ESRI DEVELOPER SUMMIT

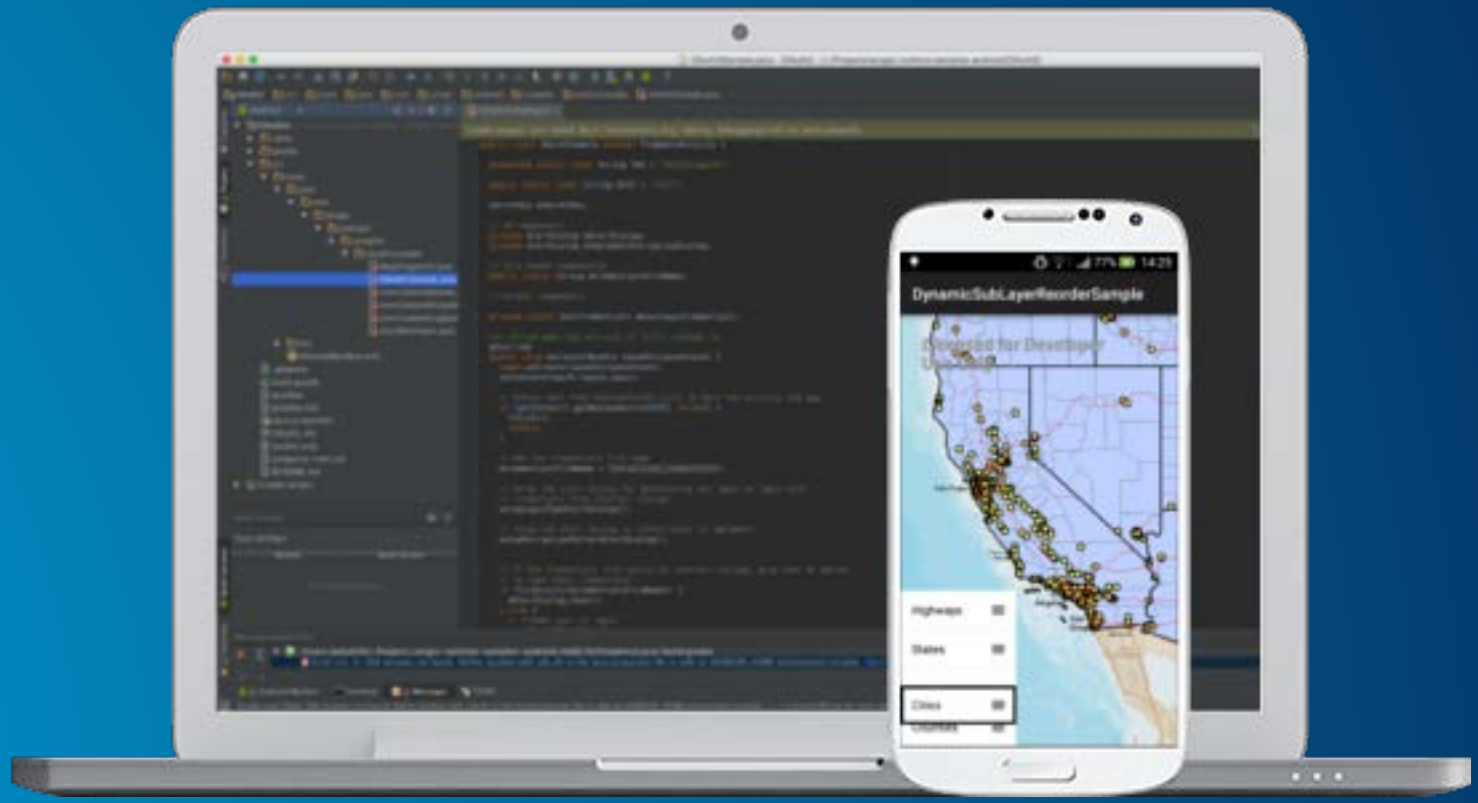
10-12 November | Berlin, Germany

ArcGIS Runtime SDK for Android: Building Apps

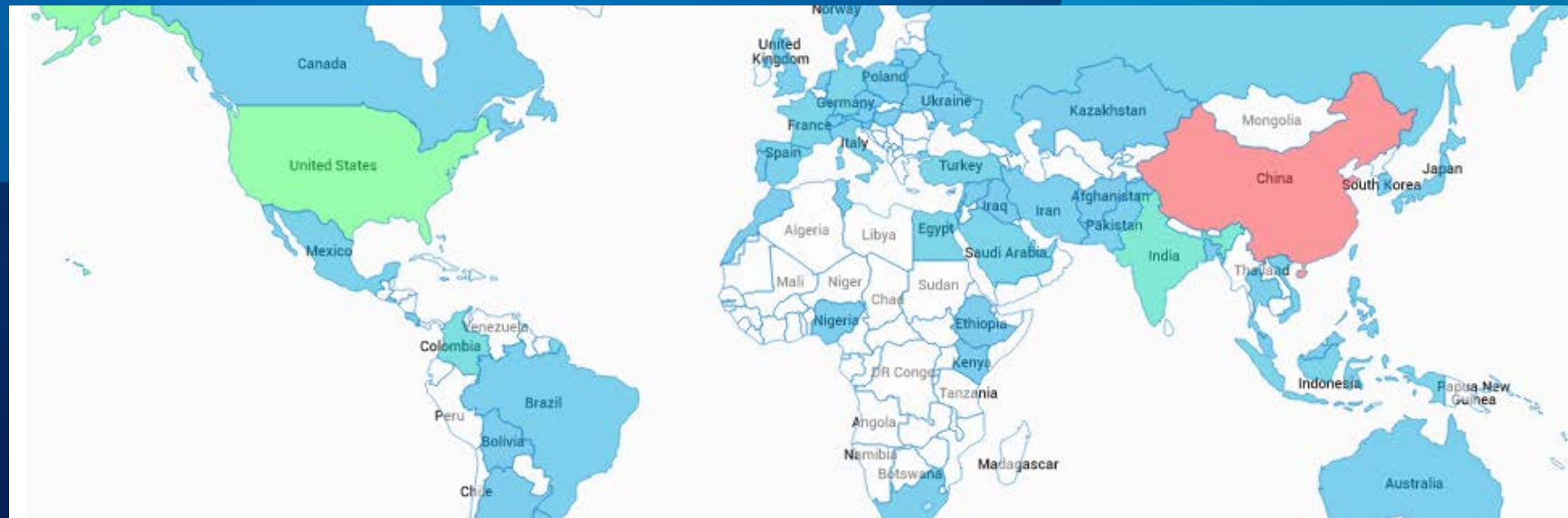
Alan Lucas
Mark Baird

What's in this session?

- Getting started in 5 minutes
- Guide to common workflows
  - Displaying current location
  - Finding places
  - Identifying features
  - Editing
- What about Quartz



Getting started



Getting started

1. Install Android Studio http://developer.android.com/sdk
2. Get dependencies automatically with gradle and try it out!

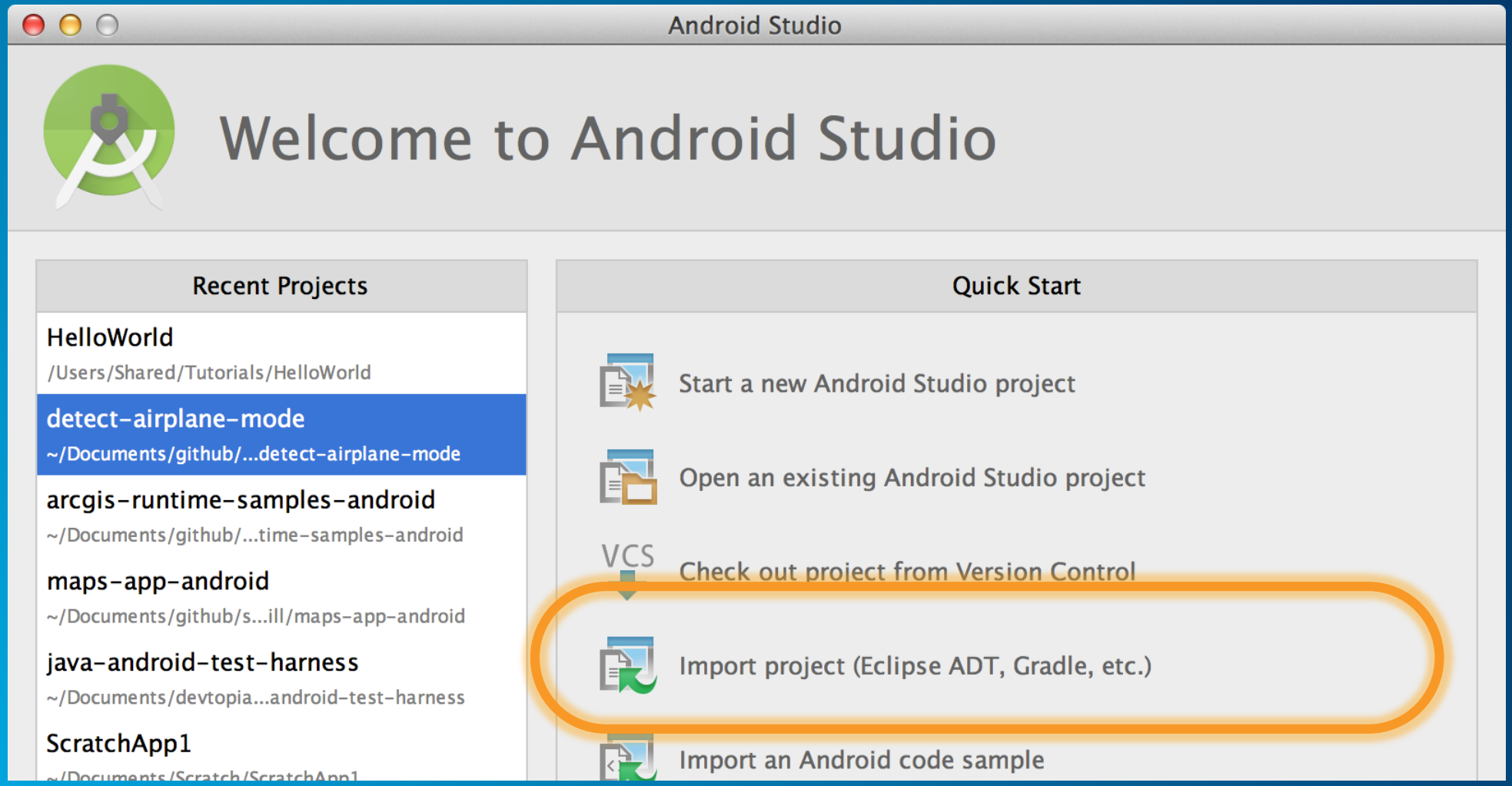
Build a map app from scratch?
http://developers.arcgis.com/android/guide/develop-your-first-map-app.htm

Clone samples repo from GitHub and run a sample?

http://github.com/Esri/arcgis-runtime-samples-android

Add the Esri ArcGIS maven dependency to your existing app?

Tip: Import cloned projects



Not “Open an existing Android Studio Project”

Gradling the ArcGIS dependency

- Define the maven repository location (typically in project root build.gradle)

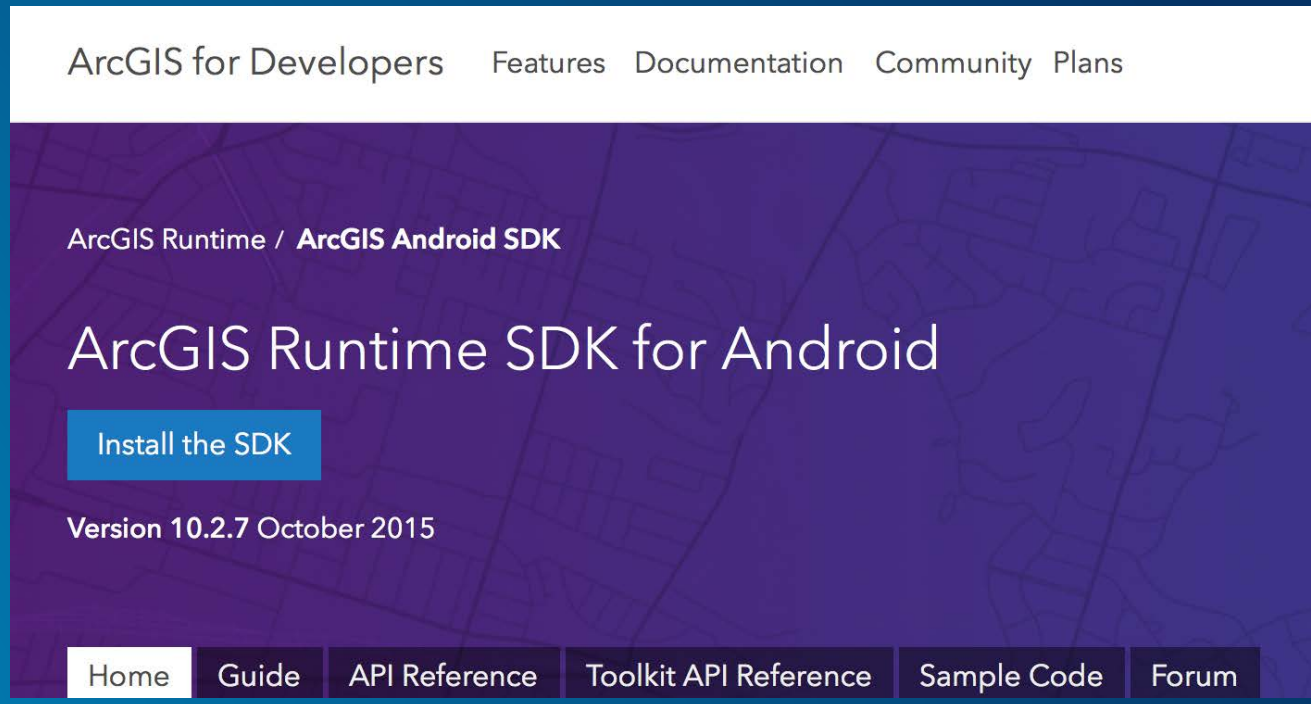
```
repositories {
  jcenter()
  maven {
    url 'https://esri.bintray.com/arcgis'
  }
}
```

- Add the ArcGIS Android AAR dependency (typically in app module build.gradle)

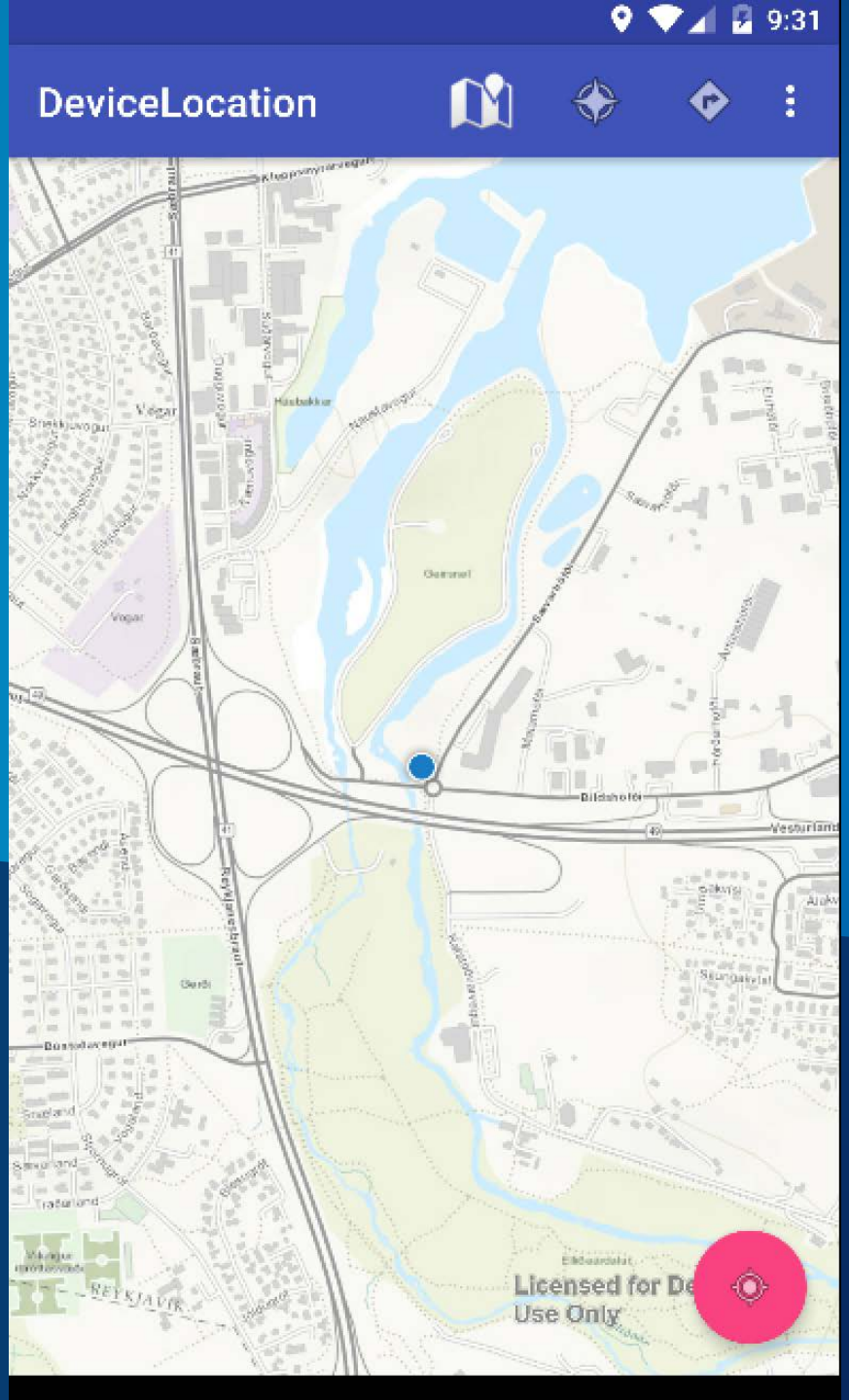
```
dependencies {
  compile 'com.esri.arcgis.android:arcgis-android:10.2.7’
}
```

SDK resources

- **Developers site**
  - http://developers.arcgis.com/android
  - **Guide**
  - **API Reference**
  - **Toolkit reference**
  - **Downloadable SDK**
- **GitHub**
  - **Samples -** http://github.com/Esri/arcgis-runtime-samples-android
  - **Maps app -** http://github.com/Esri/maps-app-android
- **GeoNet user community**
  - http://geonet.esri.com/community/developers/native-app-developers/arcgis-runtime-sdk-for-android



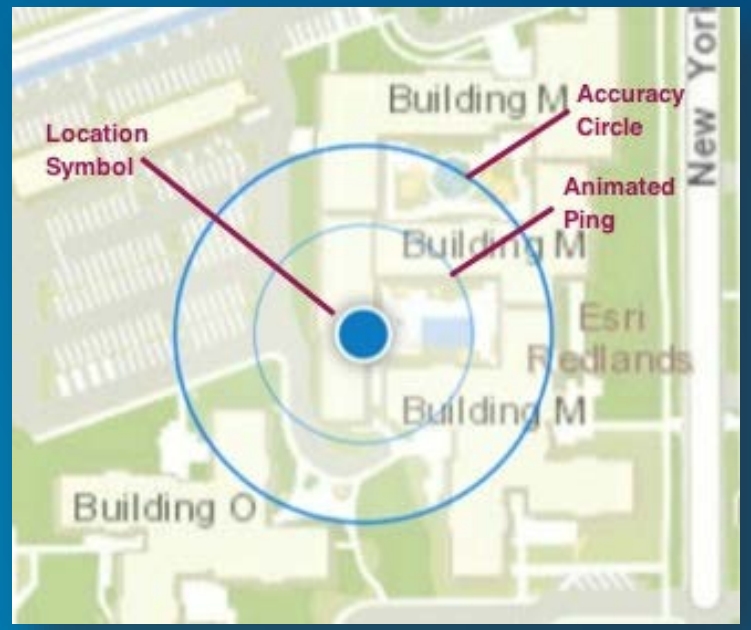
Device Location

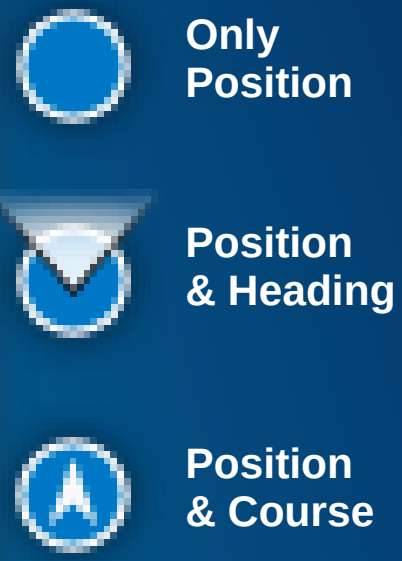


Demo: DeviceLocation – updating the map extent based on current location

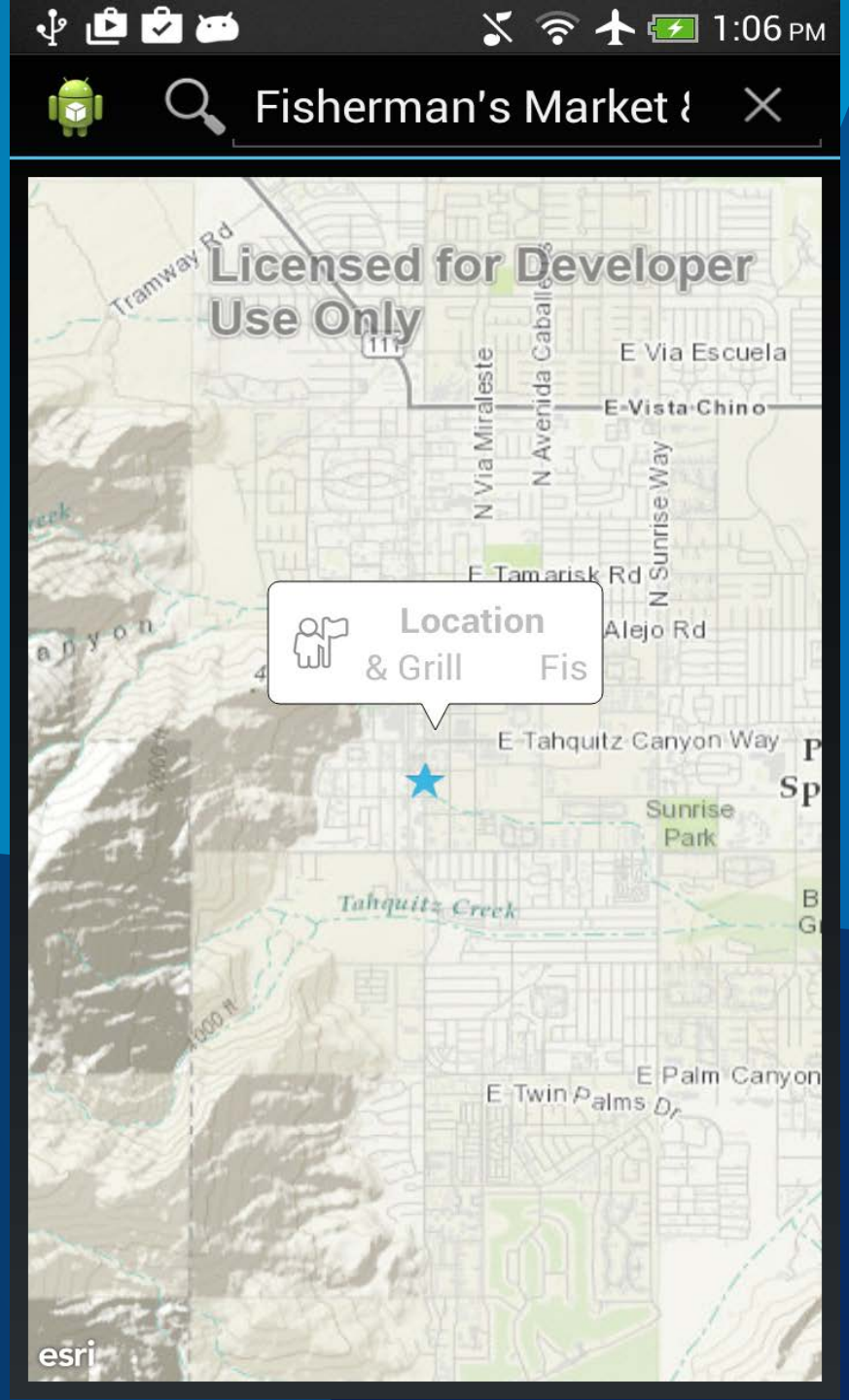
Customize location symbology

- Match application theme or style
- Indicate changing conditions
  - Moving in/out of geofences
- Show accuracy
- LocationDisplayManager setters
  - setDefaultSymbol()
  - setHeadingSymbol()
  - setCourseSymbol
  - setAccuracySymbol,
  - setPingSymbol()
  - More…





Finding places



Demo: PlaceSearch - finding places by name, showing suggestions

https://github.com/Esri/arcgis-runtime-samples-android/tree/master/PlaceSearch

Fetching suggestions

```
final Locator locator = Locator.createOnlineLocator();


searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener() {

  public boolean onQueryTextChange(final String newText) {

    LocatorSuggestionParameters suggestParams = new LocatorSuggestionParameters(newText);
    suggestParams.setLocation(mapView.getCenter(), mapView.getSpatialReference());
    suggestParams.setDistance(500.0); // meters

    locator.suggest(suggestParams, suggestCallback);

    return true;
  }
```

Fetching suggestions - callback

```
CallbackListener<List<LocatorSuggestionResult>> suggestCallback =
    new CallbackListener<List<LocatorSuggestionResult>>() {

  public void onCallback(final List<LocatorSuggestionResult> results) {

    runOnUiThread(new Runnable() {

      public void run() {
        // display results
      }
    });
  }

  public void onError(Throwable throwable) {
    // Handle the error
  }

};
```

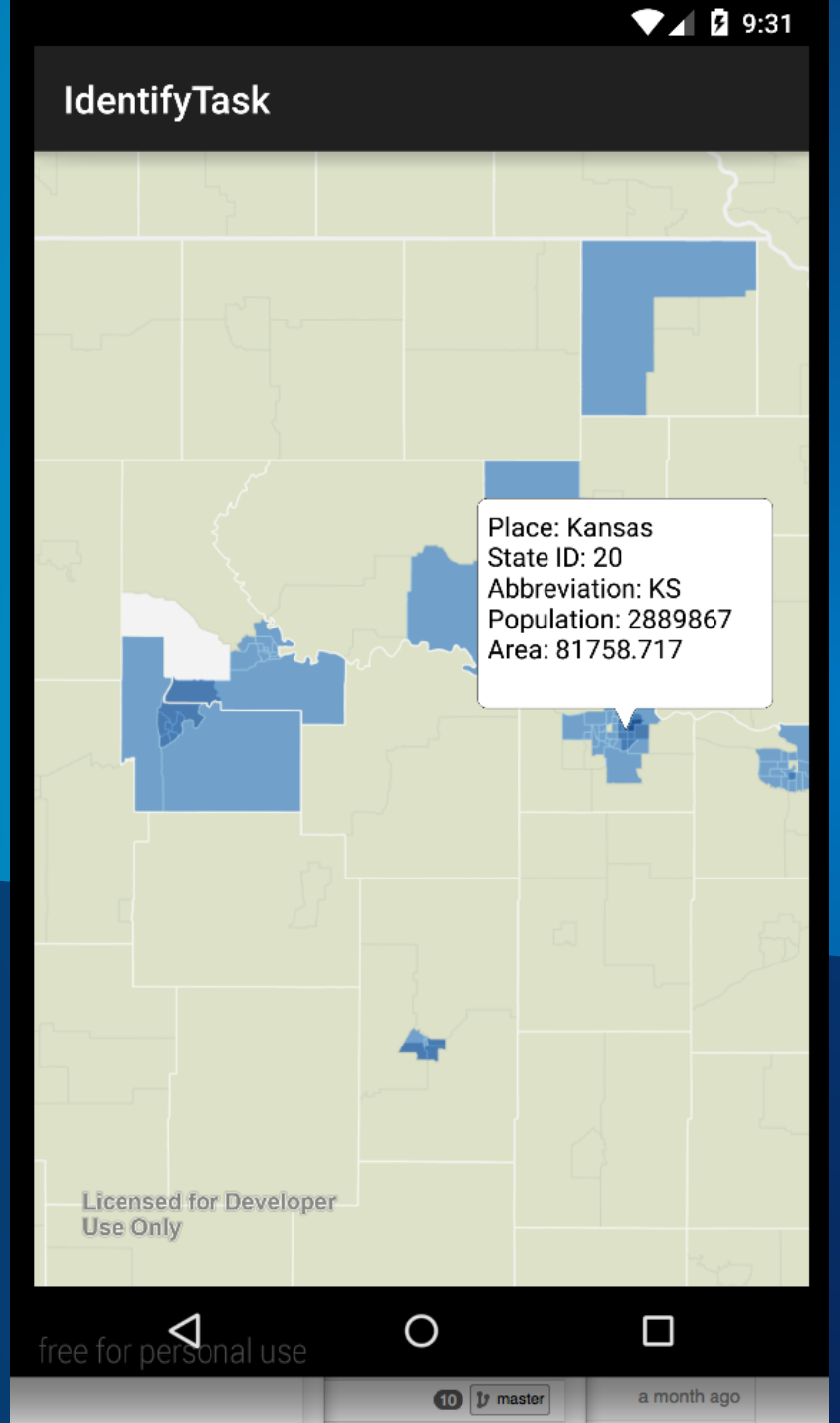
Finding address candidates

```
LocatorSuggestionResult result = resultSelectedByUser();
locator.find(result, 1, null, mapSpatialReference, findCallback);
```

Finding address candidates - callback

```
CallbackListener<List<LocatorGeocodeResult>> findCallback =
    new CallbackListener<List<LocatorGeocodeResult>>() {

      public void onCallback(final List<LocatorGeocodeResult> results) {

        runOnUiThread(new Runnable() {

          public void run() {
            LocatorGeocodeResult result = results.get(0);
            Point location = result.getLocation();
            display(location); // displays the first result
          }
        });
      }

      public void onError(Throwable throwable) {
        // Handle the error
      }

    };
```

Identifying features



Demo: IdentifyTask – getting more information about a feature by tapping on the map

https://github.com/Esri/arcgis-runtime-samples-android/tree/master/IdentifyTask

IdentifyTask

```
final IdentifyTask identifyTask = new IdentifyTask(IDENTITY_SERVICE_URL);

mapView.setOnSingleTapListener(new OnSingleTapListener() {

  public void onSingleTap(float x, float y) {

    Point identifyPoint = mapView.toMapPoint(x, y);
    Envelope env = new Envelope();
    mapView.getExtent().queryEnvelope(env);
    int[] layerIds = new int[]{ STATES_LAYER };

    IdentifyParameters params = new IdentifyParameters(
        identifyPoint, env, mapView.getSpatialReference(), layerIds, mapView.getWidth(),
        mapView.getHeight(), dpi, false);

    identifyTask.execute(params, identifyCallback);
  }
});
```

IdentifyTask - callback

```
CallbackListener<IdentifyResult[]> identifyCallback =
    new CallbackListener<IdentifyResult[]>() {

      public void onCallback(final IdentifyResult[] results) {

        runOnUiThread(new Runnable() {
          public void run() {
            // display results
          }
        });
      }

      public void onError(Throwable throwable) {
        // Handle the error
      }
    };
```

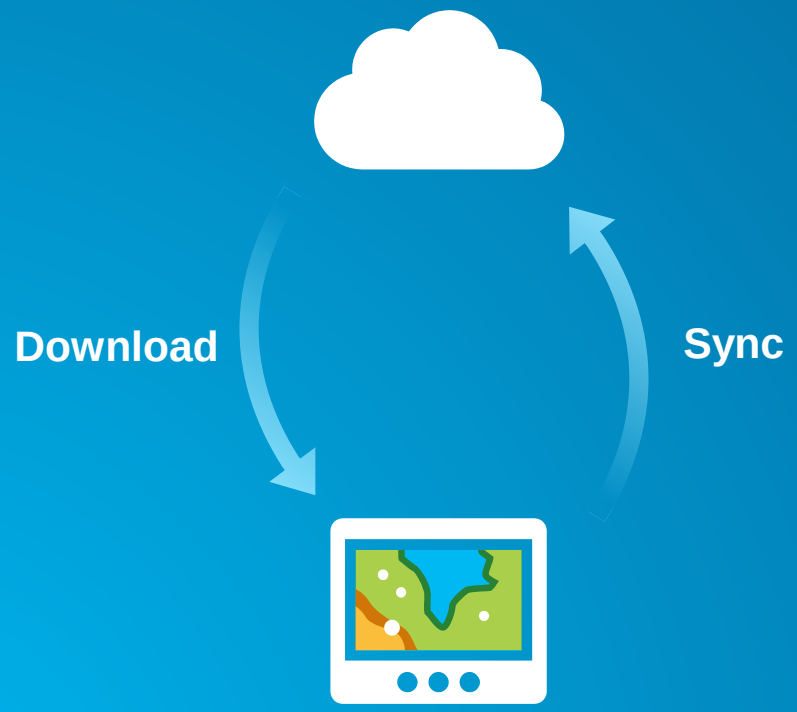
Similar workflows

- GraphicsLayer
  - getGraphicIds(x,y,tolerance)
- Query a FeatureLayer
  - SQL query for any attribute, spatial (or temporal if applicable) criteria
  - FeatureTable.queryFeatures(QueryParameters, CallbackListener<FeatureResult>)
- Reverse geocode
  - Convert map coordinates to address using Locator.findAddress()
- Popup
  - Specifically show attributes of a feature
  - Can be shown as a ‘more detail’ step after a Callout
  - Often used in editing

Editing

Offline

Services Pattern



Desktop Pattern

Getting an offline geodatabase

- **Simple coding steps:**
  - **Make a task : a geodatabase task**
  - **Create some parameters**
  - **Request the geodatabase using the task and parameters**
  - **Use asynchronous code to monitor progress and completion.**

Connecting to the offline geodatabase

- Open Geodatabase
- Get Feature Table
- Make feature layer and add it to the map

```
geodatabase = new Geodatabase(path);
final GeodatabaseFeatureTable geodatabaseTable = geodatabase.getGeodatabaseFeatureTableByLayerId(0);
localFeatureLayer = new FeatureLayer(geodatabaseTable);
```

Edit operations

- GeodatabaseFeatureTable has methods for
  - addFeature, deleteFeature, updateFeature

```
//make a set of attributes
Map<String, Object> attrs = new HashMap<String, Object>();
attrs.put("FIRSTNAME", "Mark");
attrs.put("TYPDAMAGE", "Minor");
```

```
//create a feature with these attributes at the given point
GeodatabaseFeature f = new GeodatabaseFeature(attrs, pt, gdbTable);
```

```
//add the feature into the geodatabase table returning the new feature id
long fid = gdbTable.addFeature(f);
```

Synchronization

- 2 way process
  - Upload your edits
  - Download other changes

  Alternatively you can:
  - Upload only
  - Download only

- Familiar development patterns
  - Task
  - Parameters
  - Call method on task to Sync.

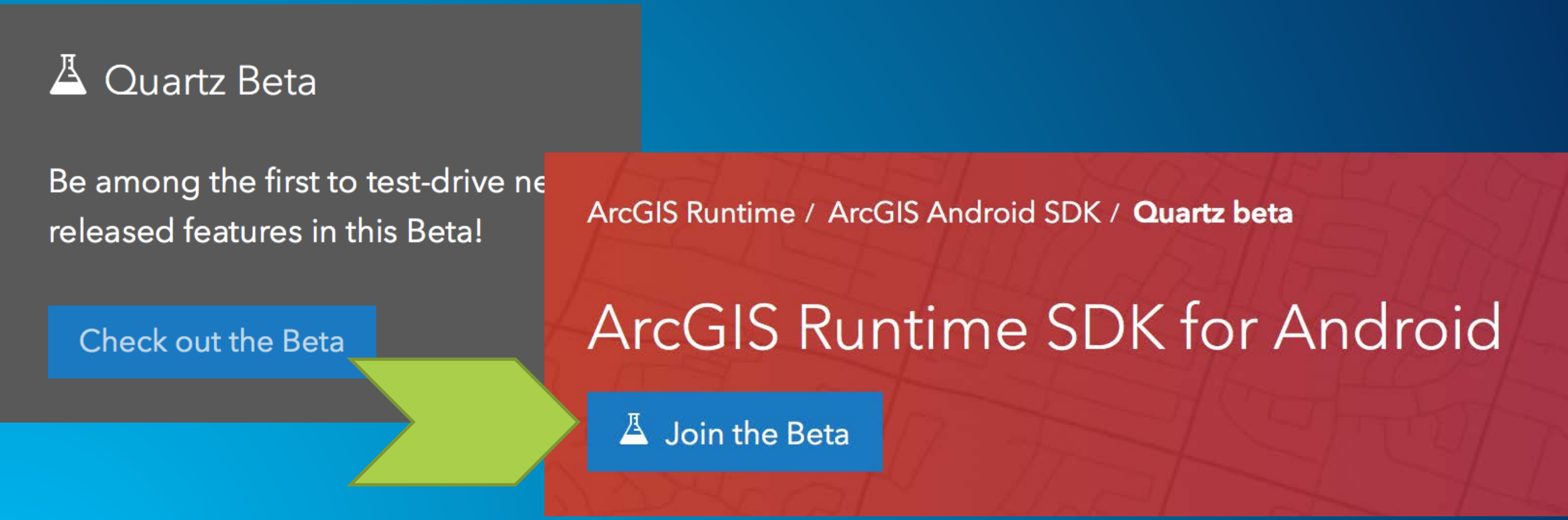
Quartz

Quartz

- Major release
  - Many new capabilities
  - New and changed APIs
  - Improved internal architecture
- Goals
  - Support the ArcGIS Platform
  - Move ArcGIS Engine developers to the ArcGIS Runtime
  - Synchronize APIs across all ArcGIS Runtime platforms
  - Support specific user workflows

Get the Quartz beta

https://developers.arcgis.com/android/



Map and MapView

- Content and presentation are separated
- Map is a separate class
  - Map contains layers, content
- MapView
  - Presentation layer
  - Contains the map
  - Will help with material design apps
  - Extends android.view.ViewGroup
  - Viewpoints, including navigation completed handler

When is the map navigating?

- Listening for navigation events on MapView
- To take account of animation, user interaction

```
mMapView.addDrawStatusChangedListener(new DrawStatusChangedListener() {
    @Override
    public void drawStatusChanged(DrawStatusChangedEvent drawStatusChangedEvent) {
        if(drawStatusChangedEvent.getDrawStatus() == DrawStatus.IN_PROGRESS){
            progressBar.setVisibility(View.VISIBLE);
        }else if (drawStatusChangedEvent.getDrawStatus() == DrawStatus.COMPLETED){
            progressBar.setVisibility(View.INVISIBLE);
        }
    }
});
```

Chaining navigation events (Quartz Beta 2)

- Navigation in Quartz uses Viewpoints
  - MapView.setViewpointWithDurationAsync(Viewpoint)
- Quartz Beta 2
  - Returns ListenableFuture<Boolean>
  - Listen for completion of previous navigation before starting new navigation
  - Navigation can be interrupted by user – future.get()

```
final ListenableFuture<Boolean> booleanListenableFuture = mMapView.setViewpointWithDurationAsync(firstViewpoint, 3);
booleanListenableFuture.addDoneListener(new Runnable() {
  @Override
  public void run() {
    try {
      if (booleanListenableFuture.get()) {
        // First navigation is complete, was not interrupted by the user or another navigation.
        mMapView.setViewpointWithDurationAsync(secondViewpoint, 3);
      }
    } catch (InterruptedException | ExecutionException e) {
      e.printStackTrace();
```

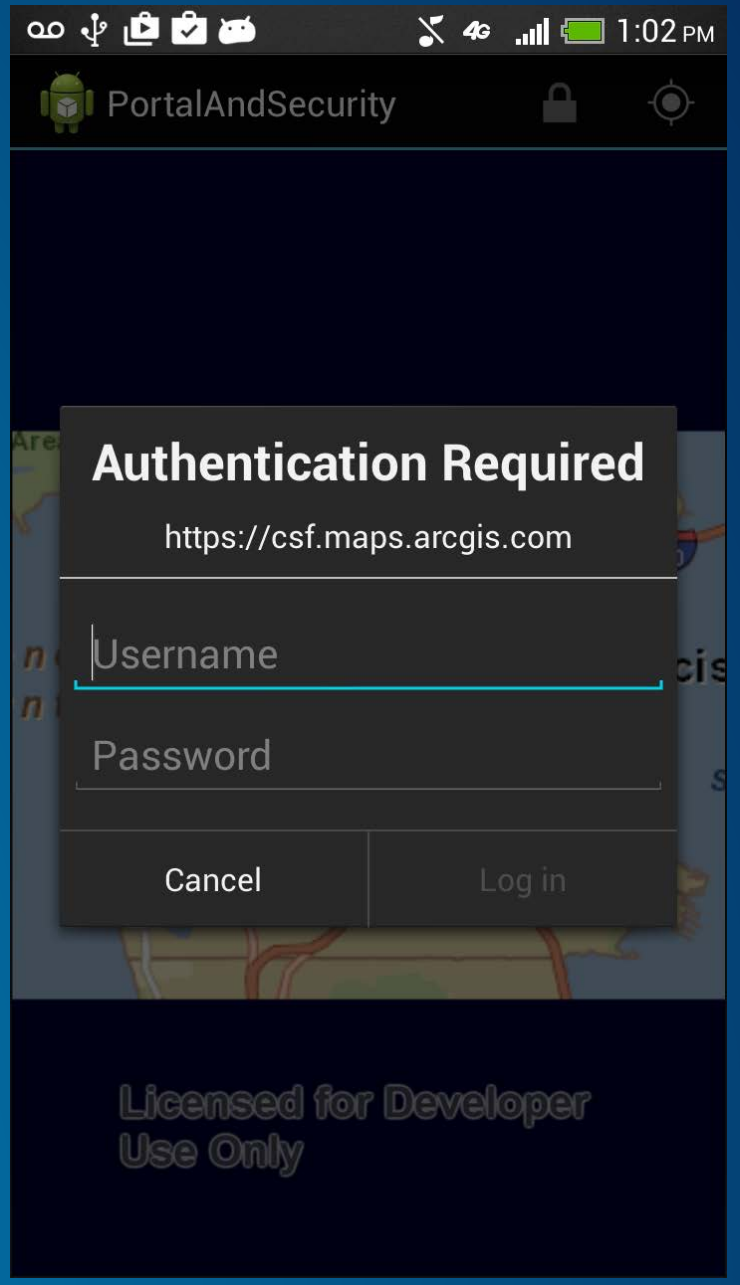
Security: authentication and challenges

- AuthenticationManager
  - Provides a single entry point for service authentication errors
  - Issues a challenge to its current handler on authentication errors
  - Contains a credential cache – can be re-used on subsequent authentication attempts
- AuthenticationChallenge
  - Created by the API code when it’s denied access to a service (missing or incorrect credential, etc)
  - Contains type of request attempted, exception details, number of failed attempts
- AuthenticationChallengeHandler
  - Set onto AuthenticationManager
  - Receives AuthenticationChallenge
  - Returns what to do next
  - DefaultAuthenticationChallengeHandler, or create a custom handler

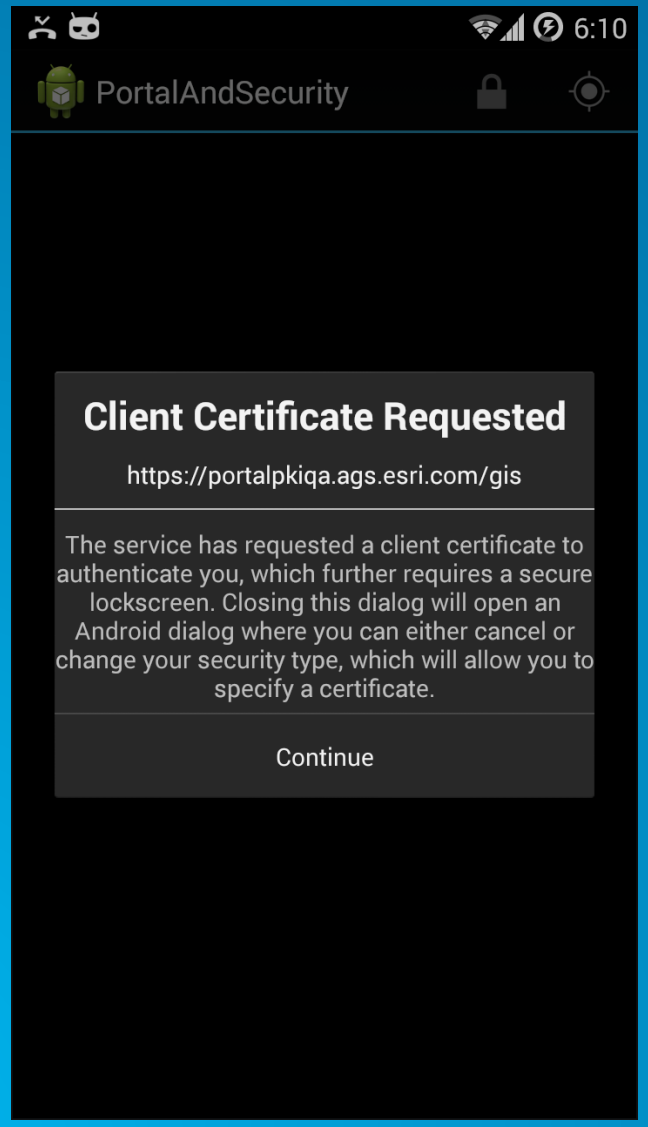
DefaultAuthenticationChallengeHandler

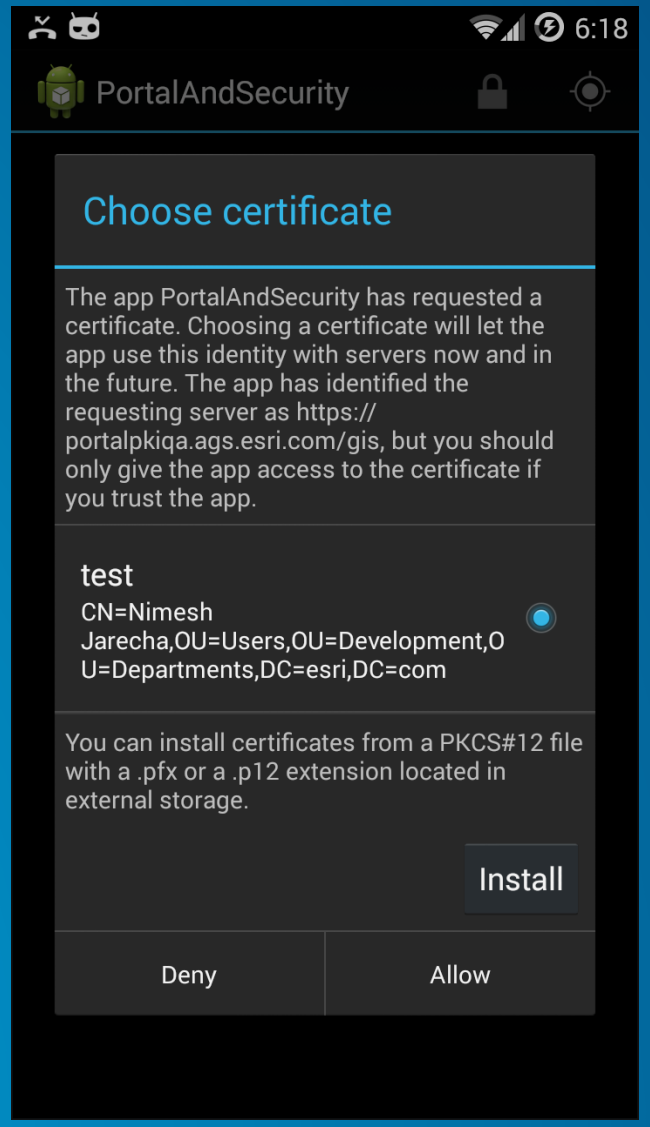
- **Supporting workflows:**
  - **credentials (http, token-based)**
  - **certificates (PKI)**
  - **self signed certificates**
- **Can be styled to match your app**

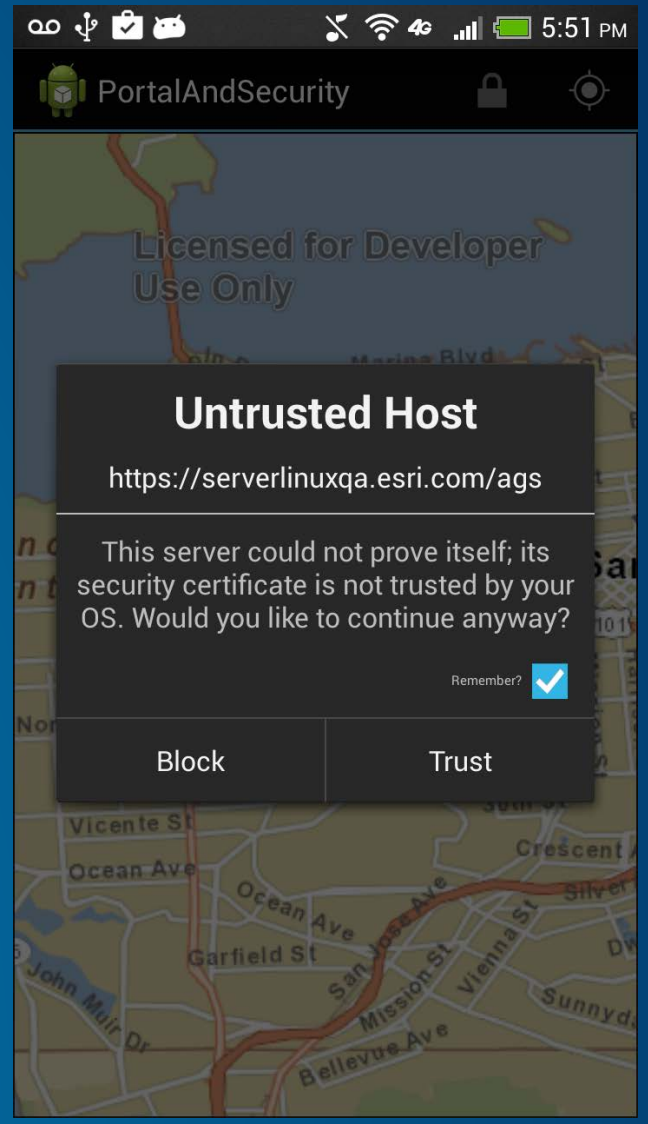
```
AuthenticationManager.setAuthenticationChallengeHandler(
    new DefaultAuthenticationChallengeHandler(this));
```



DefaultAuthenticationChallengeHandler – certificate



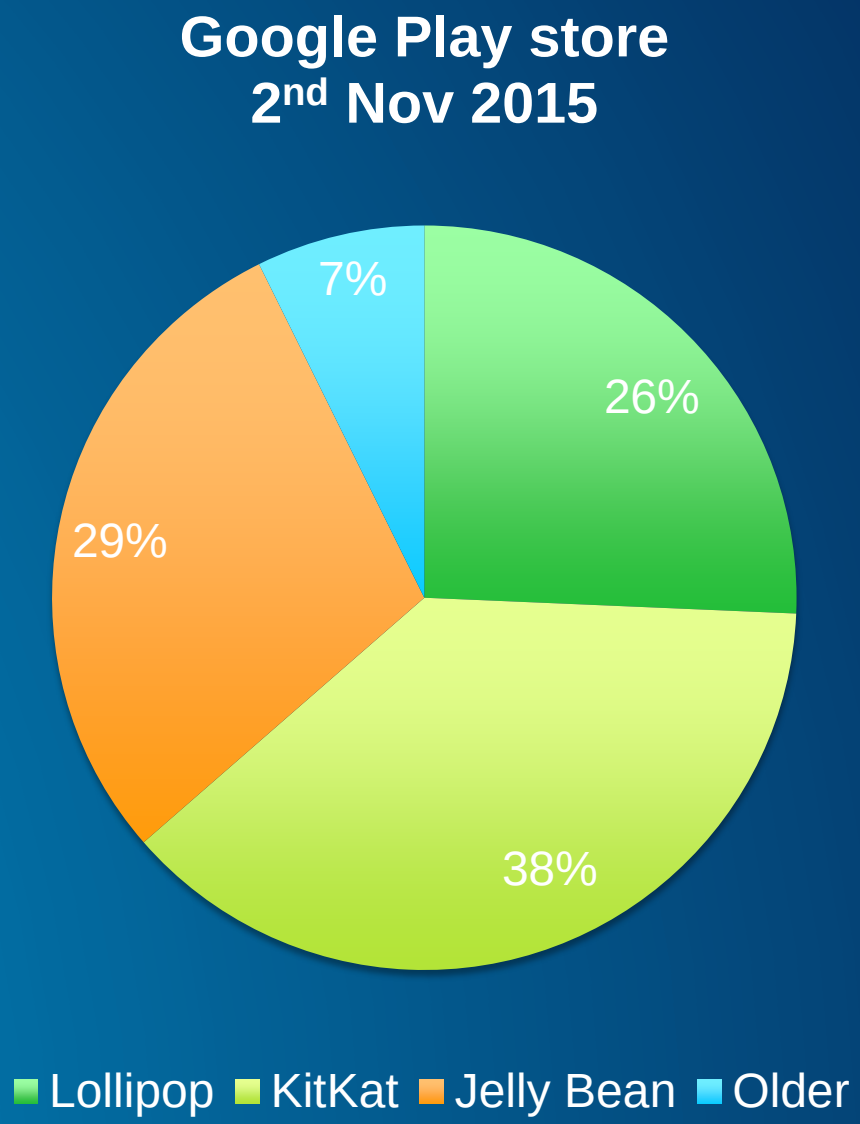




And finally…

Android SDK requirements

- Android SDK version 14 – Ice Cream Sandwich
  - 10.2.7, Quartz Beta 1
  -
```
defaultConfig {
    minSdkVersion 14
}
```
- Quartz Beta 2
  - Possibly Android SDK version 16 - Jelly Bean
  - What about your users?
- For your dev machine:
  - Requires JDK v6 or v7

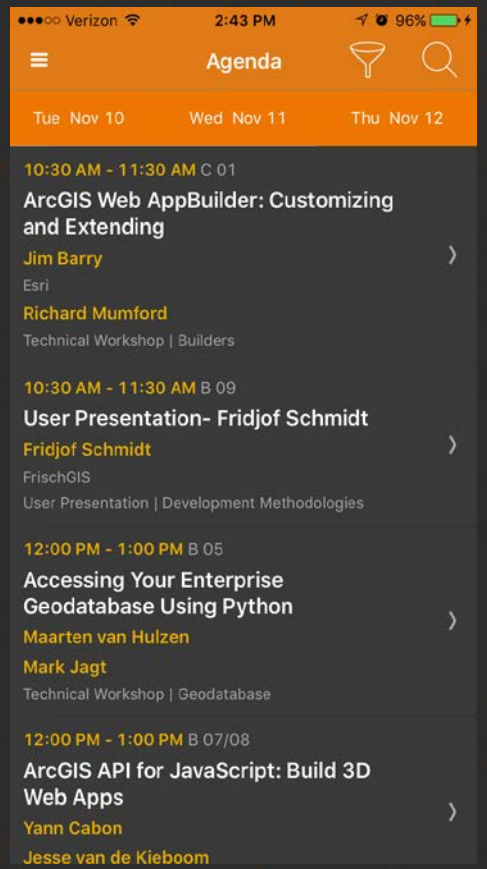


Questions?

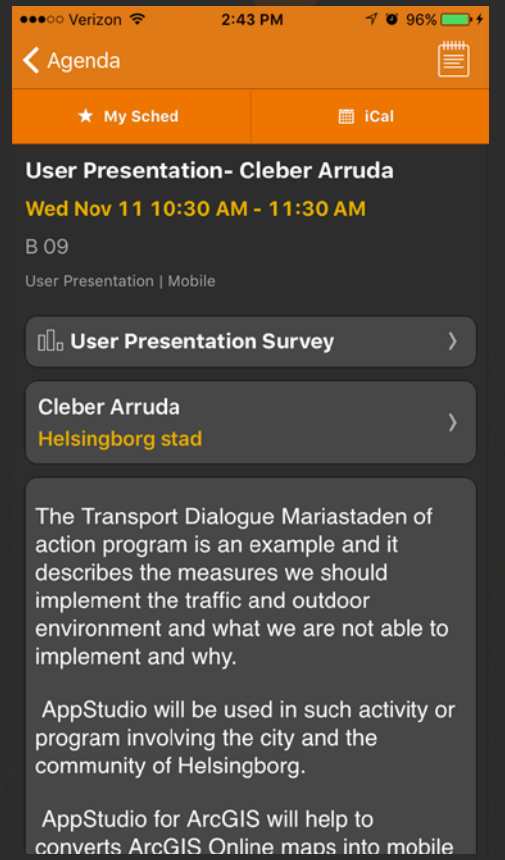
Please Take Our Survey!

Download the Esri Events app and find your event

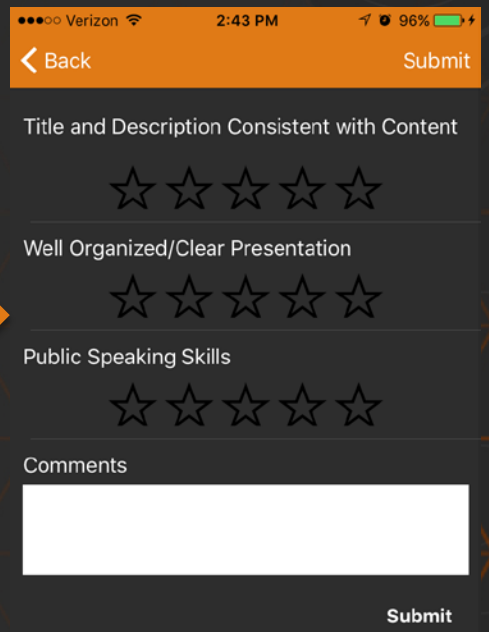
Select the session you attended



Select “User Presentation Survey” or “Technical Workshop Survey”



Complete Answers and Select “Submit”




esri®