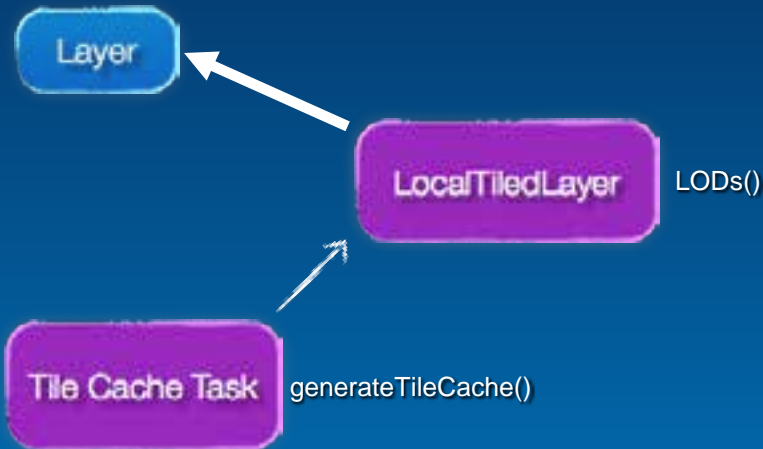


# From Desktop





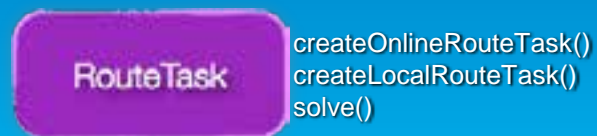
## Take basemaps offline



## Offline geocoding



## Offline routing



# Offline Geodatabase Editing

- Add a feature to the database

```
QMap<QString, QVariant> attrs;
attrs.insert("SymbolID", "100");
attrs.insert("EventType", addType);
attrs.insert("EventDate", QDateTime::currentMsecsSinceEpoch());
attrs.insert("Description", "Fire reported");

EsriRuntimeQt::Feature f(pt, attrs);
int uid = -1;
try
{
    uid = m_featureLayer.featureTable().addFeature(f);
}
catch (EsriRuntimeQt::Exception e)
{
    qDebug() << "Exception: Adding feature error: " << e.what();
}
```



# Syncing your Geodatabase Edits

- Prepare **SyncGeodatabaseParameters**:
  - Can you update?
  - Can you edit?
  - Upload only? Bidirectional?
- Call **submitSyncJobAnd ApplyResults** on the GeodatabaseTask

# What's coming in Runtime

Runtime  
SDK



## 10.2.2 release – End of March 2014

- Offline API is completed
- Improved Licensing models
- Developer productivity
- Display improvements

# Licensing @ 10.2.2

## Development:

- Get a free developer subscription: [developers.arcgis.com](https://developers.arcgis.com)
- Download SDK and build
- Create a client ID, embed it into your app

## Deployment: 2 levels of licensing:

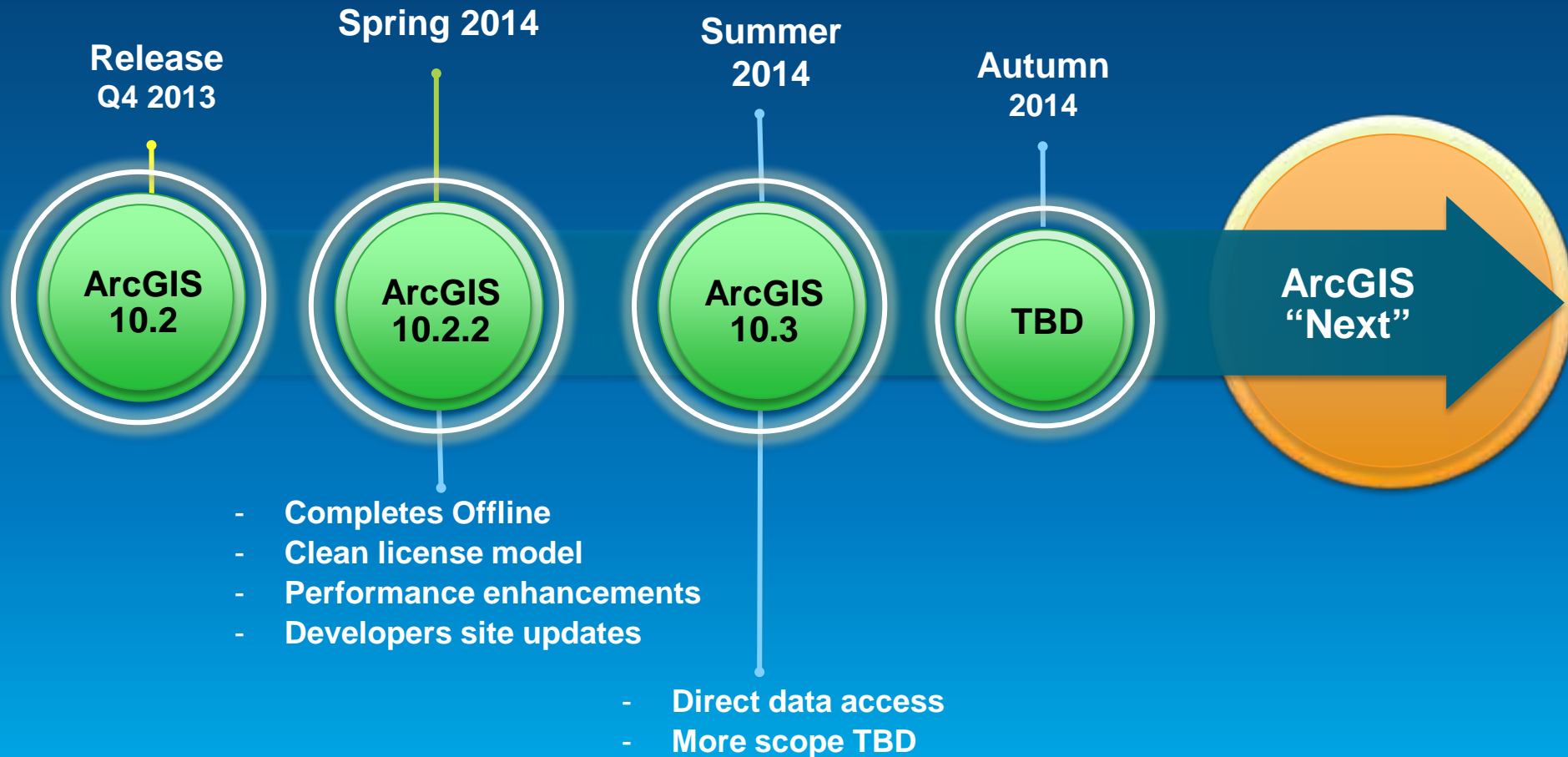
- **Basic** (free) and **Standard** (purchase, or named user)
- Basic app: contains client ID only
- Standard app: contains client ID + (license string OR named user credentials)

# What are we working on?

- QML as an API + new platforms
- Architecture
- Mapping
- 3D
- Data Formats
- Geometries
- ...more...but I can't say.



# What's next - Release Schedule



*Thank you!*



Questions?





Understanding our world.