

IMMOBILIEN
SCOUT24

Android JetPack

Navigation Component



@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)
    }
}
```



@askashdavies



Scoping

Application

Activity 1

Activity 2

Activity 3

"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**



@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 *

```
public class MainActivity extends FragmentActivity {

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.news_articles);

        // Check that the activity is using the layout version with
        // the fragment_container FrameLayout
        if (findViewById(R.id.fragment_container) != null) {

            // However, if we're being restored from a previous state,
            // then we don't need to do anything and should return or else
            // we could end up with overlapping fragments.
            if (savedInstanceState != null) {
                return;
            }

            // Create a new Fragment to be placed in the activity layout
            HeadlinesFragment firstFragment = new HeadlinesFragment();

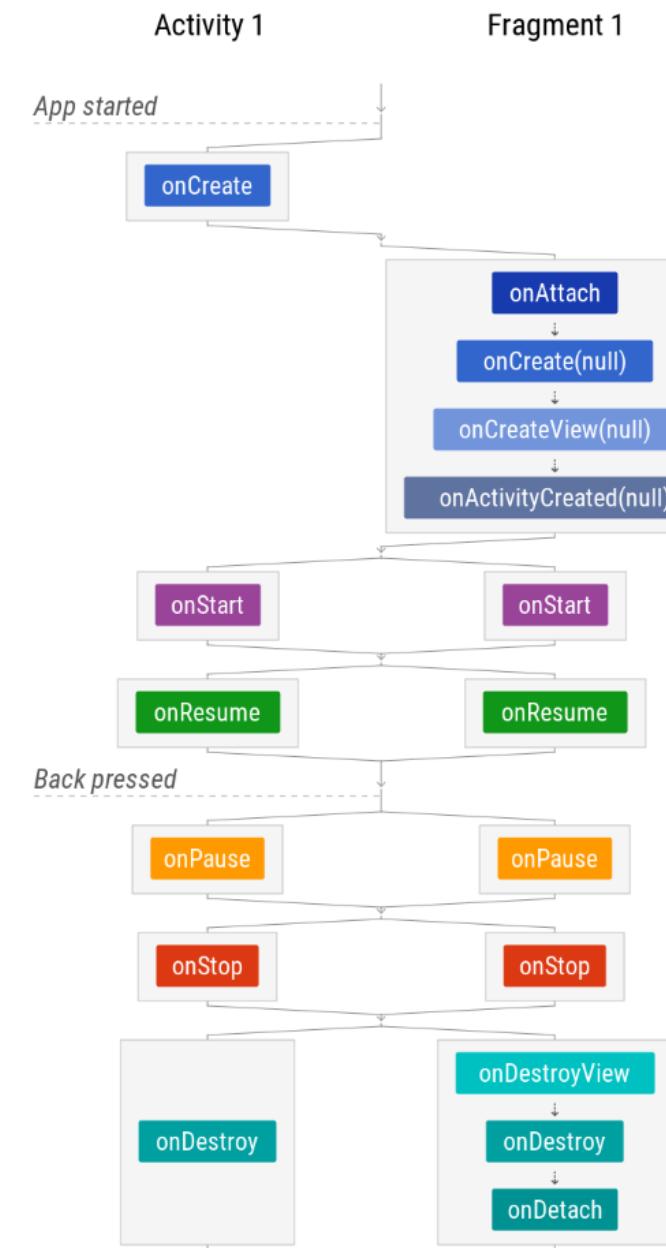
            // In case this activity was started with special instructions from an
            // Intent, pass the Intent's extras to the fragment as arguments
            firstFragment.setArguments(getIntent().getExtras());

            // Add the fragment to the 'fragment_container' FrameLayout
            getSupportFragmentManager()
                .beginTransaction()
                .add(R.id.fragment_container, firstFragment)
                .commit();
        }
    }
}
```

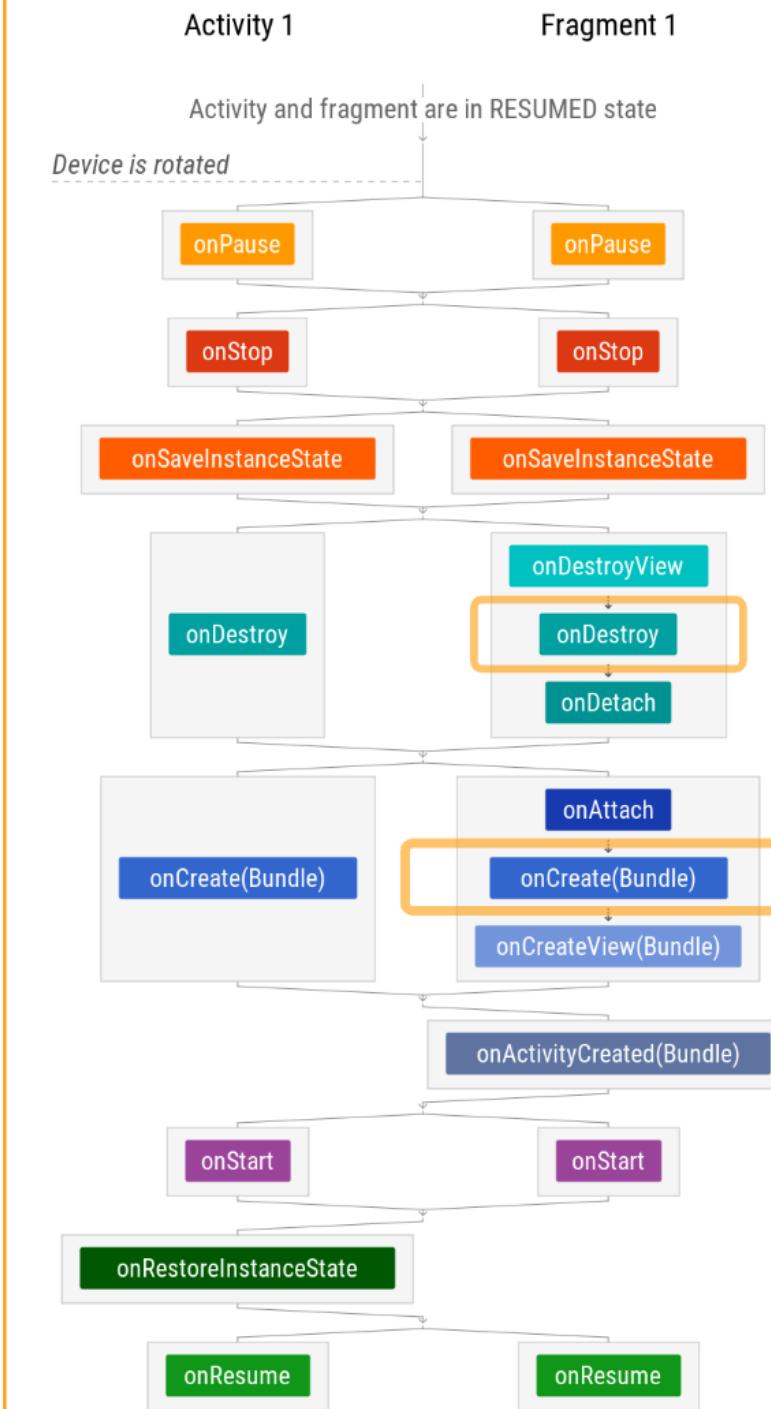
@askashdavies

Fragments

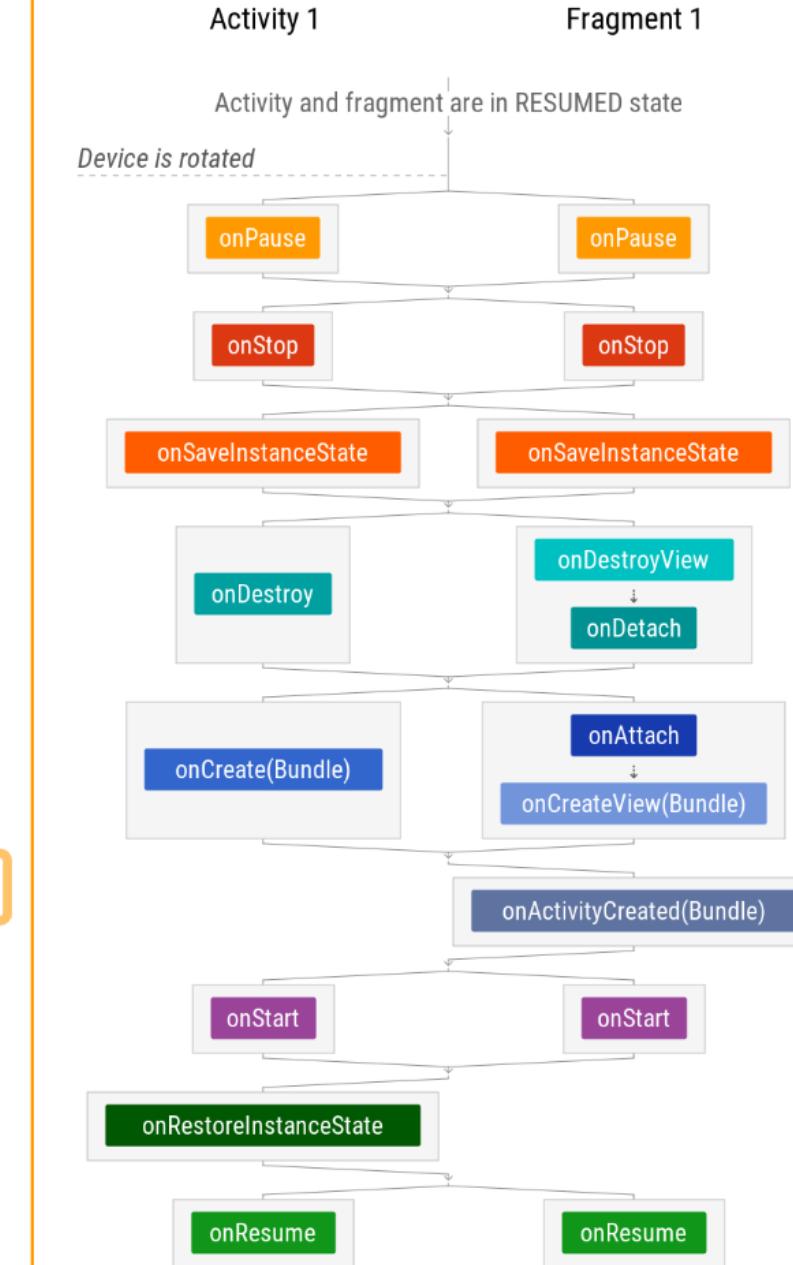
Start and finish

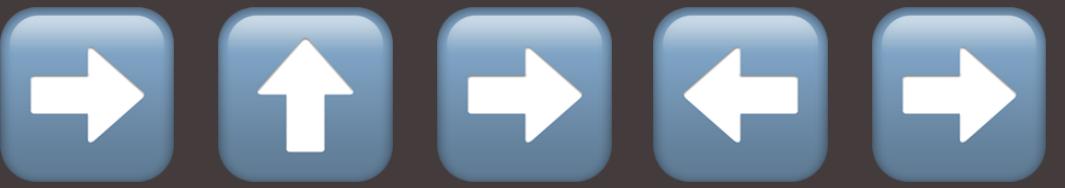


Rotation (non-retained)



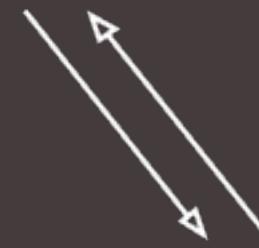
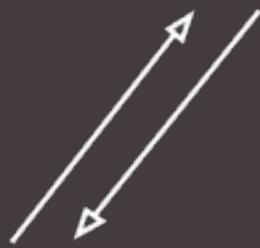
Rotation (retained)





@askashdavies

Activity



Fragment 1

Fragment 2

JetPack: Navigation





Libraries



Plugin

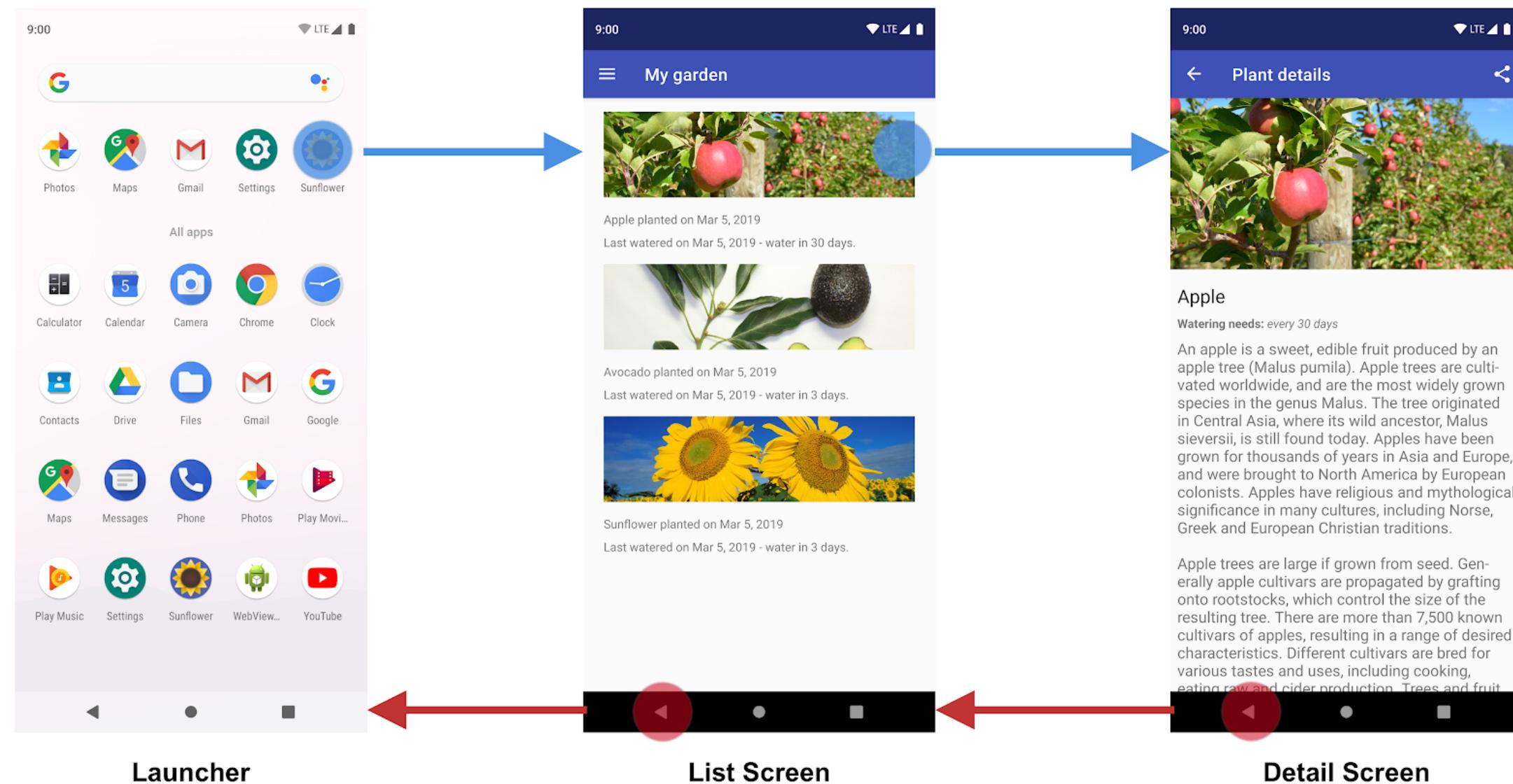


Tooling

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

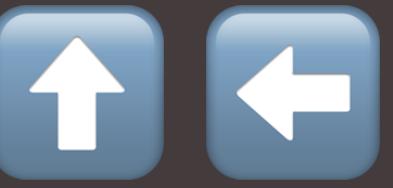
Principles of Navigation

Fixed Start Destination





@askashdavies



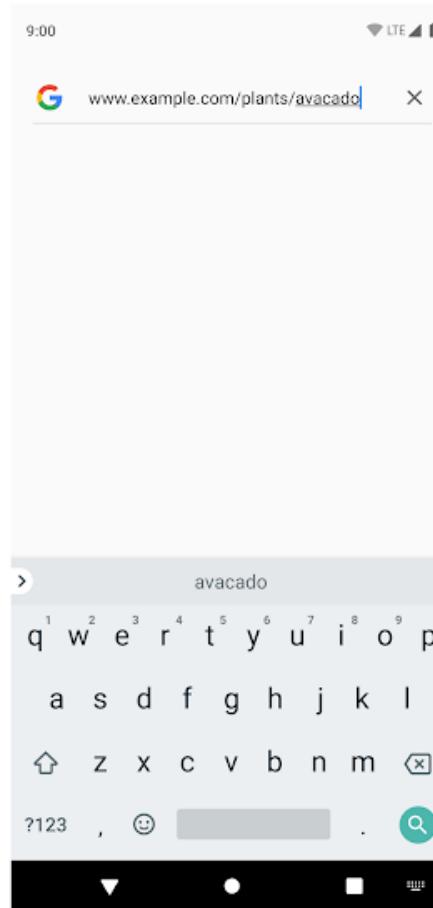
@askashdavies



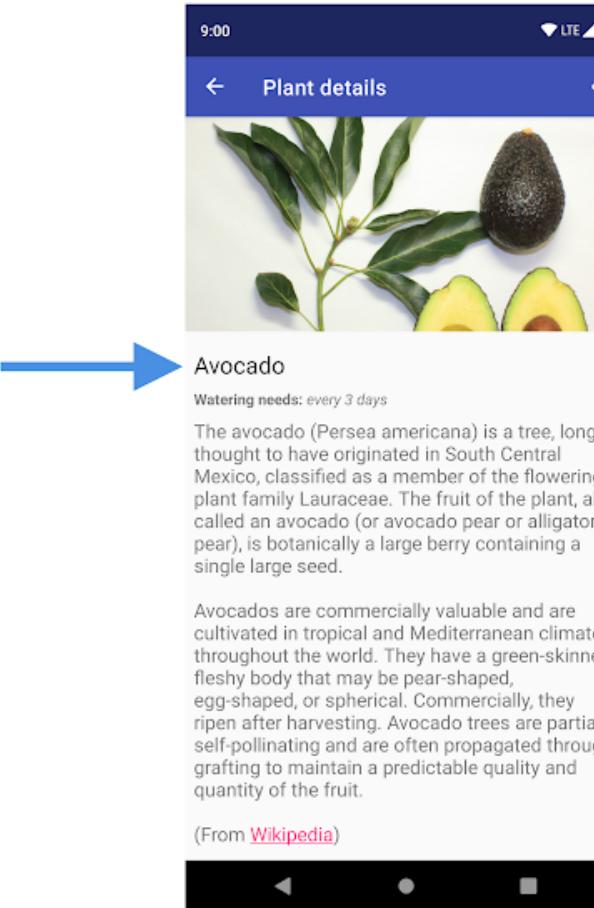
@askashdavies

Deep linking simulates manual navigation

Deep Link into Plant Details (Linking to avocado details)

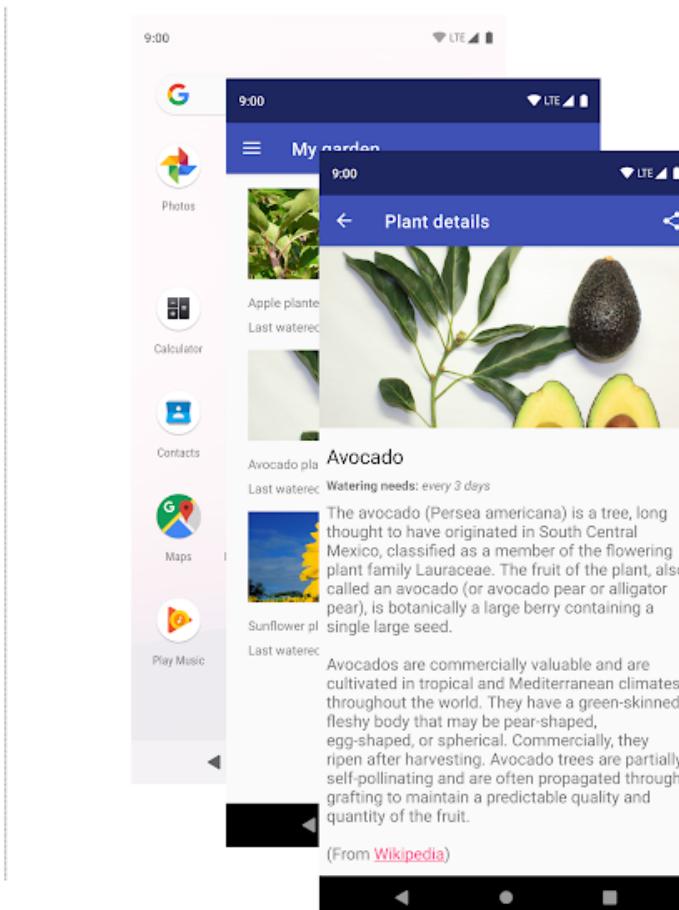


Follow Deep Link



Detail Screen

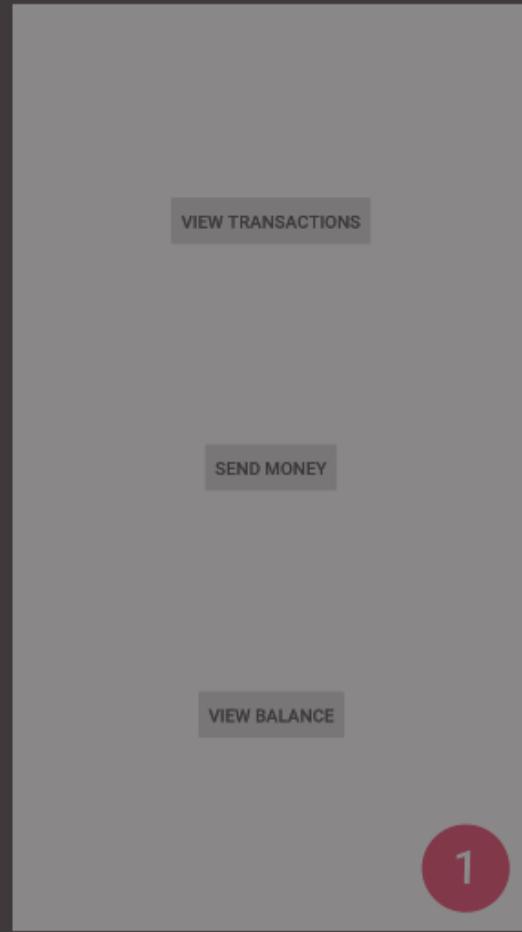
Resulting Sunflower Task Back Stack
(after user enters through deep link)



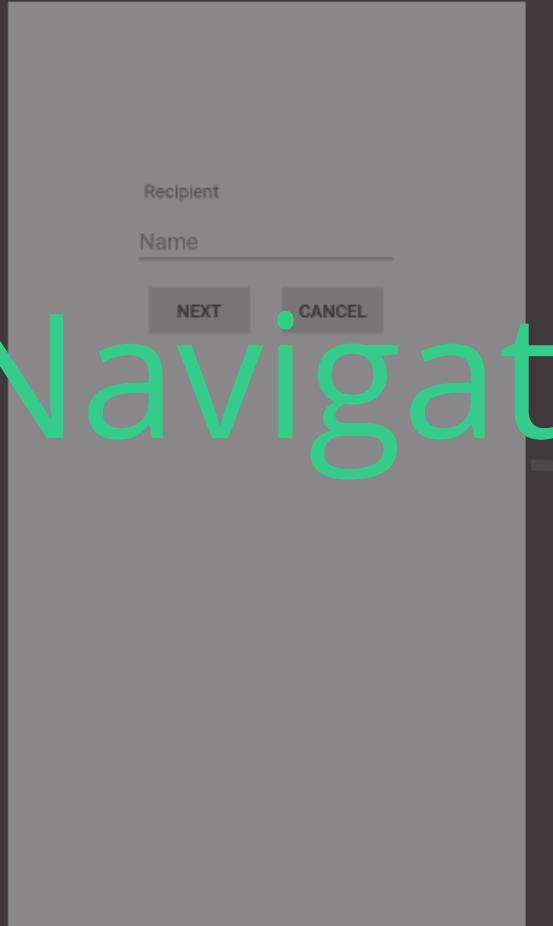
JetPack: Navigation



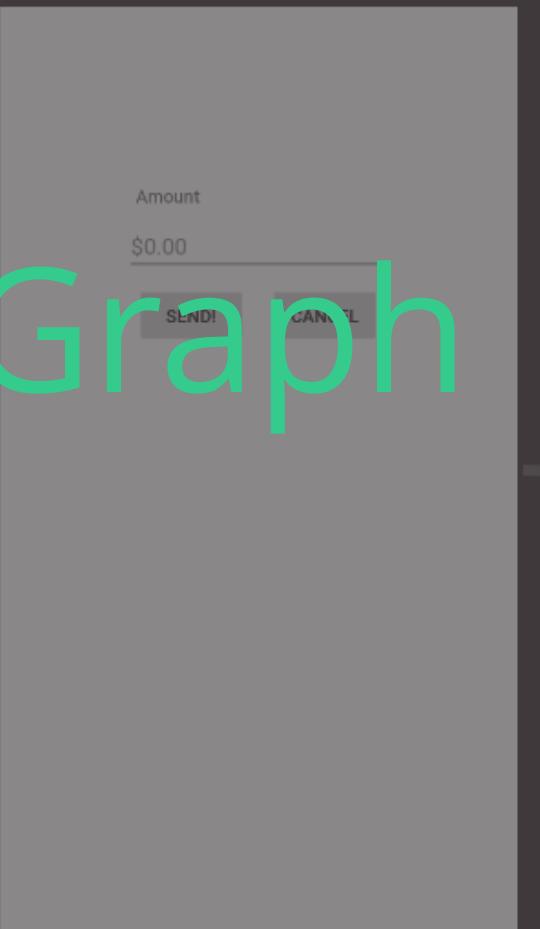
fragmentMain



fragmentChooseRecipient...



fragmentSpecifyAmount



fragmentConfirmation



fragmentViewBalance



@askashdavies

2

nav_graph.xml

Destinations

HOST

- activity_main (fragment)

GRAPH

- fragmentMain - Start
- fragmentViewBalance
- navigation

navigation

fragmentMain

fragmentViewBa...

VIEW TRANSACTIONS

SEND MONEY

VIEW BALANCE

\$1

Type Root Graph

Label

ID nav_graph.xml

Start Destination fragmentMain

Argument Default Values

No arguments on the destination

Global Actions

Click + to add Actions

Deep Links

Click + to add Deep Links

Design Text

```
graph LR; fragmentMain -- "VIEW TRANSACTIONS" --> fragmentViewBalance; fragmentMain -- "SEND MONEY" --> fragmentViewBalance; fragmentMain -- "VIEW BALANCE" --> fragmentViewBalance;
```

1

2

3

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>
```

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)  
)
```



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: 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

 Migrating

- Move screen behaviour away from activities
- Create new activity for NavHostFragment
- Move existing activity logic to fragment
- Initialise fragment in host activity
- Pass arguments as necessary
- Create navigation graph
- Profit



Scoping

Application

Activity 1

Activity 2

Activity 3

Application

Activity 1

Destination 1

Destination 2

Activity 2

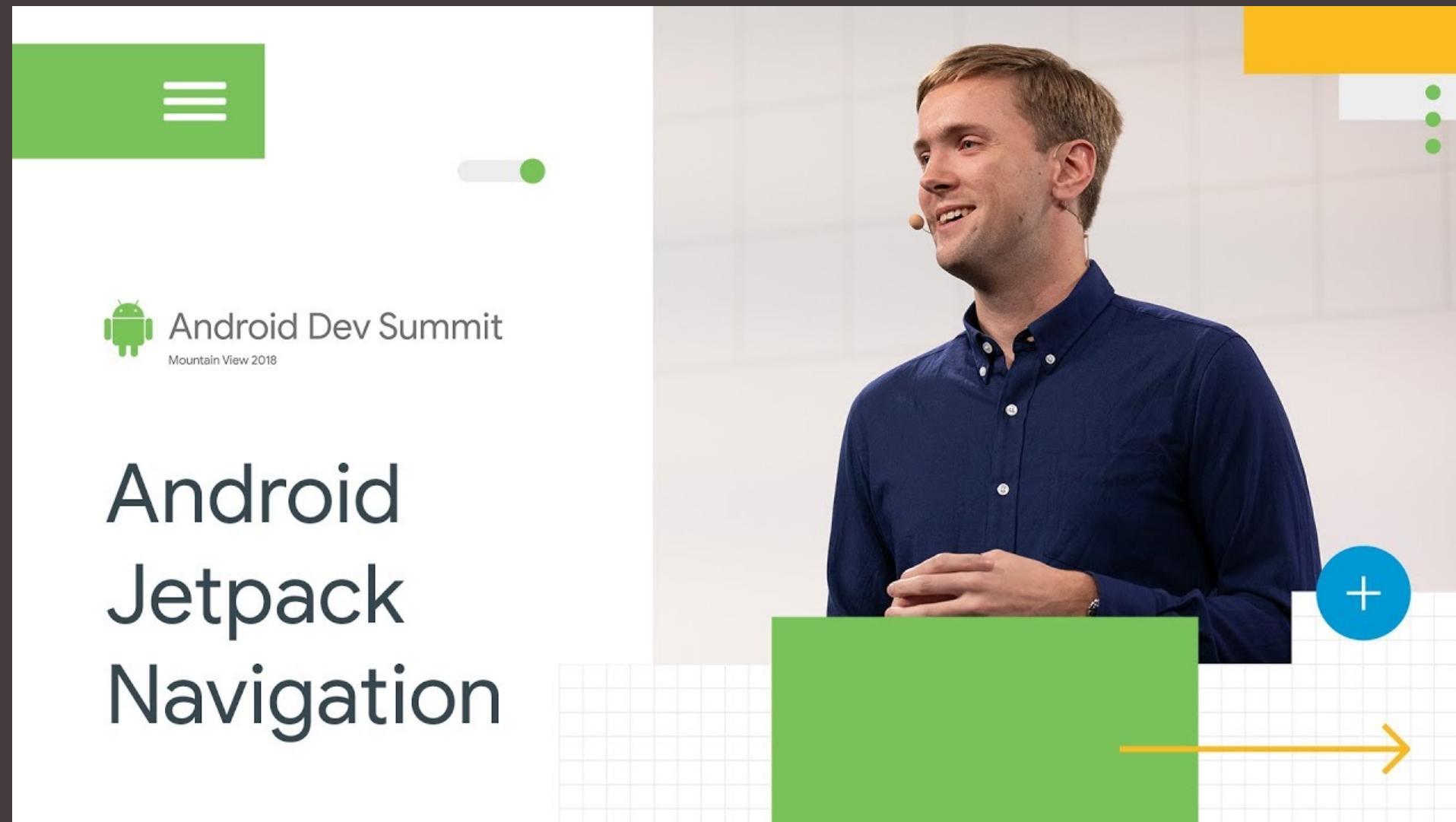
Destination 1

Destination 2

Destination 3

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

bit.ly/2Jo94x9



New Features: 2.1.0-alpha2

You can now create ViewModels that are scoped at a navigation graph level via the `by navGraphViewModels()` property delegate for Kotlin users or by using the `getViewModelStore()` API added to NavController. b/111614463

Thanks!

@askashdavies

