



Esri International Developer Summit  
Palm Springs, CA

# Getting Started with ArcGIS Runtime SDK for Qt

Thomas Dunn & Nandini Rao

# Agenda



**Getting Started**

**Creating the Map**

**Geocoding and Routing**

**Geoprocessing**

**Message Processing**

**Work Offline**

**The Next Release**

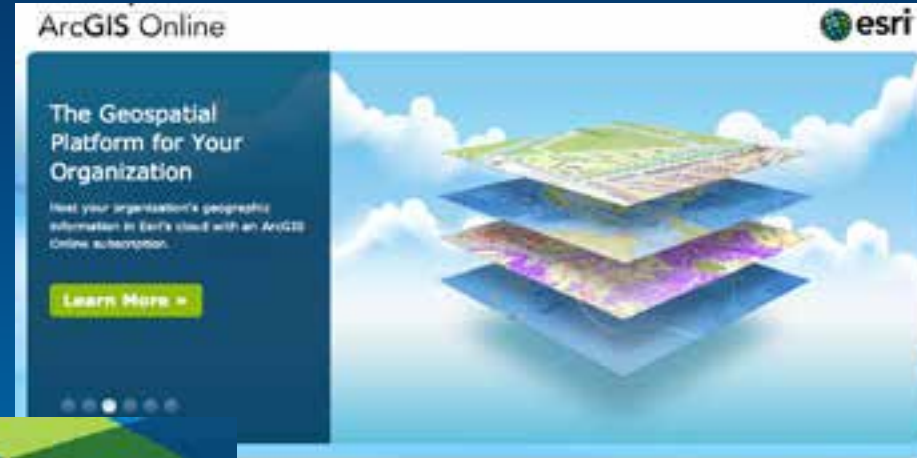
# Runtime

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



# Runtime is part of the ArcGIS enterprise system

- ArcGIS Online & ArcGIS for Portal
  - Maps, services, content and Organization branding



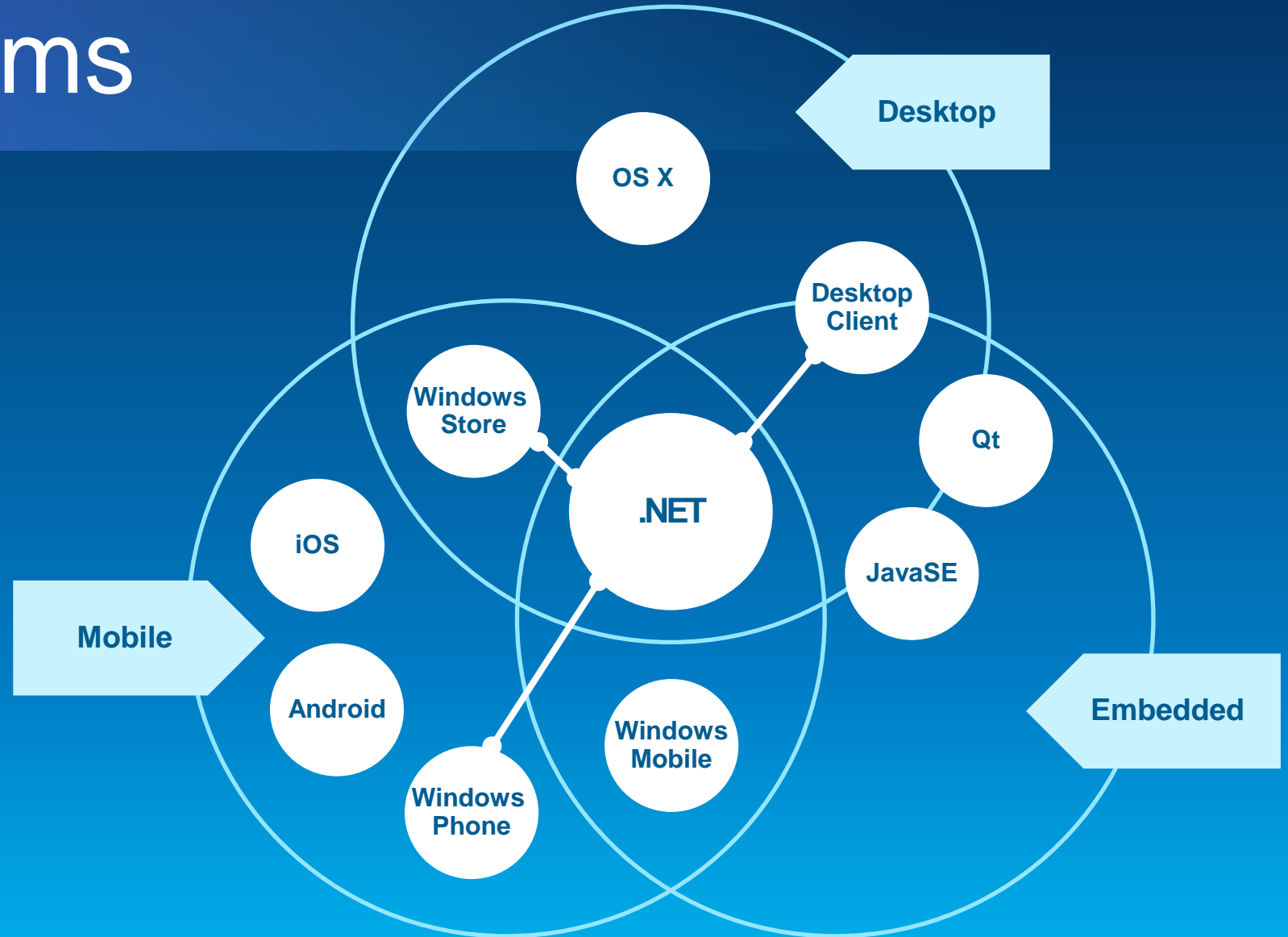
- ArcGIS for Server
  - Services



- ArcGIS for Desktop
  - Packages



# Target Platforms



# 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

# Extensive cross-platform development library



And much, much more...

Demonstration

# Getting Started

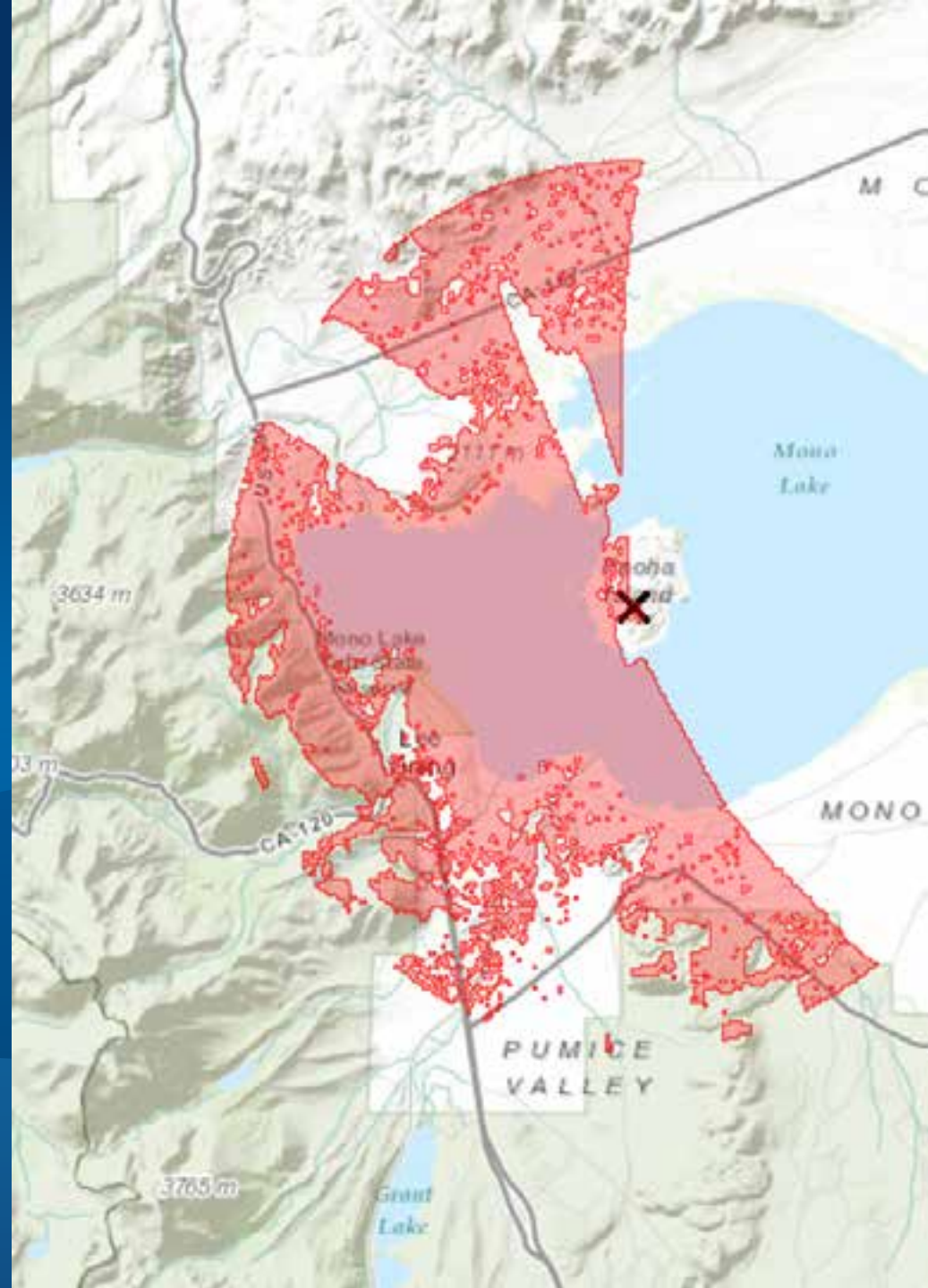




# System Requirements

- **Operating Systems (development and deployment)**
  - Windows 7, 8 and 8.1, Server 2008 R2, 2012 and 2012 R2
  - Red Hat 6.2
  - Ubuntu 12.04 LTS
  - 32 or 64-bits
- **Qt SDK from <http://qt-project.org>**
  - Version 5.1.1 for Windows or Linux
- **Qt Creator IDE**
  - Version 2.7 or higher

# Creating a map



# Map Layers

Operational layers: dynamic features  
Facilities, buildings, zones, networks

Layers are added to the map in  
order, bottom-to-top

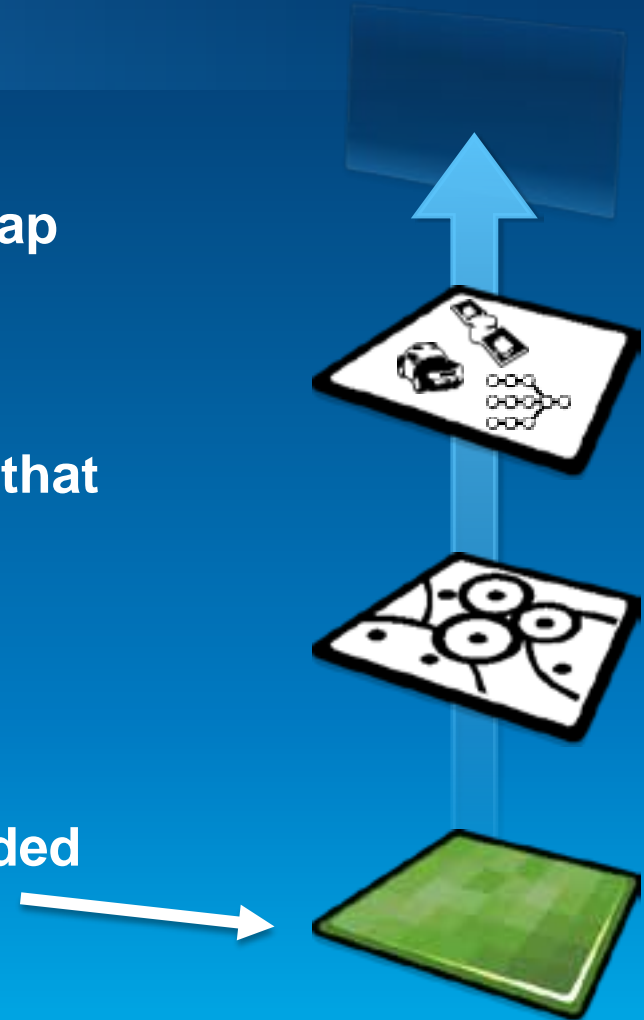


Graphics layers: live / temporary data:  
Vehicles, people, events

Basemap layer: spatial context  
Imagery, topography

# Spatial Reference

- **Decide on the spatial reference to be used by your map**
  - e.g. Web Mercator Auxiliary Sphere, WGS-84, UTM ...
- **Use a tiled service or tile package for your base map that uses that spatial reference**
  - Tiled basemaps cannot be re-projected
- **Spatial reference of map determined by first layer added**



# Provisioning Content

- **Online Services:** access these via a URL
  - Tiled services, dynamic map services, feature services, Geoprocessing services, Geocoding services
- **Local Content:** author packages and deploy them with your app
  - Tile Packages as a high performance base map layer
  - Map Packages for querying or editing
  - Geoprocessing packages built from ArcGIS models or Python scripts
  - Geocoding locator packages or locator files



# Geocoding

Santa Claus  
North Pole



Where is this address?  
(Geocoding)



Van Ittersumstraat 14,  
9621 CV Groingen,  
The Netherlands

(reverse geocoding)

# Reverse Geocoding

# Task-Based Framework

1. **Construct a task object**
2. **Provide parameters**
3. **Connect slots for completion signals**
4. **Execute task**
5. **Grab results in slot when task completes**

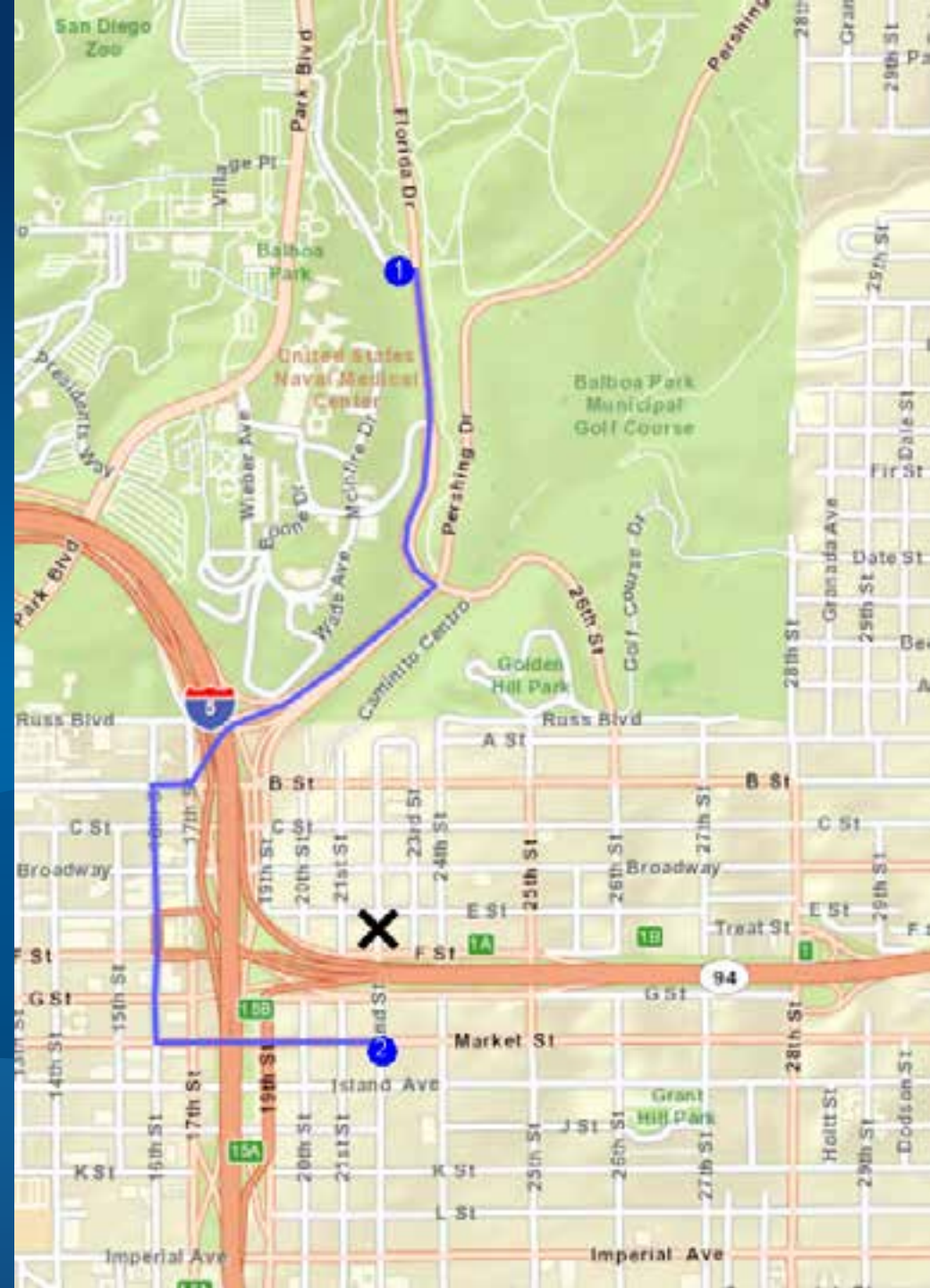
## Runtime tasks:

- **Locator (geocoding)**
- **Routing**
- **Geoprocessing**
- **Geodatabase sync**
- **Find**
- **Query**
- **Identify features**
- **Find closest facility**
- **Calculate service area**
- **Tile cache generation**



Demonstration

# Geocoding and Routing

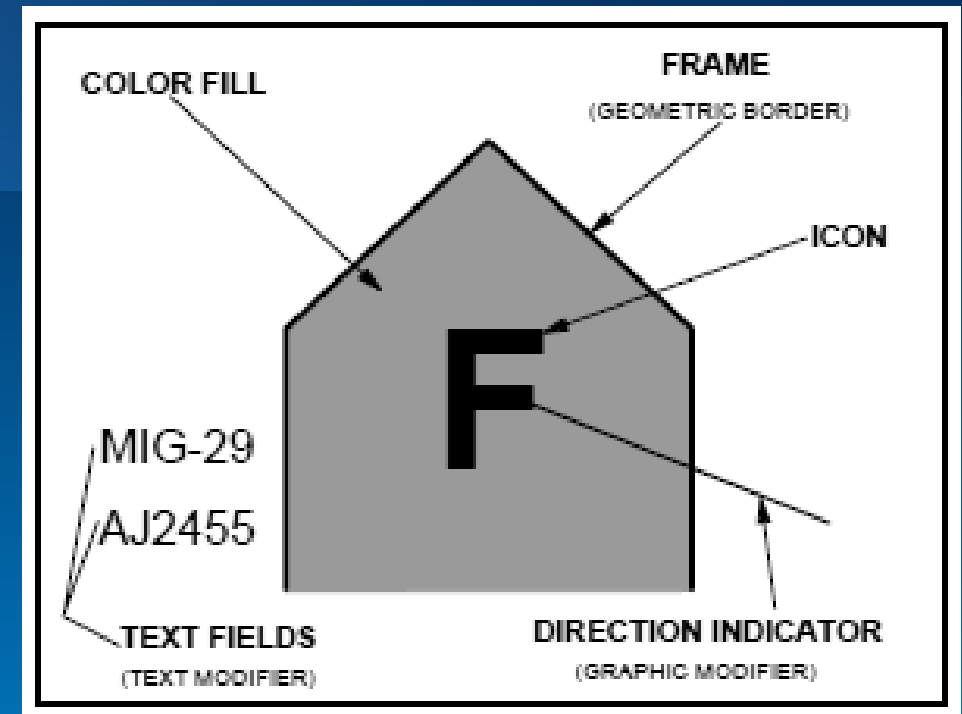


# Geoprocessing

- **Geoprocessing provides advanced GIS analysis in Runtime**
  - See Qt Sample Application for some samples
  - <https://developers.arcgis.com/qt/guide/supported-geoprocessing-tools.htm>
- **Use one tool, or build a Model or Python script to chain tools**
- **Display results on map or use in subsequent processing**
- **Connected**
  - ArcGIS Server and ArcGIS Online geoprocessing services
- **Disconnected**
  - Local geoprocessing service from geoprocessing packages (GPKs)

# Message Processor

- **Streamline display of command and control symbols**
  - Message communicates a symbol's type, location and status
  - Message Processor displays the symbol on a special graphics layer
  
- **Works with standard symbol dictionaries**
  - MIL-STD-2525C
  - App6B



*Drawing is from MIL-STD-2525C dated 11/17/2008*

# Using the Message Processor

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

# Work Offline

- **Download and view basemaps from Esri or your own authoritative basemaps**
- **Edit data from feature services (hosted in the cloud or on premises) and sync changes back**
- **Query and analyze your operational data locally**
- **Search for places (geocode and reverse geocode) using your organization's spatial data assets**
- **Find optimal routes using your organization's network dataset**

Offline features that are Beta at 10.2 are final in 10.2.2

Demonstration

# Work Offline



# Development and Deployment Workflow



1. Download and Install



2. Develop and Test



3. Deploy and Distribute

# License levels and functionality

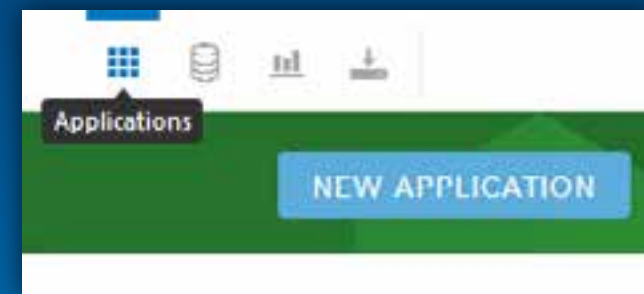
License Level	Available functionality
Developer (development and testing only)	All functionality including Local Server extensions (watermarks and debug messages will be generated, nag screens with Local Server)
Basic	All functionality, <u>except</u> : <ul style="list-style-type: none"><li>• Local locators (geocoding)</li><li>• Local routing</li><li>• Local geodatabase editing</li><li>• Local geodatabase sync operations with an upload</li><li>• Local Server</li></ul>
Standard	All functionality. Note: Local Server extension licenses are additional and require the Standard license.



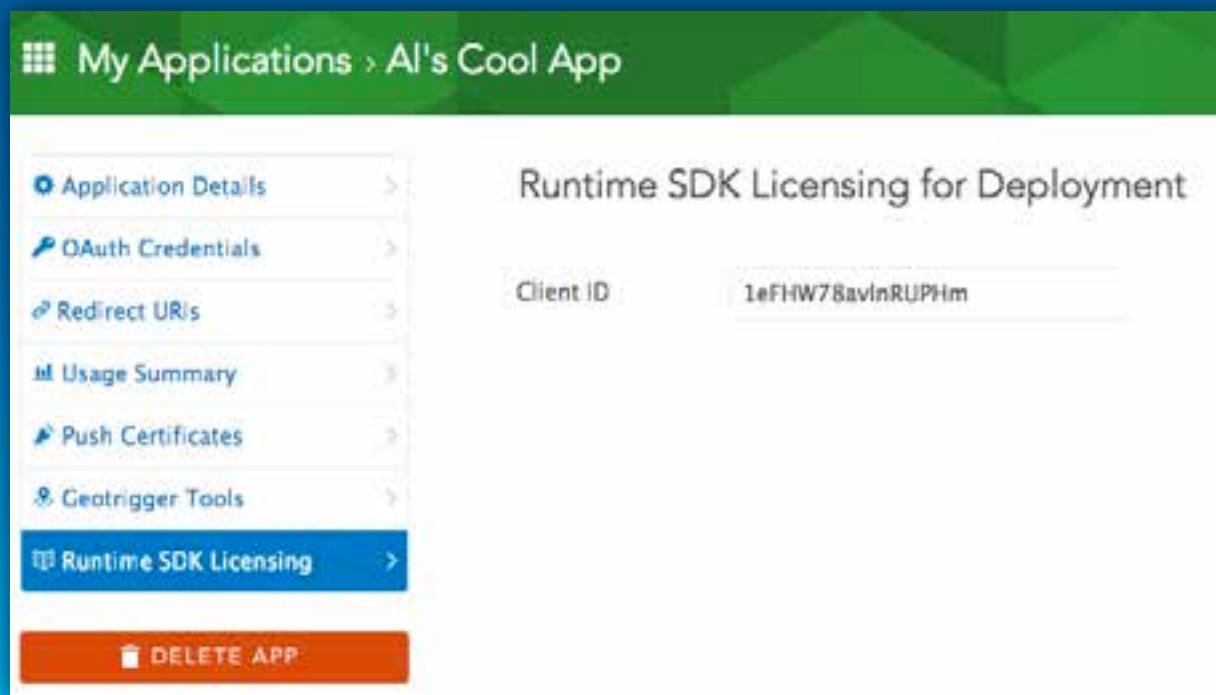
# License your app at Basic level

1. Go to [developers.arcgis.com](https://developers.arcgis.com) and log in (or create a developer account)
2. Create a New Application (or select existing)
3. Click on Runtime SDK Licensing
4. Copy the Client ID and set it in your app

2.



3.



4.

```
// set the client ID  
ArcGISRuntime::setClientId("myclientid");
```

# License your app at Standard level

There are two options to license at Standard level:

1. Use an **organization account** (ArcGIS Online or Portal for ArcGIS)

- Requires users of your app to log in with their account

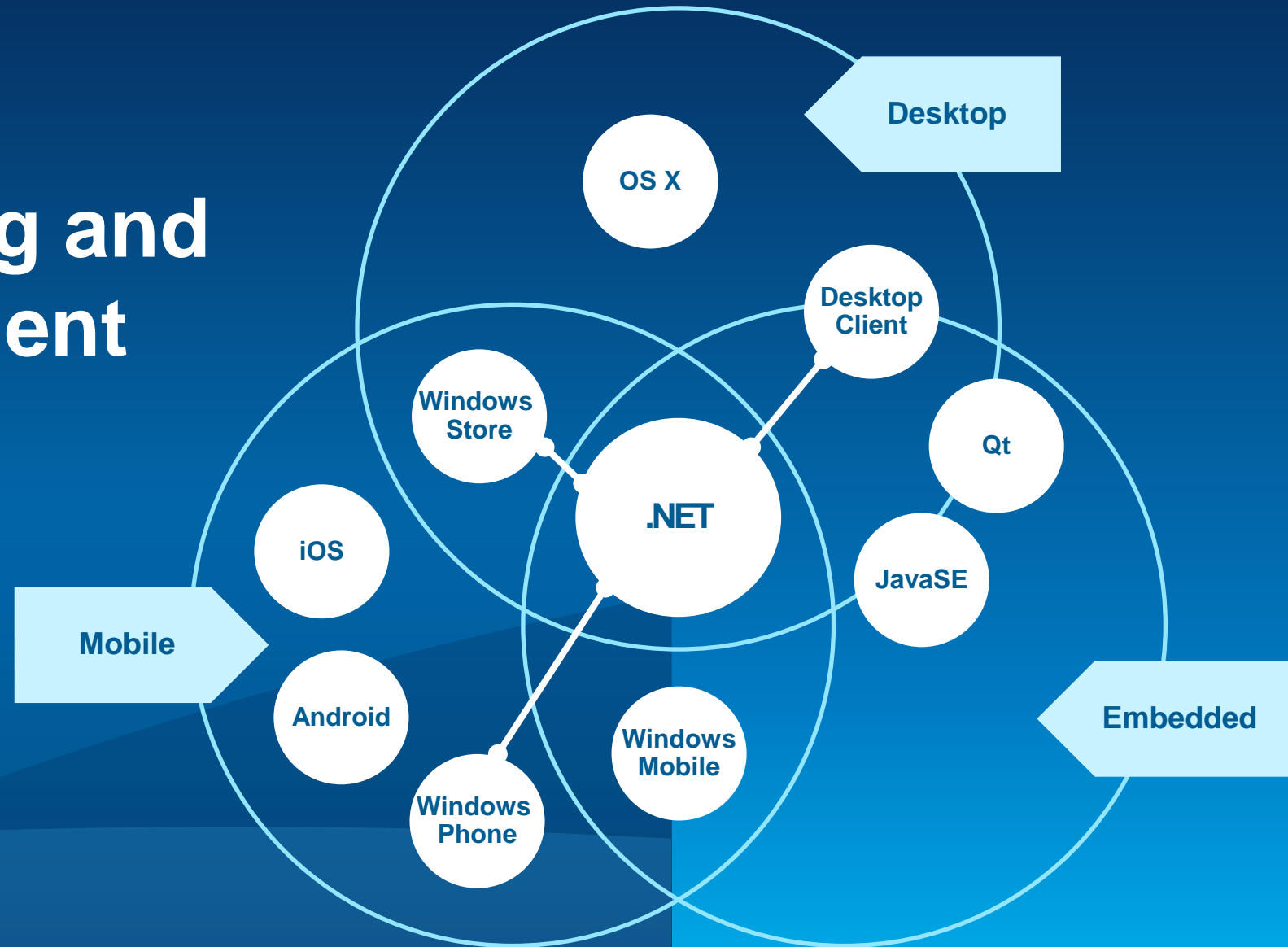
2. Use a **license string** obtained from Customer Service or your international distributor

- License string burned into the app
- Extensions can be supported

**\*\* You must use workflow 2 if you want to license any extensions \*\***

For more info speak to sales or product management

# Licensing and Deployment



# What's next for Runtime?



# Next Release

- **Direct read of raster datasets\***
- **Direct read of vector data (KML, Shapefiles)**
- **3D\***
- **Toolkits**
- **Common conceptual model**
- **More offline capabilities**
  - **Feature service table**
  - **Versioned data support**

\*Some SDKs will release items before others



# Related sessions



Session Name	Time	Location
Building Qt Apps with ArcGIS Runtime SDK	Wednesday 2:30pm – 3:30pm	Smoketree F
Building Offline Apps with ArcGIS Runtime SDK – Part 1	Wednesday 4:00pm – 5:00pm	Primrose B
Building Offline Apps with ArcGIS Runtime SDK – Part 2	Wednesday 5:30pm – 6:30pm	Primrose B
The Road Ahead: ArcGIS Runtime SDKs	<b>Thursday</b> 8:30am – 9:30am	Primrose A
Everything (or Anything) You Wanted to Know about the ArcGIS Runtime SDKs	<b>Thursday</b> 10:00am – 11:00am	Primrose A

<http://www.esri.com/events/devsummit/session-rater>

Getting Started with ArcGIS Runtime SDK for Qt



Understanding our world.