



Esri International Developer Summit
Palm Springs, CA

Customizing Android Pop-ups to Fit Your App's UI

Xueming Wu

Agenda

- **Introduction**
- **What to customize a pop-up**
- **How to customize a pop-up**
- **Q & A**

Introduction

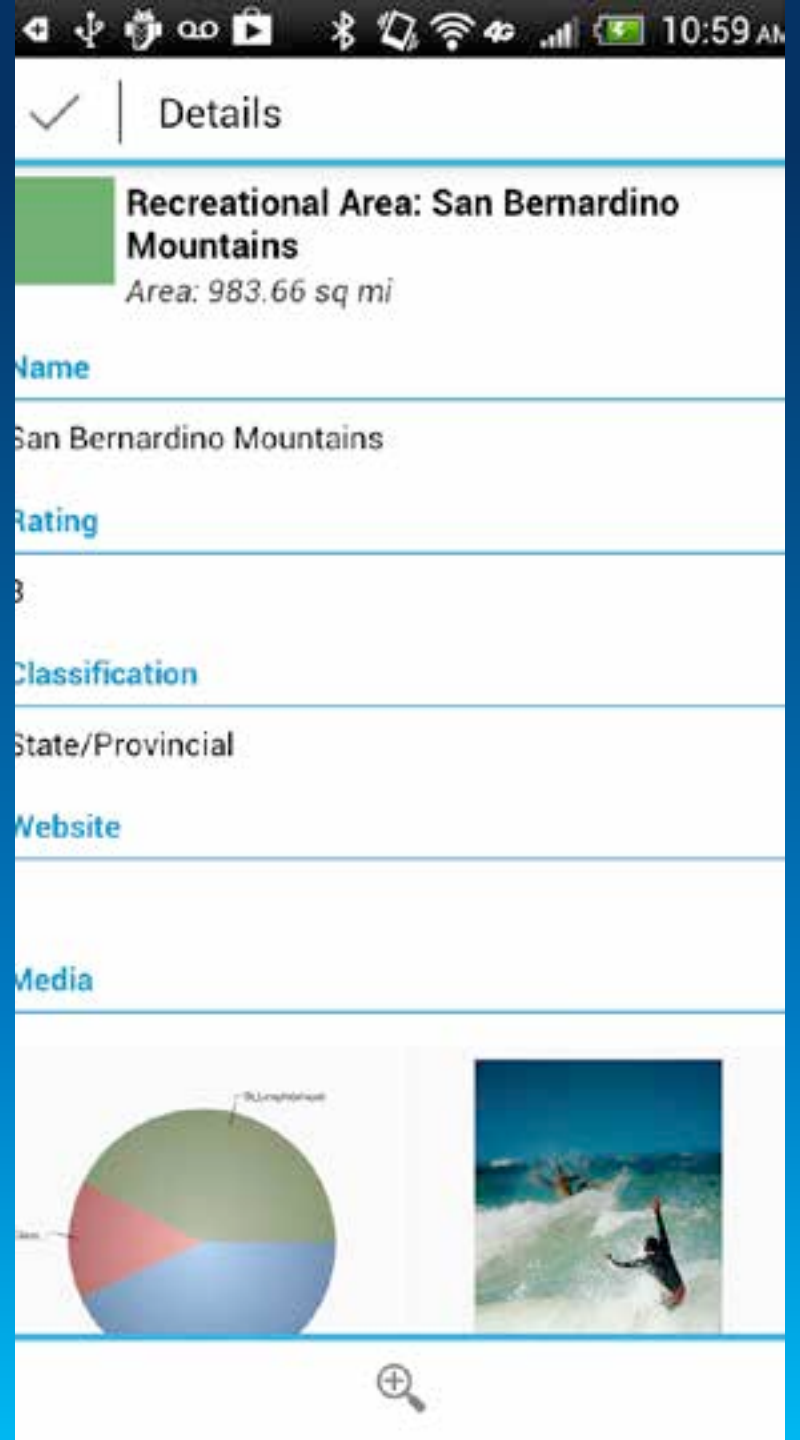
Product Engineer of Android SDK Development team

- Xueming Wu

Who Are You?

Pop-up API

UI and behaviors for displaying and editing information about features.

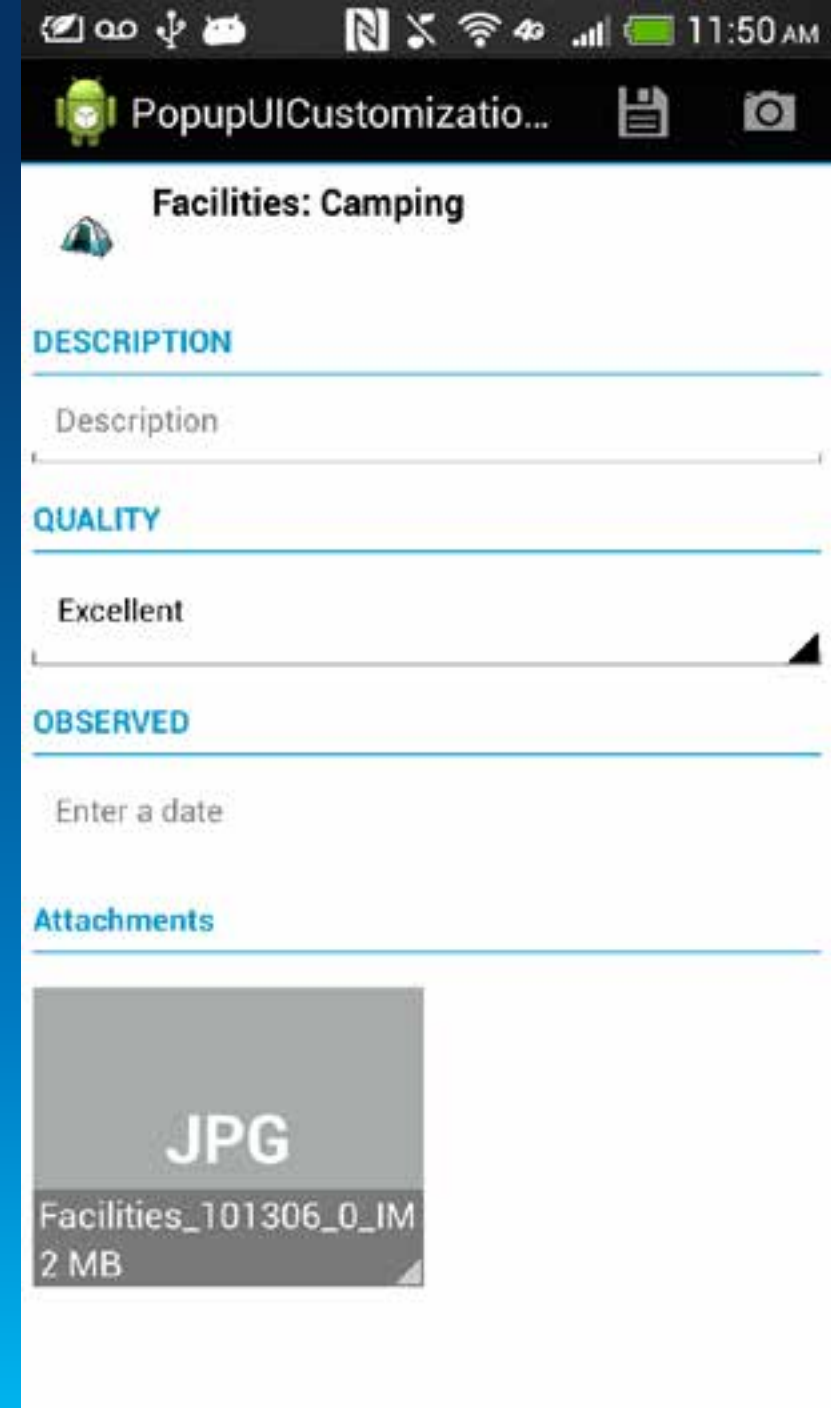
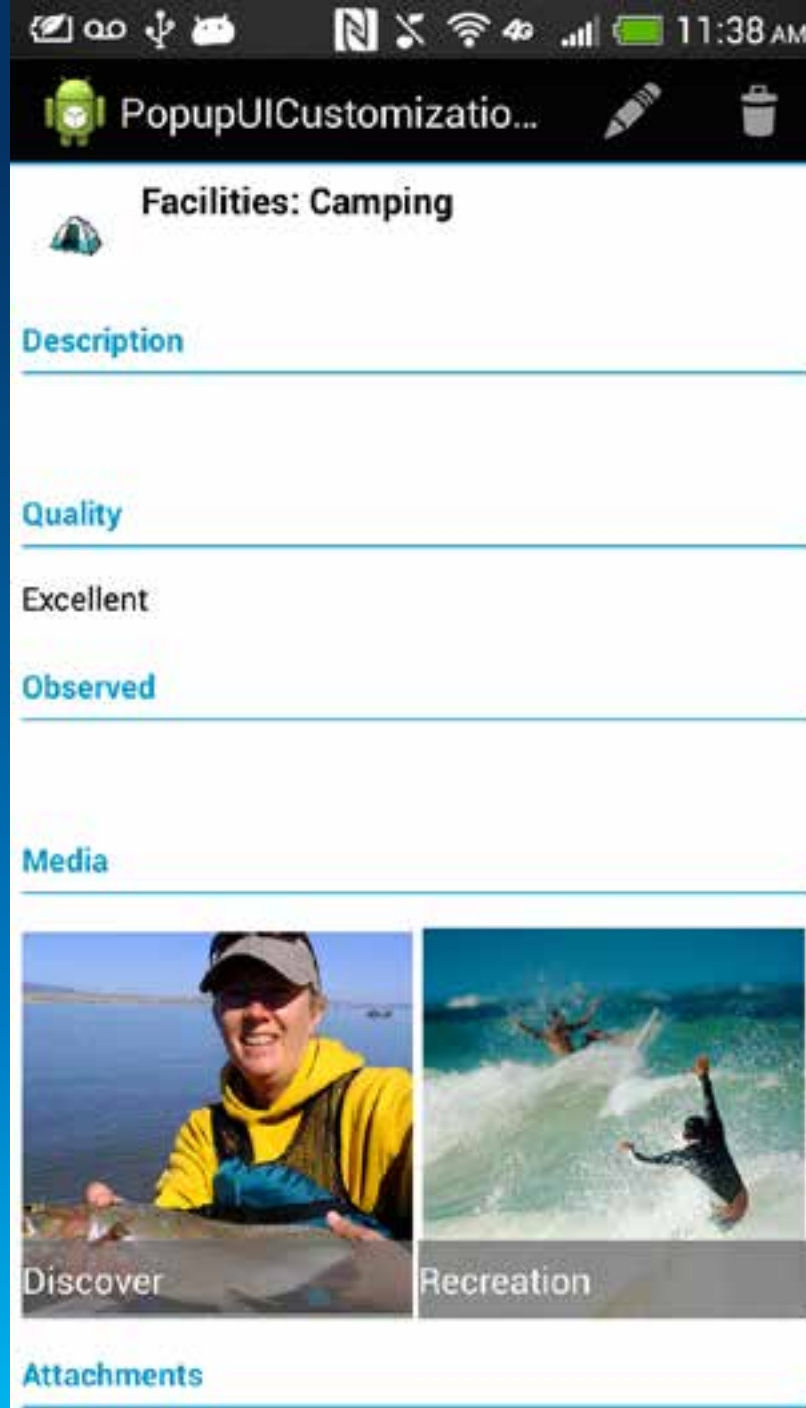


General workflow of using pop-up API

- **Display information using pop-ups**
 - Instantiate `PopupContainer`
 - Create a `Popup` and add it to the `PopupContainer`
 - Display `PopupContainerView`
- **Handle user interaction**
- **Edit information using pop-ups**
- **Customize UI**

Customize pop-up UI

- Title
- Attributes
- Media
- Attachments



Pop-up UI Customization Demo

Xueming Wu

 **Facilities: Camping**
X: -12,941,705.977, Y:3,924,638.037

DESCRIPTION

QUALITY
Excellent
OBSERVED



MEDIA



Discover



Recreation

What to customize pop-up

- **Layout**
- **Style**
 - *Color*
 - *Font*
 - *Size*
 - *Separator*
- *Localization*

How to customize pop-up

- **Methods**

- **Change background color** -- *ArcGISPopupStyle.setPopupBackgroundColor(int)*
- **Turn on/off separator** -- *ArcGISPopupStyle.setShowFieldSeparator(boolean)*
- **Change text font** -- *ArcGISPopupStyle.setFieldValueFontType(Typeface)*
-

- **XML + sub-class of built-in class**

- *Layout*
- *Style*

XML + sub-class of built-in class

- Layout

```
private PopupLayoutInfo createPopupLayout(Context context) {  
  
    PopupLayoutInfo layout = new PopupLayoutInfo();  
    // inflate the layout  
    ViewGroup xmlLayout = (ViewGroup) ((Activity)context).getLayoutInflater().inflate(R.layout.popup_layout, null);  
  
    // set the xml layout on the popup layout  
    layout.setLayout(xmlLayout);  
  
    // Tell the PopupLayout where to place the different elements  
    layout.setTitleViewPlaceholder((ViewGroup) xmlLayout.findViewById(R.id.title_view_placeholder));  
    layout.setAttributesViewPlaceholder((ViewGroup) xmlLayout.findViewById(R.id.attribute_view_placeholder));  
    layout.setMediaViewPlaceholder((ViewGroup) xmlLayout.findViewById(R.id.media_view_placeholder));  
    layout.setAttachmentsViewPlaceholder((ViewGroup) xmlLayout.findViewById(R.id.attachment_view_placeholder));  
  
    return layout;  
}
```

```
PopupLayoutInfo layout = createPopupLayout(mMapView.getContext());  
Popup popup = layer.createPopup(mMapView, 0, gr, layout, false);  
createViews(popup);
```

XML + sub-class of built-in class

- Layout

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >
    <FrameLayout
        android:id="@+id/title_view_placeholder"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" >
    </FrameLayout>
    <FrameLayout
        android:id="@+id/attribute_view_placeholder"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" >
    </FrameLayout>
    <LinearLayout
        android:id="@+id/second_inner_linearlayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <ImageButton
            android:id="@+id/find_directions_button"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="50dp"
            android:background="@null"
            android:src="@android:drawable/ic_menu_directions"/>
        <ImageButton
            android:id="@+id/favorite_button"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="50dp"
            android:background="@null"
            android:src="@android:drawable/star_big_on"/>
        <ImageButton
            android:id="@+id/share_button"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="50dp"
            android:background="@null"
            android:src="@android:drawable/ic_menu_share"/>
    </LinearLayout>
    <FrameLayout
        android:id="@+id/media_view_placeholder"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" >
    </FrameLayout>
    <FrameLayout
        android:id="@+id/attachment_view_placeholder"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" >
    </FrameLayout>
</LinearLayout>
```

XML + sub-class of built-in class

- Style

```
private void createViews(Popup popup) {
    if (popup == null || popup.getLayout() == null)
        return;

    PopupLayout popupLayout = popup.getLayout();
    Context context = mapView.getContext();

    // Create a view for the title
    popupLayout.setTitleView(new MyTitleView(context, popup, mapView));

    // Create a view for attributes
    popupLayout.setAttributesView(new MyAttributeView(context, popup));

    // Create an adapter for the attributes in read-only mode
    ((ArcGISAttributeView) popup.getLayout().getAttributesView())
        .setReadOnlyAdapter(new MyReadOnlyAttributesAdapter(context, popup));

    // Create an adapter for the attributes in edit mode
    ((ArcGISAttributeView) popup.getLayout().getAttributesView())
        .setEditAdapter(new MyEditAttributesAdapter(context, popup));

    // Change the view for the attachments
    if (popup.getPopupInfo() != null && popup.getPopupInfo().isShowAttachments()) {
        popup.getLayout().setAttachmentsView(new MyAttachmentsView(context, popup));
    }

    // Change the view for the media
    popup.getLayout().setMediaView(new MyMediaView(context, popup));
}
```

XML + sub-class of built-in class

- Style

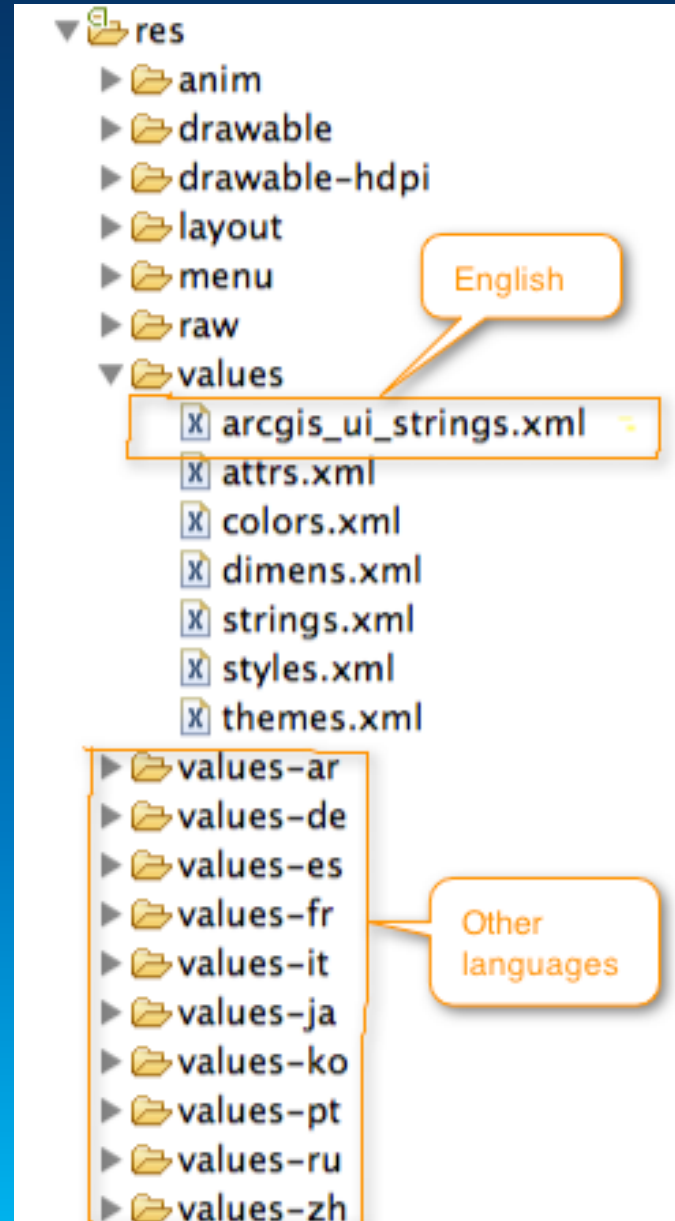
```
public class MyTitleView extends ArcGISMapView {  
  
    public MyTitleView(Context context, Popup popup, MapView mapView) {  
        super(context, popup, mapView);  
  
        removeAllViews();  
  
        LinearLayout layout = (LinearLayout) ((Activity) context).getLayoutInflater().  
            inflate(R.layout.popup_title_view, null);  
        addView(layout, new LayoutParams(android.view.ViewGroup.LayoutParams.MATCH_PARENT,  
            android.view.ViewGroup.LayoutParams.WRAP_CONTENT));  
  
        mSymbolView = (ImageView) layout.findViewById(R.id.symbol_imageView);  
  
        LinearLayout innerLayout = (LinearLayout) layout.findViewById(R.id.inner_linearlayout);  
  
        mTitle = (TextView) innerLayout.findViewById(R.id.title_textView);  
        mEditInfo = (TextView) innerLayout.findViewById(R.id.editor_tracking_info_textView);  
        mGeometryInfo = (TextView) innerLayout.findViewById(R.id.geometry_info_textView);  
  
        refresh();  
    }  
  
    @Override  
    public void setGeometryInfo(String geometryInfo) {  
        if (this.mGeometryInfo != null) {  
            this.mGeometryInfo.setText(geometryInfo);  
            this.mGeometryInfo.setVisibility(VISIBLE);  
        } else {  
            this.mGeometryInfo.setVisibility(GONE);  
        }  
    }  
  
    public void setSymbol(Symbol symbol) {  
        mSymbol = symbol;  
    }  
}
```

Localization of pop-up UI

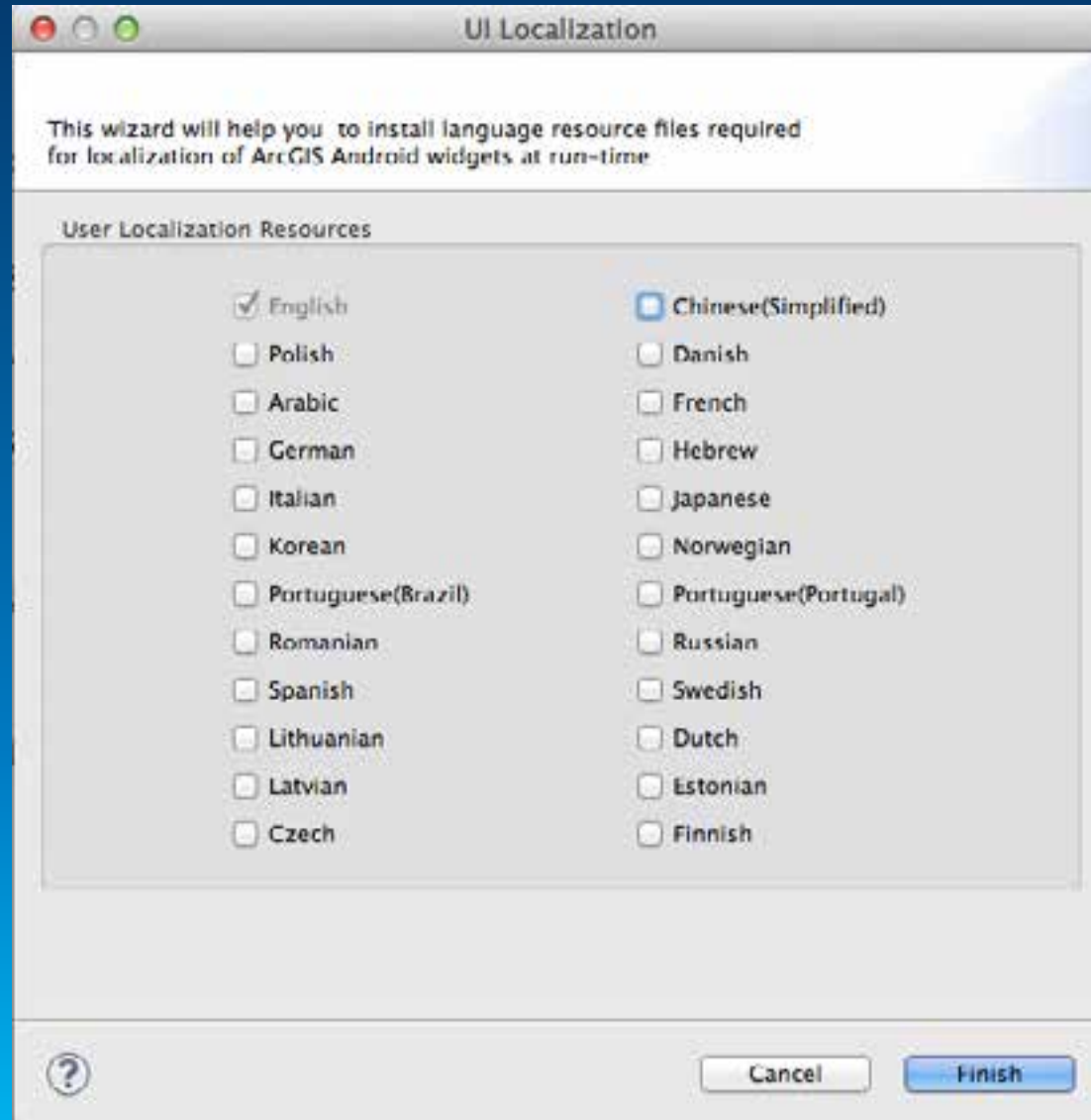
Text in the pop-up views has been externalized into an XML file called `arcgis_ui_strings.xml`.

```
<string name="ags_open">Open</string>
<string name="ags_download">Download</string>
<string name="ags_no_application_available_to_view_this_file_type">No application available to view this file type.</string>
<string name="ags_cannot_exceed_maximum_date">Cannot exceed maximum date</string>
<string name="ags_cannot_exceed_minimum_date">Cannot exceed minimum date</string>
<string name="ags_creation_user">Created by ${userId}</string>
<string name="ags_creation_seconds_ago">Created seconds ago</string>
<string name="ags_creation_a_minute_ago">Created a minute ago</string>
<string name="ags_creation_minutes_ago">Created ${minutes} minutes ago</string>
<string name="ags_creation_an_hour_ago">Created an hour ago</string>
<string name="ags_creation_hours_ago">Created ${hours} hours ago</string>
<string name="ags_creation_weekday">Created on ${weekday} at ${formattedTime}</string>
<string name="ags_creation_full">Created on ${formattedDate} at ${formattedTime}</string>
<string name="ags_creation_user_seconds_ago">Created by ${userId} seconds ago</string>
<string name="ags_creation_user_a_minute_ago">Created by ${userId} a minute ago</string>
<string name="ags_creation_user_minutes_ago">Created by ${userId} ${minutes} minutes ago</string>
<string name="ags_creation_user_an_hour_ago">Created by ${userId} an hour ago</string>
<string name="ags_creation_user_hours_ago">Created by ${userId} ${hours} hours ago</string>
<string name="ags_creation_user_weekday">Created by ${userId} on ${weekday} at ${formattedTime}</string>
<string name="ags_creation_user_full">Created by ${userId} on ${formattedDate} at ${formattedTime}</string>
<string name="ags_edit_user">Edited by ${userId}</string>
<string name="ags_edit_seconds_ago">Edited seconds ago</string>
<string name="ags_edit_a_minute_ago">Edited a minute ago</string>
<string name="ags_edit_minutes_ago">Edited ${minutes} minutes ago</string>
<string name="ags_edit_an_hour_ago">Edited an hour ago</string>
<string name="ags_edit_hours_ago">Edited ${hours} hours ago</string>
<string name="ags_edit_weekday">Edited on ${weekday} at ${formattedTime}</string>
<string name="ags_edit_full">Edited on ${formattedDate} at ${formattedTime}</string>
<string name="ags_edit_user_seconds_ago">Edited by ${userId} seconds ago</string>
<string name="ags_edit_user_a_minute_ago">Edited by ${userId} a minute ago</string>
<string name="ags_edit_user_minutes_ago">Edited by ${userId} ${minutes} minutes ago</string>
```

Localization of pop-up UI



Localization of pop-up UI



Customize pop-up UI

- **What to customize**
 - Layout
 - Style
 - *Localization*
- **How to customize**
 - Methods
 - XML + sub-class
- **Demo will be published as a sample in a future release**

Other related sessions

Date	Session Name	Time	Location
Tuesday	Deploying Android Apps	1:00pm – 1:30pm	Demo Theater 1 – Oasis 1
Wednesday	Building Android Apps with ArcGIS Runtime SDK	2:30pm – 3:30pm	Smoketree A – E
Thursday	Building Android Apps with ArcGIS Runtime SDK	2:30pm – 3:30pm	Mesquite GH
Thursday	The Road Ahead: ArcGIS Runtime SDKs	8:30am – 9:30am	Primrose A

Thank you!



Understanding our world.