

COMP30510 Mobile Application Development

Resources

Dr. Abraham Campbell
University College Dublin
Abey.campbell@ucd.ie

Resource Types

- Under /res/ of your project hierarchy

Typical:

- drawable/
- layout/
- values/
- raw/

Resource Types

Uncommon

- animator
 - anim
 - color
 - menu
 - xml
-
- Remember all xml files within res folder must be named using
 - a-z or 0-9 or _ so that they can be used as java identifier

Resources

- They provide excellent facilities for
 - Customizing your app to various hardware and software configurations
 - Localization
 - Static data storage for your app, both unstructured and semi-structured
- If possible, move ALL your data to resources, including strings, arrays, string formats, layouts, colours, menus, etc (provided that they are static)

/drawable

- Image files
 - .png
 - .jpg
 - .gif
- Remember extensions on to the drawable folder allow different images e.g
- drawable -> US flag would be shown
- Drawable-en_IE –Irish Flag (Irish phone)

/drawable

- Alternatively Vector based drawables can be defined in an xml file such as

```
<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="rectangle">
    <solid android:color="#FF00FF" />

    <corners android:topLeftRadius="5dp"
        android:topRightRadius="5dp"
        android:bottomLeftRadius="5dp"
        android:bottomRightRadius="5dp" />
</shape>
```

/layout

- XML files defining the layout to be used in an app
- Encapsulates all the files need to create a layout or defining a custom view .

/values

Default folder to set values for your application. There are a few filename conventions in android for this folder the two below are the most important.

- strings.xml
 - Useful to change your app into different languages
- styles.xml
 - Useful to set themes

/raw

- Arbitrary files that are saved in their own raw form for example movie files
- Can store any file but remember name rules
 - a-z 0-9 _ only allowed

Uncommon

- anim/
 - Animation for view objects
- animator/
 - Animations for non-view * view objects
- color/
 - Predefined colours
- menu/
 - Definitions for applications menus
- xml/
 - Arbitrary XML files that can be at runtime

Resources: Screen Support

- Screen size: actual physical size of the screen
 - small, normal, large, extra large
- Screen density: dots per inch (dpi)
 - low, normal, high, extra high
- Screen orientation: portrait, landscape, square
 - Can change at runtime!
- Resolution: total number of physical pixels
 - Less relevant, use screen size and density instead
- Density-independent pixels: virtual pixel units (dp)
 - Recommended for defining UI (based on 160 dpi)

Resources: User Input Support

- Touchscreen: stylus, finger, none
- Keyboard: QWERTY, numeric, neither
- Other input: Dpad, wheel

Resource Example

- Format of folders

res/⟨resources_name⟩⟨config_qualifier⟩

- Examples
 - res/layoutlargeandnotouchqwerty
 - res/layoutlandnotouchqwerty
 - res/layoutlargehdpi

Resource Qualifiers

Configuration	Name	Example
MCC, MNC	Country code, network code	Mcc272 (ie), mcc272-mnc01 (vodafone ie)
Language and region	ISO 639-1 language code and ISO 3166 region code (preceded by 'r')	en, en-rIE, ga, ru-rUA, pl, cn
sw<N>dp	Smallest Width	sw320dp
w<N>dp	Available Width	w640dp
h<N>dp	Available Height	h480dp
Screen size		Small, normal, large, xlarge
Screen aspect		Long, notlong
Screen orientation		Port, dock
Dock mode		Car, desk

Resource Qualifiers

Configuration	Name	Example
Night mode		Night, notnight
Screen pixel density		Ldpi, mdpi, hdpi, xhdpi, nodpi, tvdpi
Touchscreen type		Notouch, stylus, finger
Keyboard availability		Keysexposed, keyshidden, keyssoft
Primary text input method		Nokeys, qwerty, 12key
Navigation key availability		Navexposed, navhidden
Primary non-touch nav method		Nonav, dpad, trackball, wheel
Platform verison	Api level	V3, v4, v7...

Qualifier Rules

- Multiple qualifiers for single set of resources
- Qualifiers cannot be nested
- Qualifiers must appear in the exact order as listed in the table above
- Qualifiers are case-insensitive
- Only one value for each qualifier is supported
 - But aliases are possible!

Finding Best Matching Resource

- Process Android takes to finding which resource to use
 1. Eliminate resources that contradict device configuration
 2. Pick next available highest precedence qualifier
 3. If not found, go to step 2, otherwise go to step 4
 4. Eliminate all non-matching resources
 5. If only one directory remains, finish. Otherwise go to step 2.

Resource Matching Example

- drawable/
 - drawable-en/
 - drawable-fr-rCA/
 - Drawable-en-port/
 - drawable-en-notouch-12key/
 - drawable-port-ldpi/
 - drawable-port-no-touch12key
- Locale = **en-GB**
 - Screen orientation = **port**
 - Screen pixel density = **hdpi**
 - Touchscreen type = **notouch**
 - Primary text input method = **12key**

Accessing Resources

- Access from code:

```
ImageView imageView = (ImageView)  
findViewById(R.id.myimageview);  
imageView.setImageResource(R.drawable.  
myimage);
```

Accessing Resources Cont'd

- Access from XML:

<Button

android:layout_width="fill_parent"

android:layout_height="wrap_content"

android:text="@string/submit" />

Resource Localization (L10N)

- To support both English and Irish Gaelic, you would create two folders, named `res/values-en/` and `res/values-ga/`, where the value after the hyphen is the ISO 639-1 two-letter code for the language.
 - `res/values-en/strings.xml`
 - `res/values-ga/strings.xml`

Questions

- Please ask in the Discussion Forum