

IMMOBILIEN
SCOUT24

Navigation and the Single Activity

Learnings from a Skeptic



@askashdavies



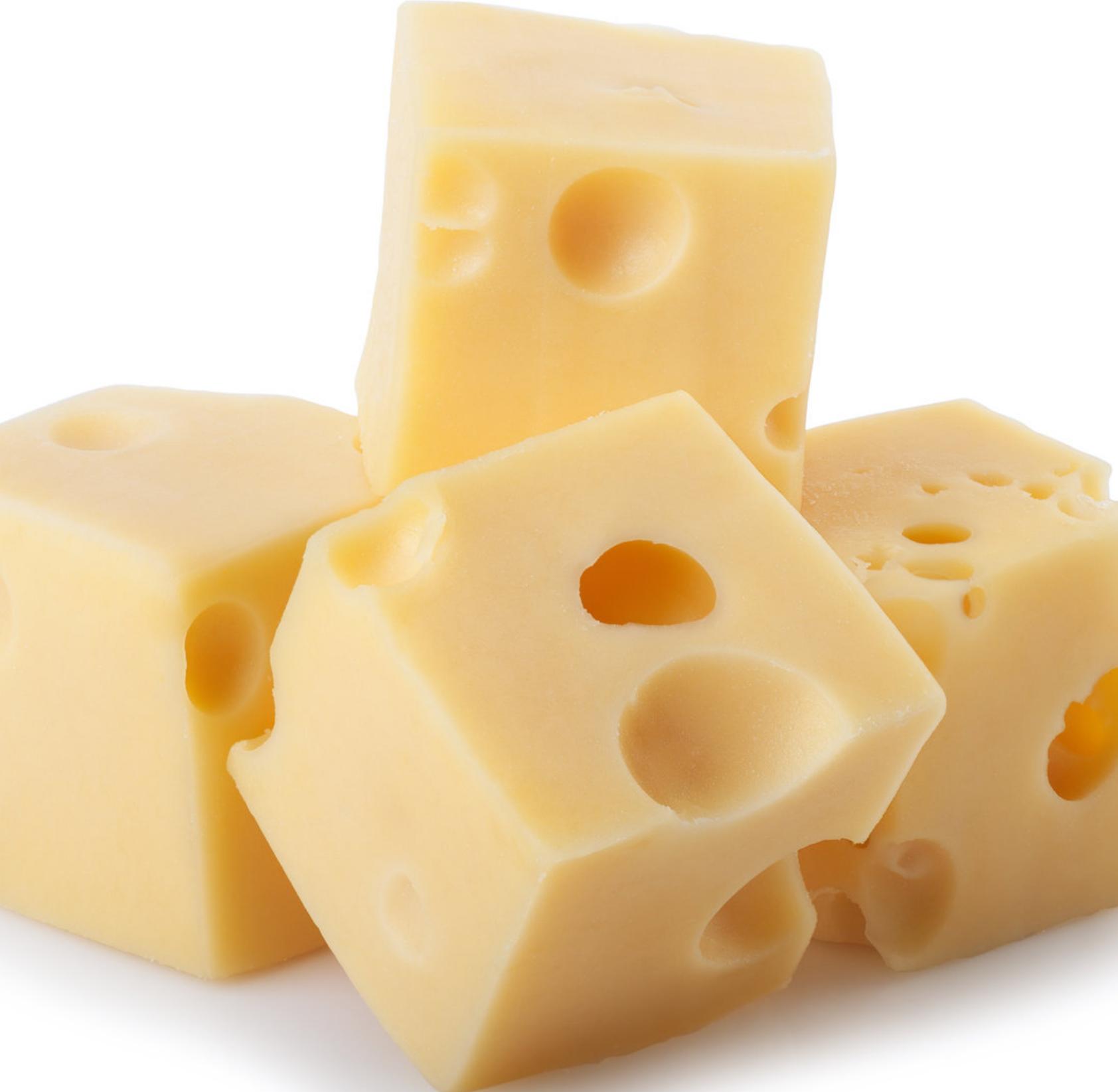
IMMOBILIEN
SCOUT24

Random Slides

Cheese



Android
GDE



@askashdavies

IMMOBILIEN
SCOUT24

Navigation and the Single Activity

Learnings from a Skeptic

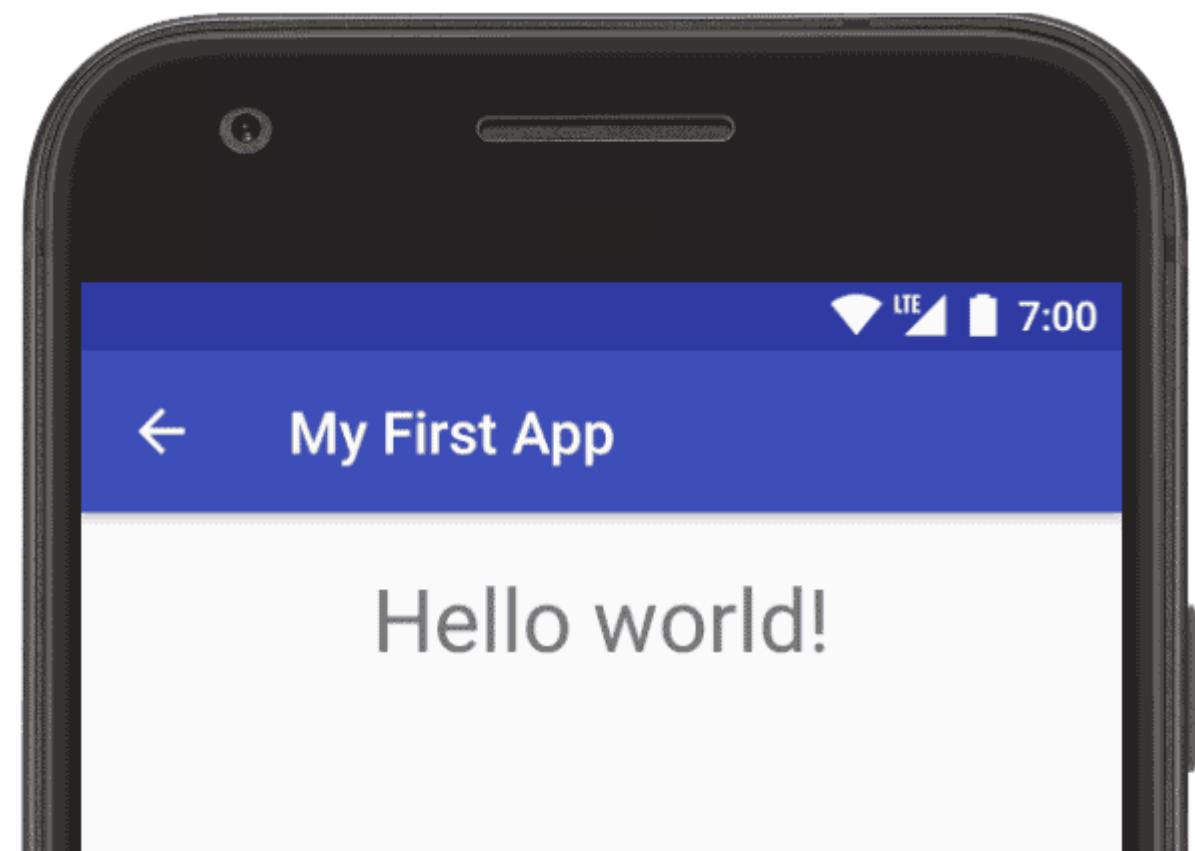
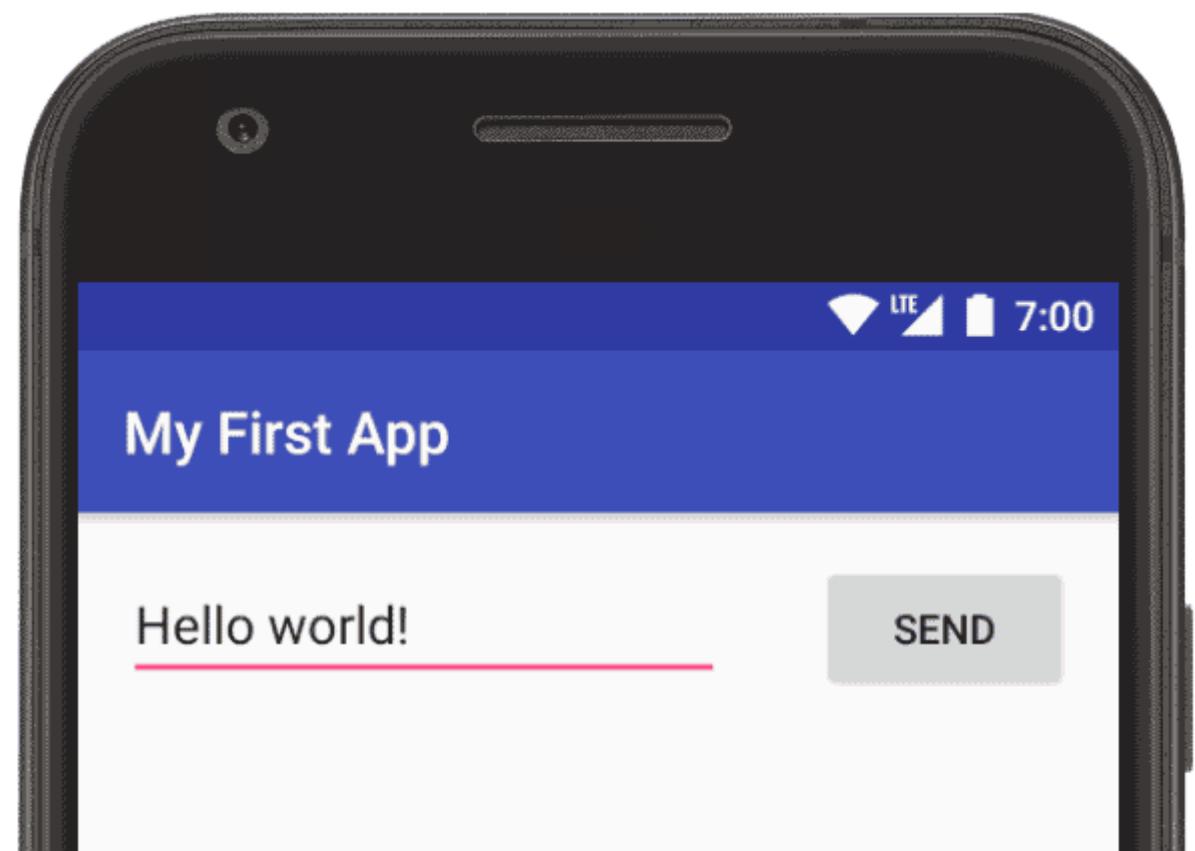


@askashdavies





@askashdavies



```
class MainActivity : AppCompatActivity {

    private val message: EditText
        get() = findViewById(R.id.message)

    override fun onCreate(savedInstanceState: Bundle) {
        /* ... */
    }

    fun sendMessage() {
        val intent = Intent(this, DisplayMessageActivity::class.java)
        intent.putExtra(DisplayMessageActivity.EXTRA_MESSAGE, message.text.toString())
        startActivity(intent)
    }
}
```


"Once we have gotten in to this entry-point to your UI, we really don't care how you organise the flow inside."

— *Dianne Hackborn, Android Framework team, 2016*



@askashdavies



Family



Find contacts

New



4 contacts

H



Home

K



K

T



Torin



東洸



★ K

Phone

(415) 555-1212

Home

Email

example@gmail.com

Home



Groups

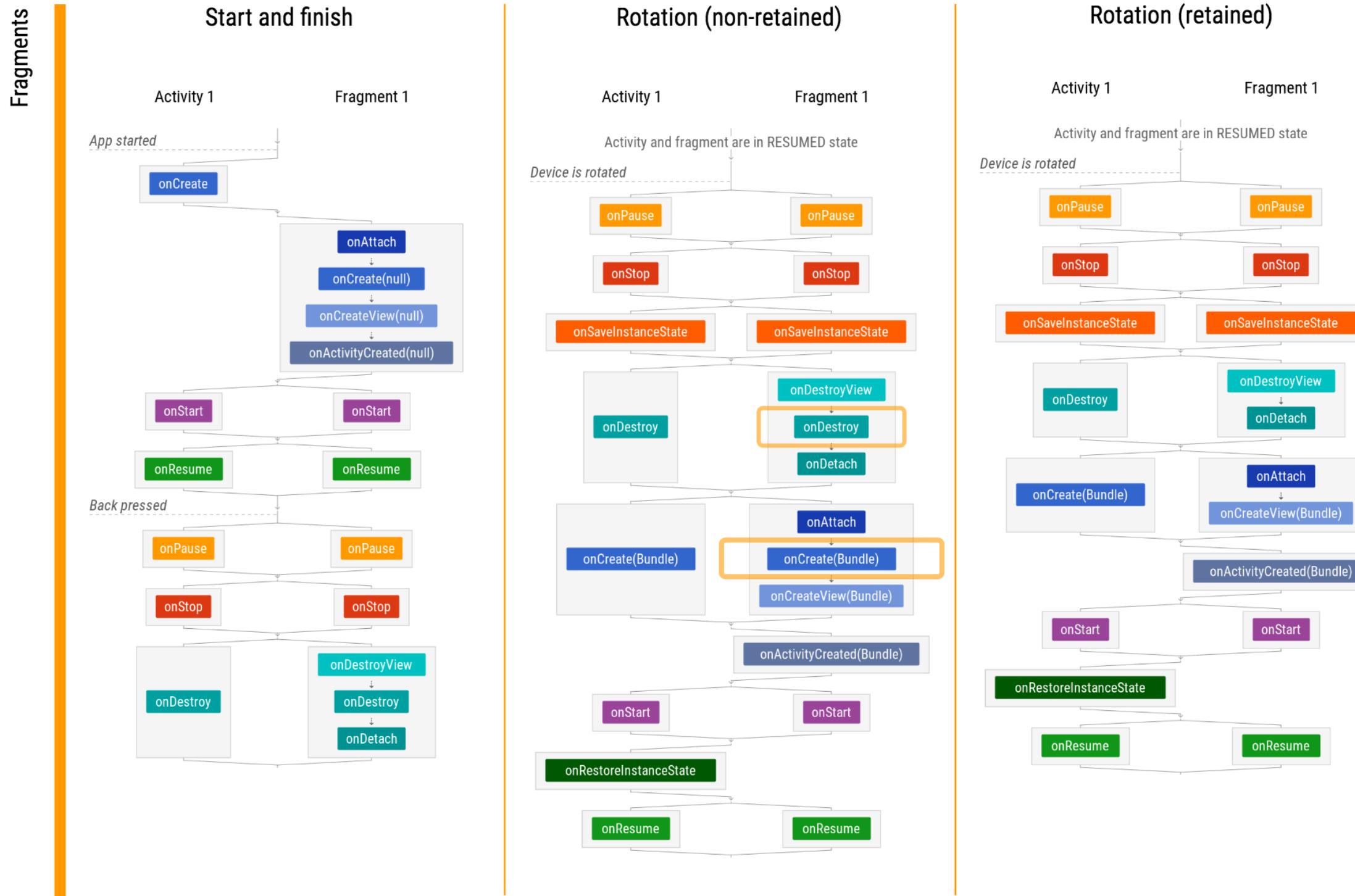
Family

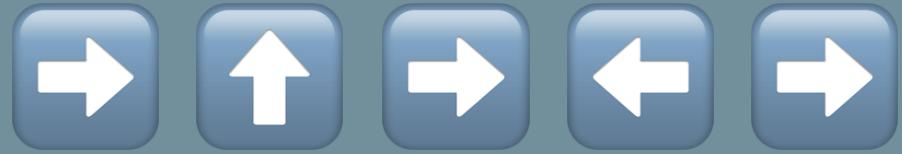


9:41 *

Activity Lifecycle != Fragment Lifecycle

Activity Lifecycle != Fragment Lifecycle != View Lifecycle





@askashdavies

(supportFragmentManager || childFragmentManager)?

Bundles



FragmentFactory

androidx.fragment:fragment:1.1.0-beta01

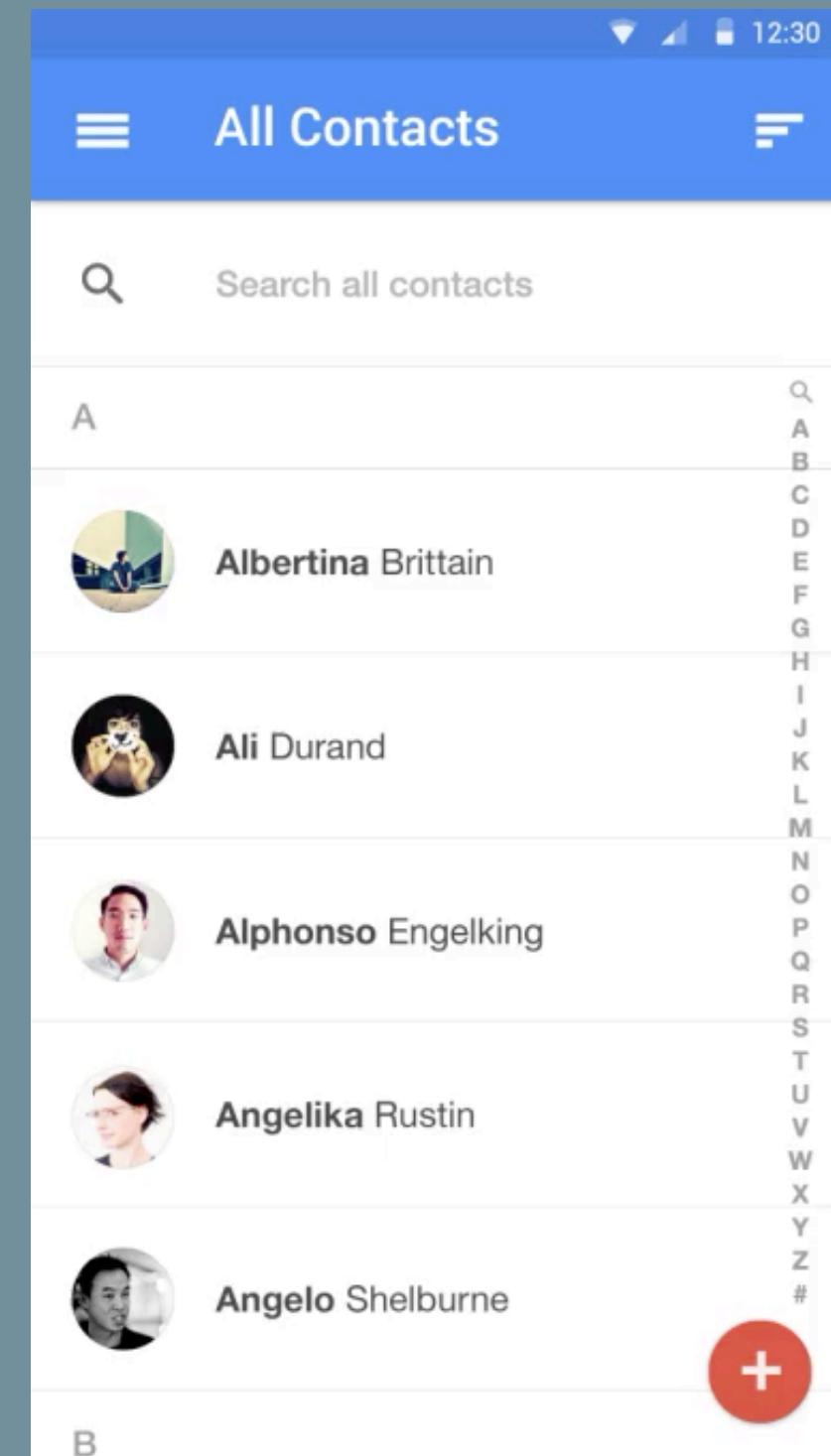
Fragments?

Fragments? 🤔

Activity Capabilities



Shared Element Transition (API 21+)



CompatShims < Enough

~~android.app.Fragment~~

androidx.fragment.app.Fragment

Scopes

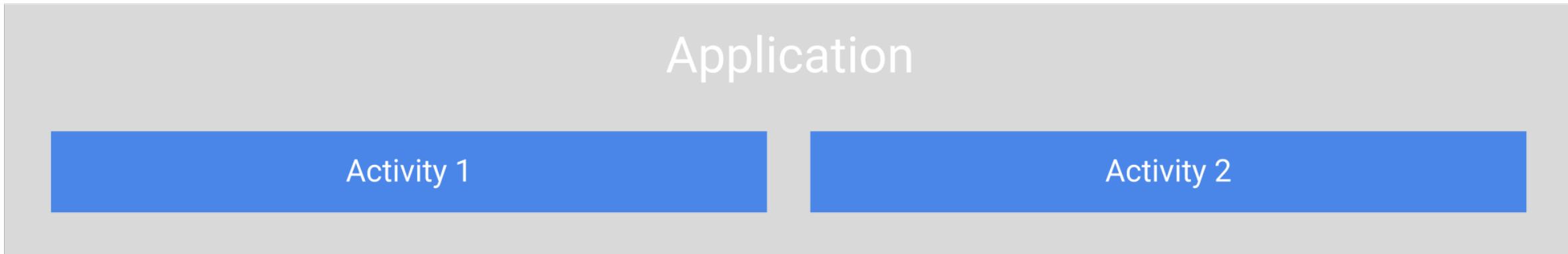


Scopes

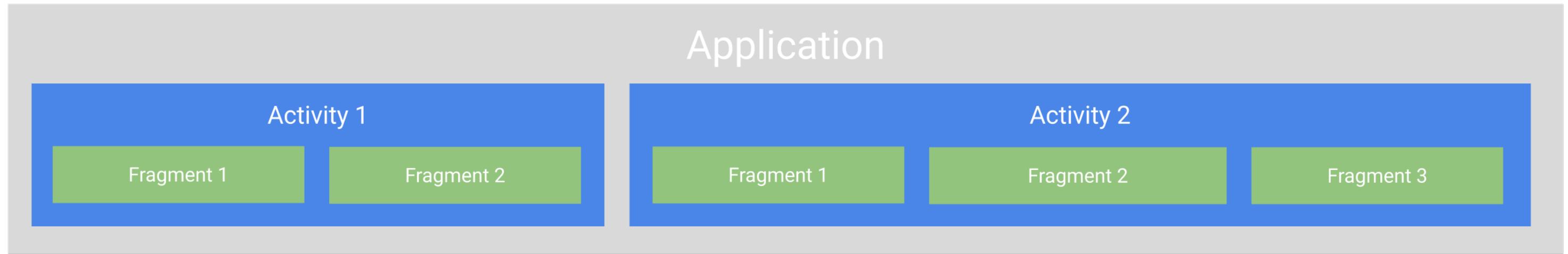
Activity 1

Activity 2

Scopes



Scopes





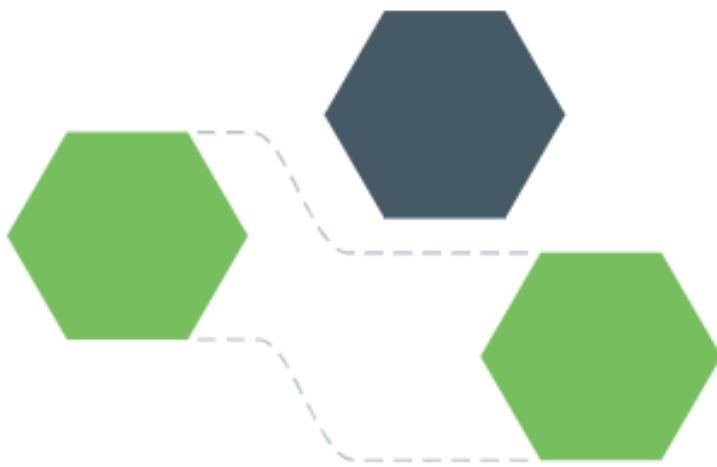
@askashdavies





@askashdavies

Android JetPack



Foundation

Android JetPack



Architecture

Android JetPack



Behavior

Android JetPack



UI

Android JetPack: Navigation



Android JetPack: Navigation

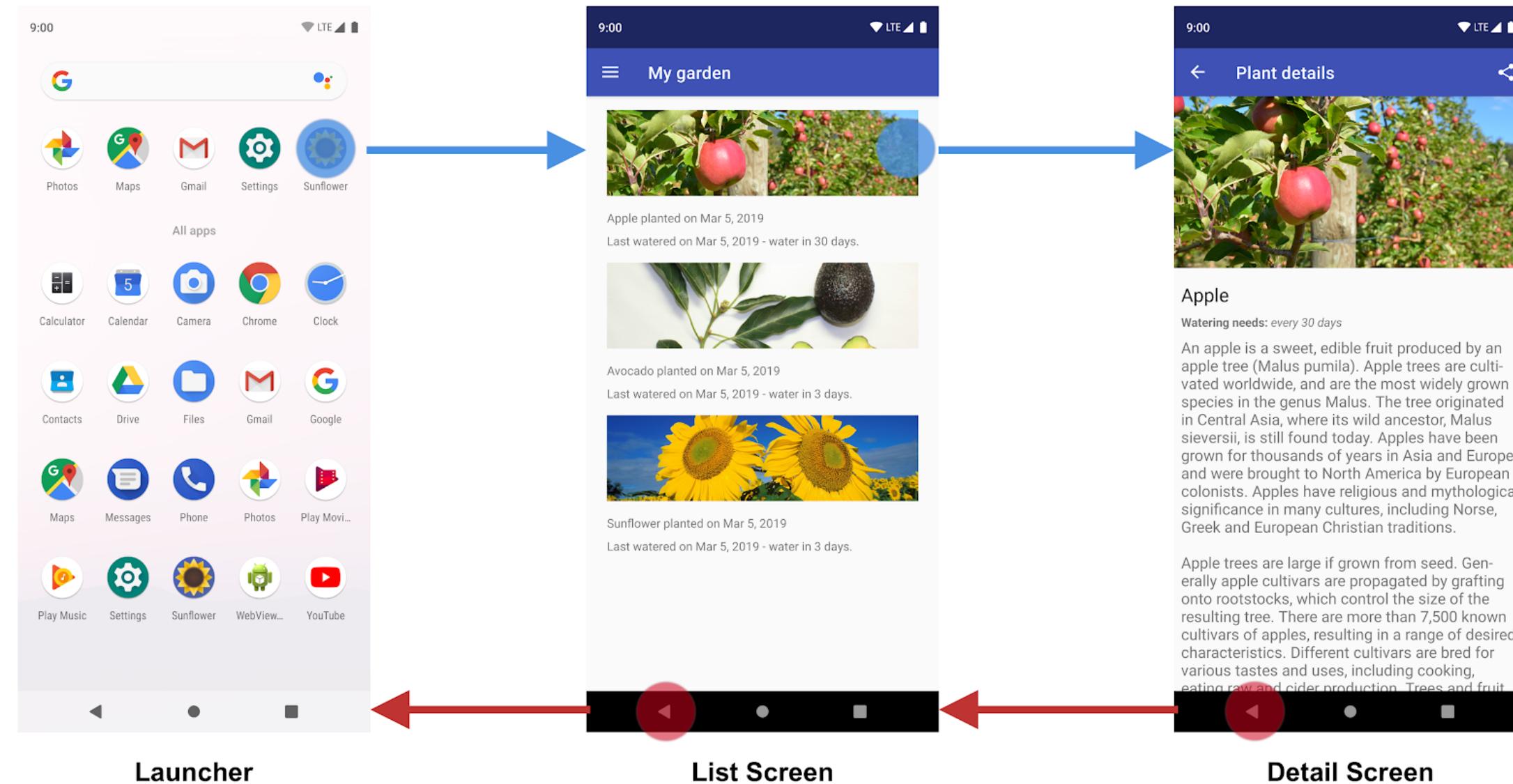


- Libraries 
- Plugin 
- Tooling 

- Fragment transactions
- Up and back actions
- Standardised transition resources
- Deep link implementation
- Easy navigation patterns
- Type safe arguments

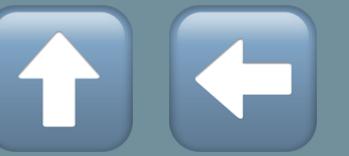
Principles of Navigation

Fixed Start Destination



State as Stack





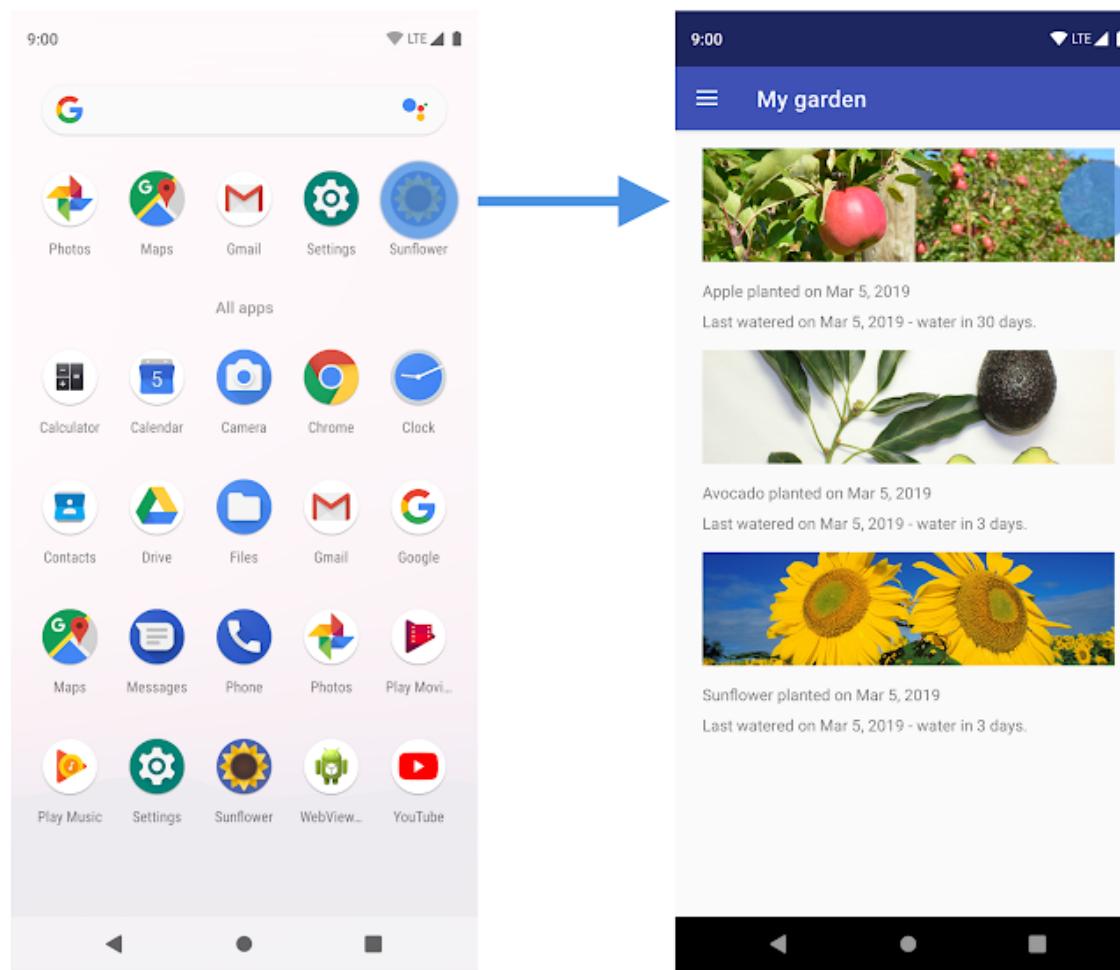
@askashdavies



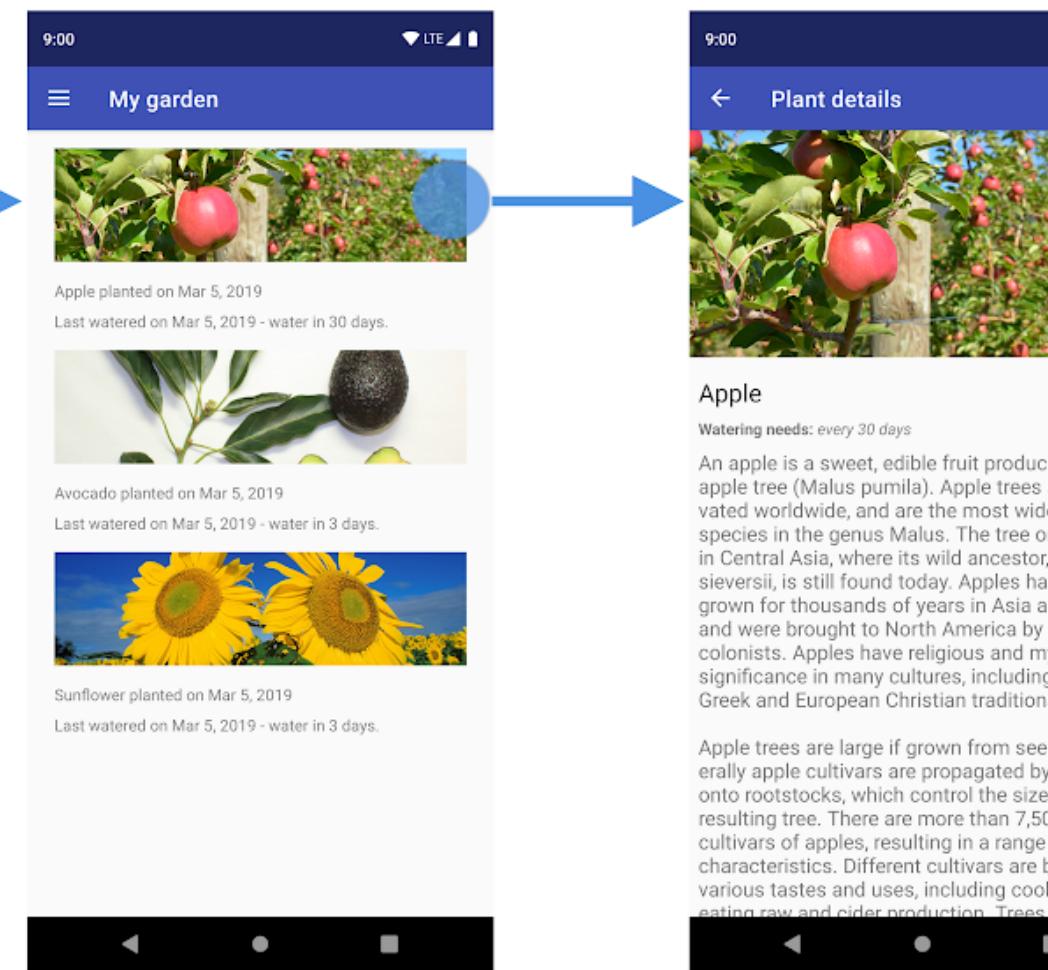
@askashdavies

Deep Link Simulates Manual Navigation

Organic Navigation through Sunflower (Browsing to apple details)



Launch Screen

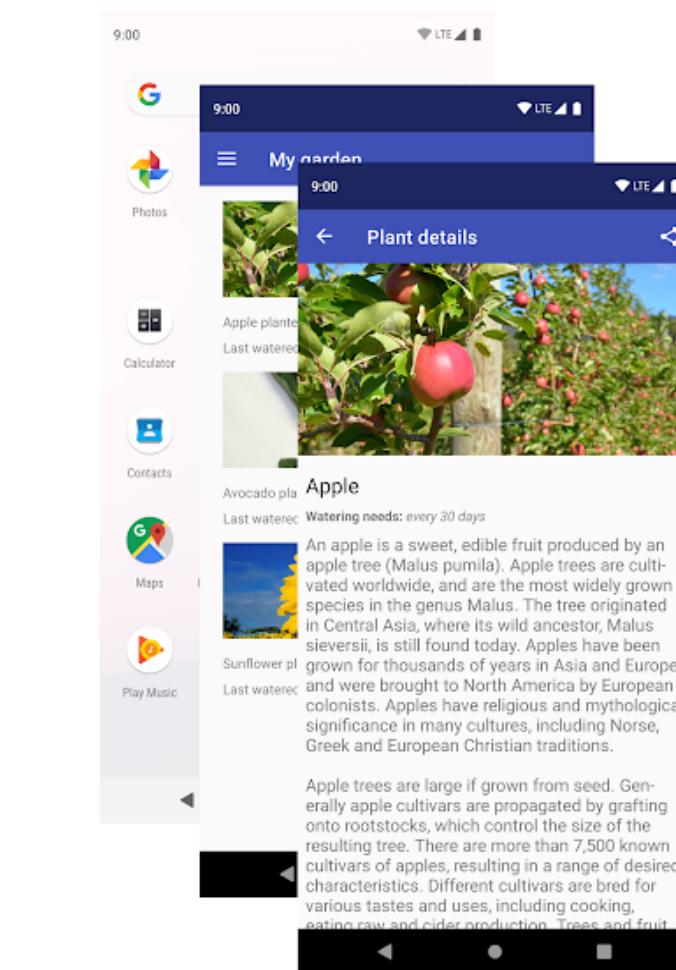


List Screen



Detail Screen

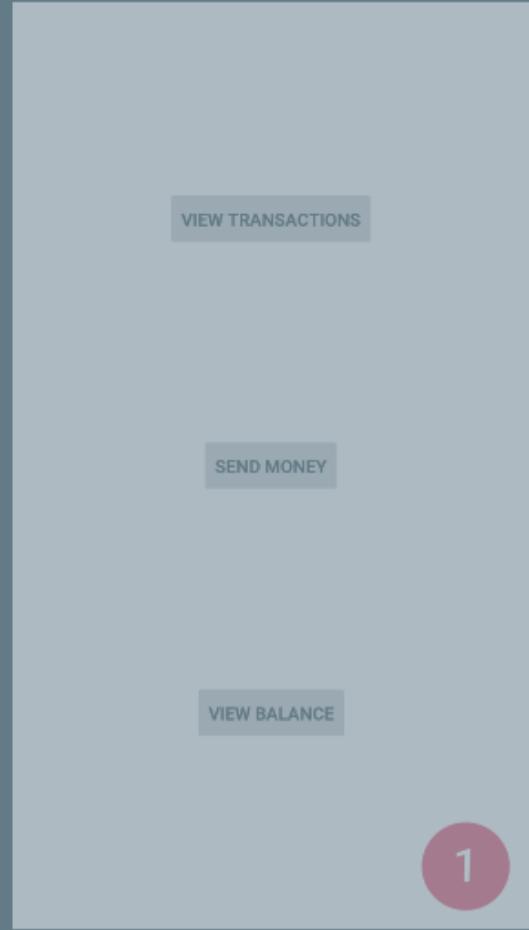
Resulting Sunflower Task Back Stack



JetPack: Navigation



fragmentMain



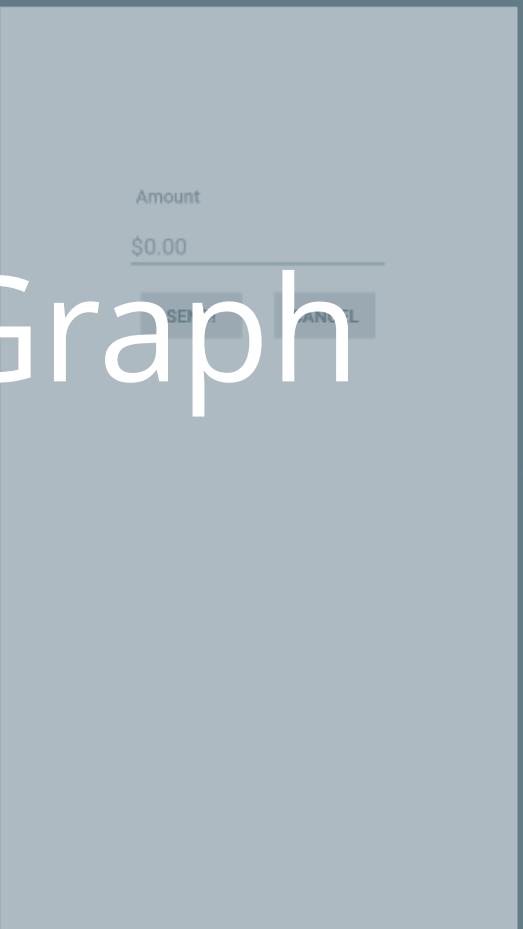
Candles \$3600

Utility \$200

fragmentChooseRecipi...



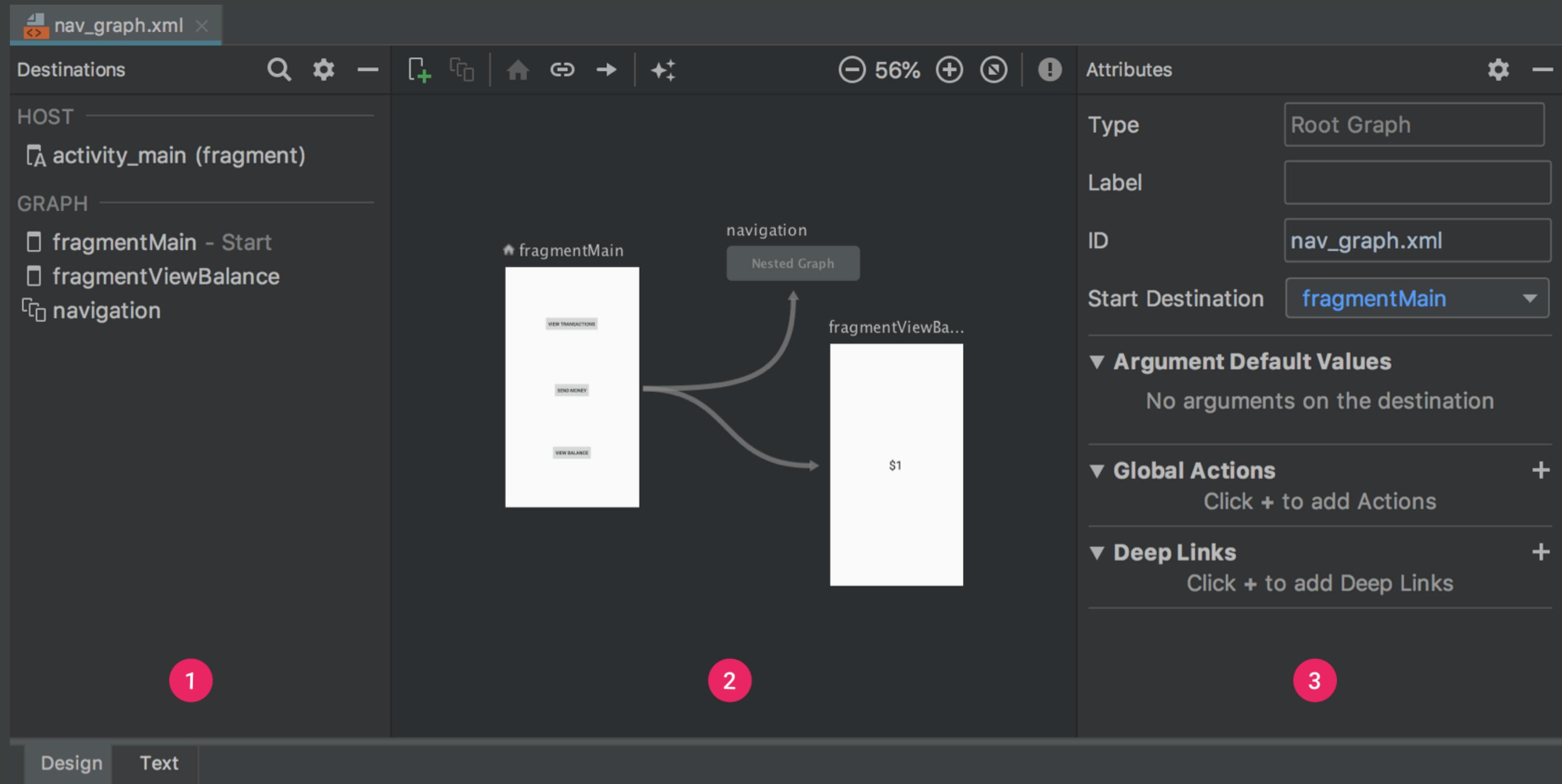
fragmentSpecifyAmount



fragmentConfirmation



Navigation Graph



Destination Types

- Activity <activity>
- Fragment <fragment>
- Dialog <dialog> (2.1.0+)

Destination Types (Custom)

developer.android.com/guide/navigation/navigation-add-new

- Extend Navigator<T> with your type
- Provide Destination implementation
- Configure arguments after inflation
- Augment NavController

nav_graph.xml

```
<navigation xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/navigation"
    app:startDestination="@+id/fragmentMain">

    <fragment
        android:id="@+id/fragmentMain"
        android:name="com.google.sample.MainFragment"
        android:label="@string/main_title"
        tools:layout="@layout/main_fragment">

        <action
            android:id="@+id/mainToViewBalance"
            app:destination="@+id/fragmentViewBalance"/>

    </fragment>

    <fragment
        android:id="@+id/fragmentViewBalance"
        android:name="com.google.sample.ViewBalanceFragment"
        android:label="@string/view_balance_title"
        tools:layout="@layout/view_balance_fragment" />

</navigation>
```

@askashdavies

Resource Inflation



NavHostFragment

```
<fragment  
    android:id="@+id/nav_host_fragment"  
    android:name="androidx.navigation.fragment.NavHostFragment"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    app:defaultNavHost="true"  
    app:navGraph="@navigation/nav_graph" />
```

NavController

- Fragment.findNavController()
- View.findNavController()
- Activity.findNavController(viewId: Int)

Actions

```
viewTransactionsButton.setOnClickListener { view ->  
    view.findNavController().navigate(R.id.viewTransactionsAction)  
}  
  
button.setOnClickListener(  
    Navigation.createNavigateOnClickListener(R.id.next_fragment, null)  
)
```

Deep Links 🔥

Deep Links 🔥

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.myapplication">

    <application ... >

        <activity name=".ProfileActivity" ...>
            ...
            <nav-graph android:value="@navigation/main.xml" />
            ...
        </activity>
    </application>
</manifest>
```

Deep Links 🔥

```
<fragment  
    android:id="@+id/profile"  
    android:name=".ProfileActivity">  
  
    <deepLink app:uri="www.example.com/profile/{userId}" />  
</fragment>
```

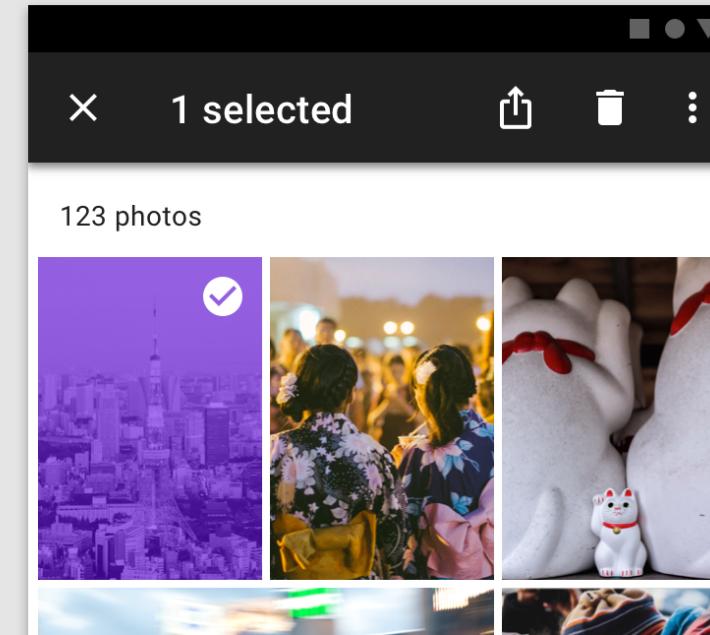
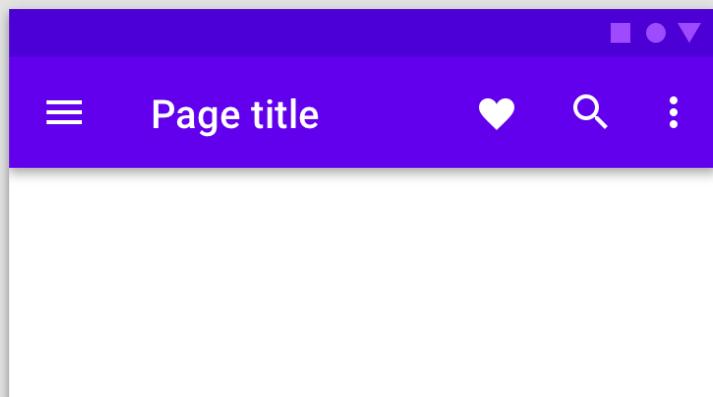
Navigation Styles



Navigation Styles



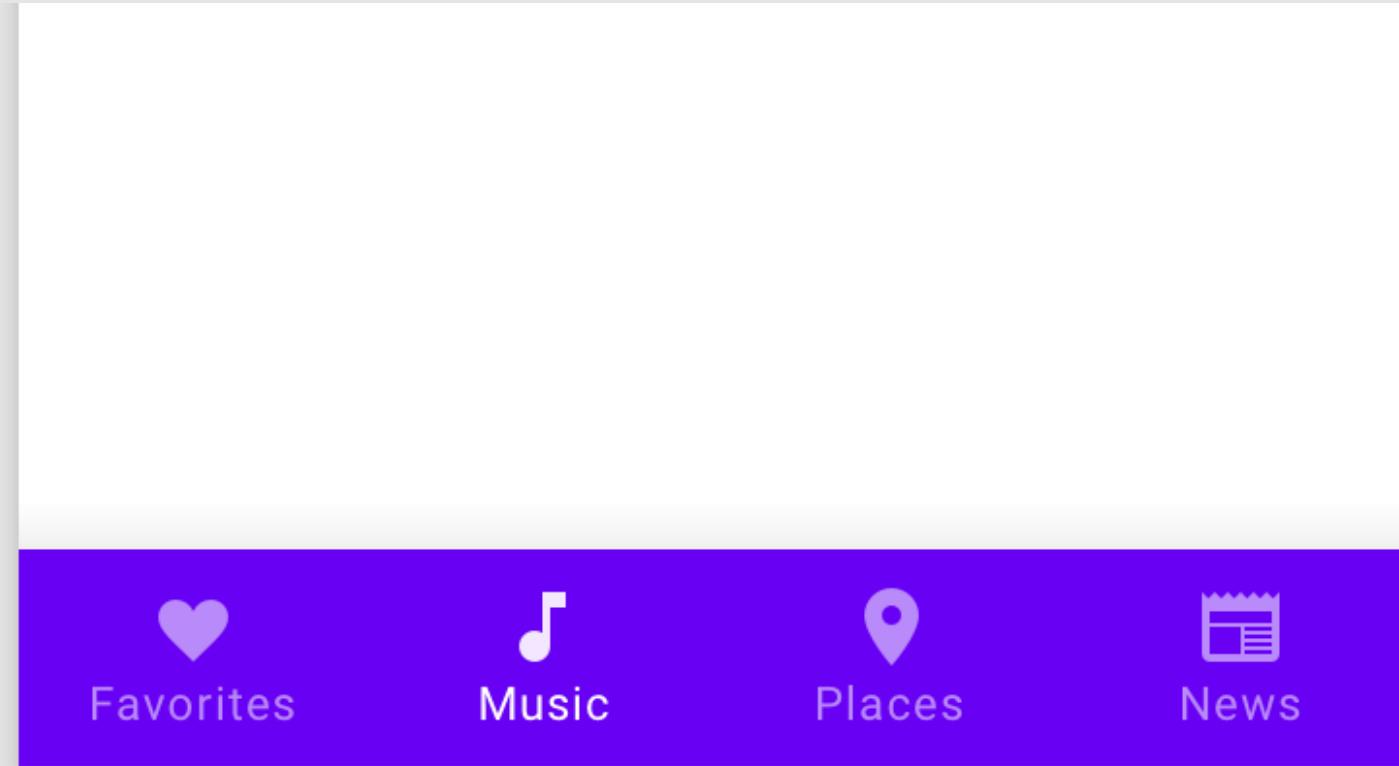
Toolbar



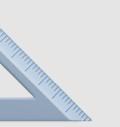
Navigation Styles



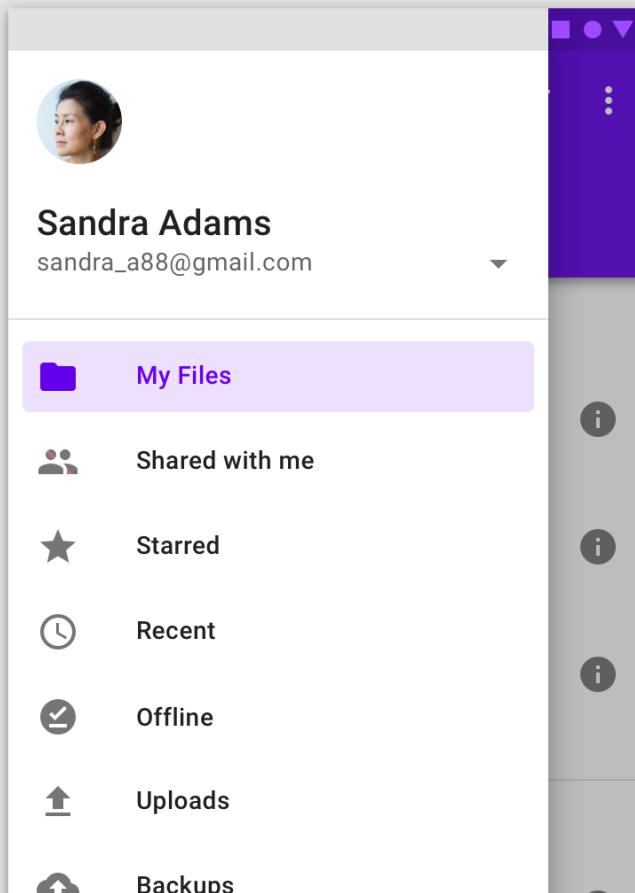
Bottom



Navigation Styles



Drawer



NavigationUI

`setupWithNavController()`

```
class AwesomeActivity : AppCompatActivity() {

    private val controller: NavController by lazy(NONE) { findNavController(this, R.id.host) }
    private val drawer: DrawerLayout by lazy(NONE) { findViewById<DrawerLayout>(R.id.drawer) }
    private val toolbar: Toolbar by lazy(NONE) { findViewById<Toolbar>(R.id.toolbar) }

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.awesome_activity)

        toolbar.setupWithNavController(controller, drawer)
    }
}
```

NavigationUI

```
NavigationUI.setupActionBarWithNavController(  
    activity: AppCompatActivity,  
    controller: NavController,  
    configuration: AppBarConfiguration  
)
```

```
NavigationUI.setupWithNavController(  
    toolbar: Toolbar,  
    controller: NavController,  
    configuration: AppBarConfiguration  
)
```

NavigationUI: AppBarConfiguration

- Top level destinations
- Drawer layout
- Fallback "up" listener

OnDestinationChangedListener

NavController.addOnDestinationChangedListener

```
interface OnDestinationChangedListener {  
  
    fun onDestinationChanged(  
        controller: NavController,  
        destination: NavDestination,  
        arguments: Bundle?  
    );  
}
```

Bundles



Honourable Mention

Eugenio Marletti: Android Extras Delegates

github.com/Takhion/android-extras-delegates

Android Extras Delegates

```
class SomeActivity : Activity() {  
  
    companion object : ActivityCompanion<IntentOptions>(  
        intentOptions = IntentOptions,  
        kclass = SomeActivity::class  
    )  
  
    object IntentOptions {  
  
        var Intent.someExtra by IntentExtra.String()  
    }  
}
```

SafeArgs 💪

SafeArgs: Directions

MainFragmentDirections.mainToViewBalance()

```
<fragment
    android:id="@+id/fragmentMain"
    android:name="com.google.sample.MainFragment"
    android:label="@string/main_title"
    tools:layout="@layout/main_fragment">

    <action
        android:id="@+id/mainToViewBalance"
        app:destination="@+id/fragmentViewBalance"/>

</fragment>
```

SafeArgs: Directions

```
<fragment
    android:id="@+id/fragmentViewBalance"
    android:name="com.google.sample.ViewBalanceFragment"
    android:label="@string/view_balance_title"
    tools:layout="@layout/view_balance_fragment">

<argument
    android:name="balanceAmount"
    app:argType="integer"/>

</fragment>
```

SafeArgs: Args

```
class MainFragment: Fragment() {  
  
    override fun onViewCreated(view: View, savedInstanceState: Bundle?) {  
        super.onViewCreated(view, savedInstanceState)  
  
        viewTransactionsButton.setOnClickListener { view ->  
            MainFragmentDirections.mainToViewBalance(100)  
        }  
    }  
}  
  
class ViewBalanceFragment : Fragment() {  
  
    override fun onViewCreated(view: View, savedInstanceState: Bundle?) {  
        super.onViewCreated(view, savedInstanceState)  
  
        val args = ViewBalanceFragmentArgs.fromBundle(arguments!!)  
        balance.setText(String.format("%.2d", args.balanceAmount))  
    }  
}
```

SafeArgs: Generated Args



```
data class ViewBalanceFragmentArgs(val balanceAmount: Int) : NavArgs {  
    @Suppress("CAST_NEVER_SUCCEEDS")  
    fun toBundle(): Bundle {  
        val result = Bundle()  
        result.putInt("balanceAmount", this.balanceAmount)  
        return result  
    }  
  
    companion object {  
        @JvmStatic  
        fun fromBundle(bundle: Bundle): FinanceBorrowerFragmentArgs {  
            bundle.setClassLoader(FinanceBorrowerFragmentArgs::class.java.classLoader)  
            val __balanceAmount : Int  
            if (bundle.containsKey("balanceAmount")) {  
                __balanceAmount = bundle.getInt("balanceAmount")  
            } else {  
                throw IllegalArgumentException("Required argument \\"balanceAmount\\" is missing and does not have an android:defaultValue")  
            }  
            return FinanceBorrowerFragmentArgs(__balanceAmount)  
        }  
    }  
}
```

SafeArgs: Generated Directions



```
class MainFragmentDirections private constructor() {
    private data class MainToViewBalance(val balanceAmount: Int) : NavDirections {
        override fun getActionId(): Int = R.id.mainToViewBalance

        @Suppress("CAST_NEVER_SUCCEEDS")
        override fun getArguments(): Bundle {
            val result = Bundle()
            result.putInt("balanceAmount", this.index)
            return result
        }
    }

    companion object {
        fun mainToViewBalance(balanceAmount: Int): NavDirections = BorrowerToBorrower(index, financeBorrower)
    }
}
```

Single Source of Truth

SafeArgs: Activities



SafeArgs: Activities

```
class MainActivity : Activity {  
  
    override fun onCreate(savedInstanceState: Bundle) {  
        super.onCreate(savedInstanceState)  
  
        val args = ViewBalanceActivityArgs(100)  
        val intent = Intent(this, ViewBalanceActivity::class.java)  
  
        startActivity(intent, bundle.toBundle())  
    }  
}
```

SafeArgs: Getting Started

```
buildscript {  
    repositories {  
        google()  
    }  
  
    dependencies {  
        classpath "androidx.navigation:navigation-safe-args-gradle-plugin:2.0.0"  
    }  
}  
  
apply plugin: 'androidx.navigation.safeargs.kotlin'
```



Migrating



Processed Cheddar



Aragon



Blue Vein Cheeses



Castiglano



Allgauer Emmentaler



Schloss



Roncal



Fondant de Brebis



White Stilton

@askashdavies



Camembert



Migrating



Stilton



Info

Jarlsberg lancashire edam. Dolcelatte hard cheese brie st. agur blue cheese caerphilly bavarian bergkase cheese and biscuits mascarpone. Cheeseburger swiss bavarian bergkase cream cheese fromage frais cheesy feet port-salut airedale. St. agur blue cheese rubber cheese caerphilly cheddar cheesecake cream cheese manchego lancashire. Roquefort squirty cheese the big cheese.

Friends

Jarlsberg lancashire edam. Dolcelatte hard cheese brie st. agur blue cheese caerphilly bavarian bergkase cheese and biscuits mascarpone. Cheeseburger swiss bavarian bergkase cream cheese fromage frais cheesy feet port-salut airedale. St. agur blue cheese rubber cheese caerphilly cheddar cheesecake cream cheese manchego lancashire. Roquefort squirty cheese the big cheese.



Migrating

Move screen behaviour away from activities

 Migrating

Create new activity for Fragment's



Migrating

Move existing activity logic to fragment

- FragmentBindings -> ActivityBindings
- onCreate() -> onCreateView()
- onViewCreated()
- getViewLifecycleOwner()



Migrating

Initialise fragment in host activity

Initialise fragment in host activity

```
override fun onCreate(savedInstanceState: Bundle?) {  
    super.onCreate(savedInstanceState)  
    setContentView(R.layout.cheese_activity)  
  
    if (savedInstanceState == null) {  
        supportFragmentManager  
            .beginTransaction()  
            .add(R.id.main_content, CheeseListFragment())  
            .commit()  
    }  
  
    fun navigateToCheeseDetail(productId: String) {  
        ...  
    }  
}
```

Pass arguments as necessary

```
override fun onCreate(savedInstanceState: Bundle?) {  
    super.onCreate(savedInstanceState)  
    setContentView(R.layout.cheese_activity)  
  
    if (savedInstanceState == null) {  
        val fragment = CheeseListFragment() // 🧀  
        fragment.arguments = intent.extras  
  
        supportFragmentManager  
            .beginTransaction()  
            .add(R.id.main_content, fragment)  
            .commit()  
    }  
  
    fun navigateToCheeseDetail(productId: String) {  
        ...  
    }  
}
```



Migrating

Create navigation graph

Create navigation graph

```
<navigation xmlns:android="http://schemas.android.com/apk/res/android"  
    xmlns:app="http://schemas.android.com/apk/res-auto"  
    app:startDestination="@+id/cheeseGraph">  
  
<fragment  
    android:id="@+id/cheeseListFragment"  
    android:name="com.sample.CheeseListFragment"  
    android:label="@string/cheese_list_title"  
    tools:layout="@layout/cheese_list_fragment" />  
  
<fragment  
    android:id="@+id/cheeseDetailsFragment"  
    android:name="com.sample.CheeseDetailsFragment"  
    android:label="@string/cheese_details_title"  
    tools:layout="@layout/cheese_details_fragment" />  
  
</navigation>
```

Create navigation host

```
<fragment  
    android:id="@+id/main_content"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:name="androidx.navigation.fragment.NavHostFragment"  
    app:navGraph="@navigation/cheese_list_graph"  
    app:defaultNavHost="true" />
```



Migrating

```
class CheeseHostActivity : AppCompatActivity() {  
  
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        setContentView(R.layout.cheese_activity)  
    }  
}
```

Navigation Directions

```
<navigation xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    app:startDestination="@+id/cheeseGraph">

    <fragment
        android:id="@+id/cheeseListFragment"
        android:name="com.sample.CheeseListFragment"
        android:label="@string/cheese_list_title"
        tools:layout="@layout/cheese_list_fragment">

        <action
            android:id="@+id/navigateToCheeseDetails"
            app:destination="@+id/cheeseDetailsFragment"/>

    </fragment>

    <fragment
        android:id="@+id/cheeseDetailsFragment"
        android:name="com.sample.CheeseDetailsFragment"
        android:label="@string/cheese_details_title"
        tools:layout="@layout/cheese_details_fragment">

        <argument
            android:name="cheeseId"
            app:argType="string" />

    </fragment>
</navigation>
```

CheeseListFragment

```
class CheeseListFragment : Fragment() {  
    ...  
    override fun onViewCreated(view: View, savedInstanceState: Bundle?) {  
        cheeseAdapter = CheeseAdapter(cheeseClickCallback)  
        binding.productsList.setAdapter(productAdapter)  
    }  
    ...  
  
    // The callback makes the call to the activity to make the transition.  
    private val productClickCallback = ProductClickCallback { cheese ->  
        val directions = CheeseListDirections.navigateToCheeseDetails(cheese.id)  
        findNavController().navigate(directions)  
    }  
}
```



Profit

onActivityResult?

onActivityResult

<https://issuetracker.google.com/issues/79672220>

Jose Alcérreca: LiveData<Event<T>>

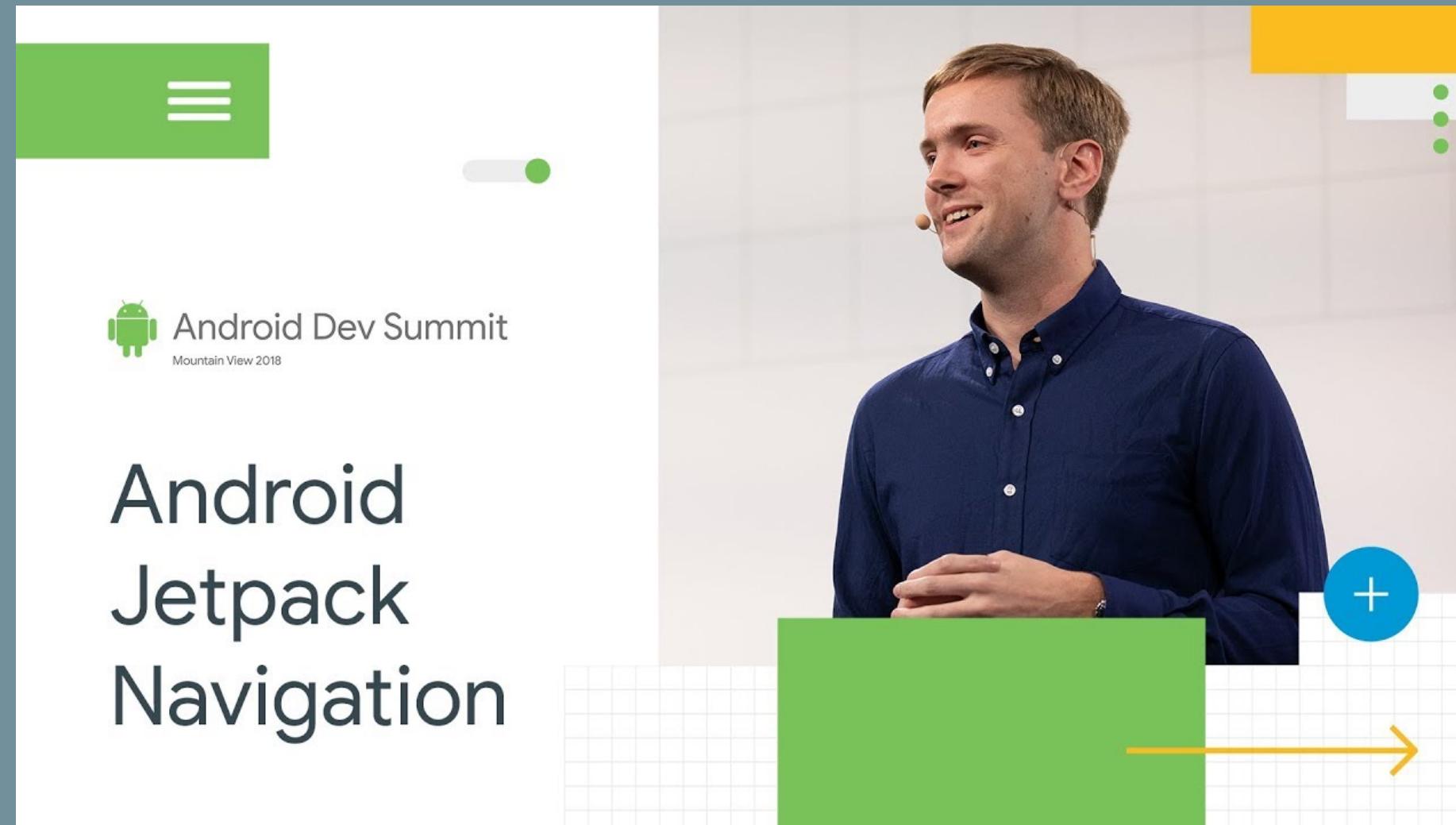
<http://bit.ly/2YuSYXi>

Dynamic Features

Coming Soon? ™

Ian Lake: Single Activity: Why, When, and How

bit.ly/2Jo94x9



Thanks!

@askashdavies

