ITSMAP F13 Lesson 4
Part 2

Androids User Interface:

Views + Fragments
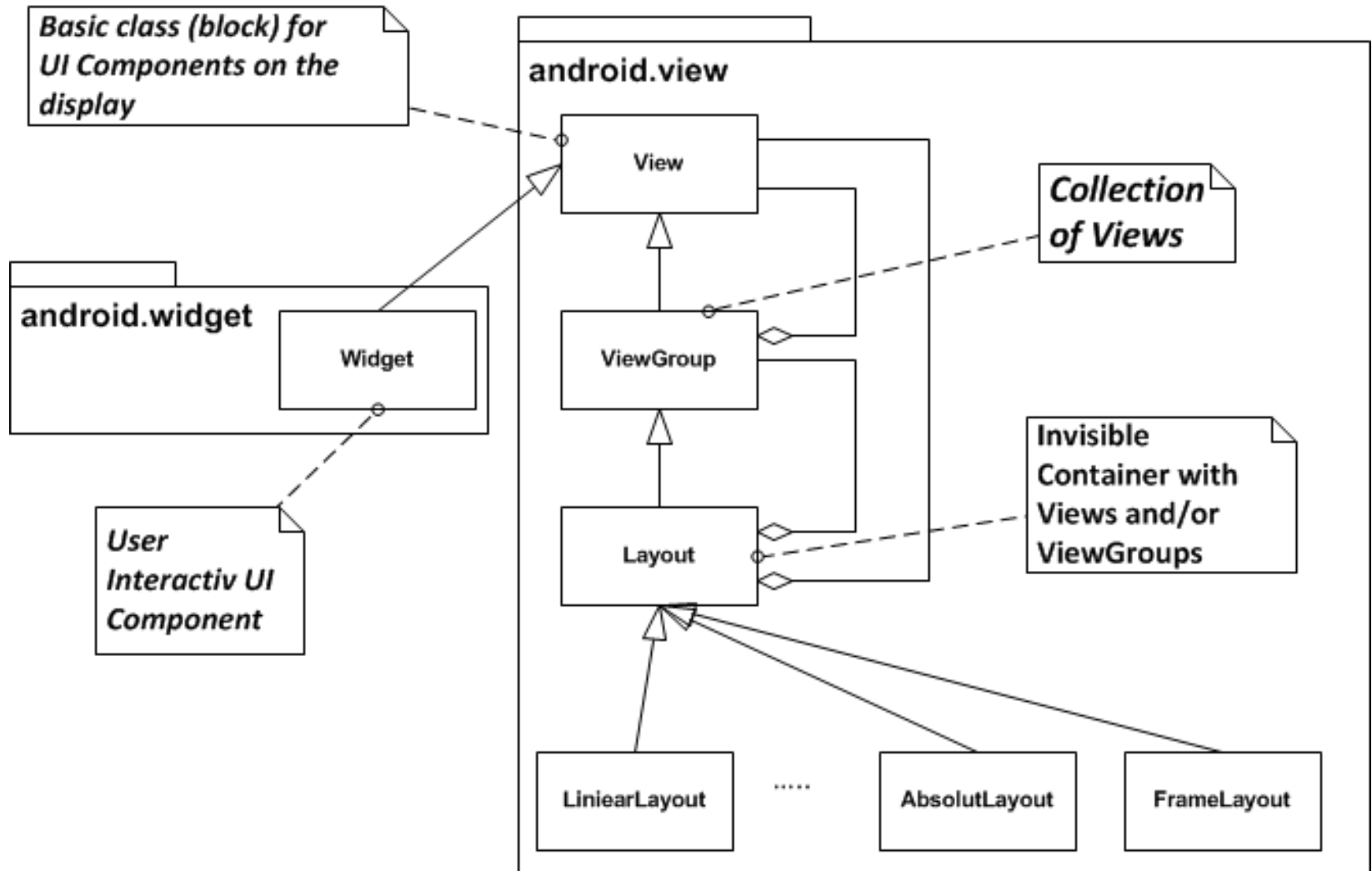
Jesper Rosholm Tørresø

Subjects
Part 2

1. View with Fragments

2. Event handling in Fragments

3. Adapters

4. Exercise 3

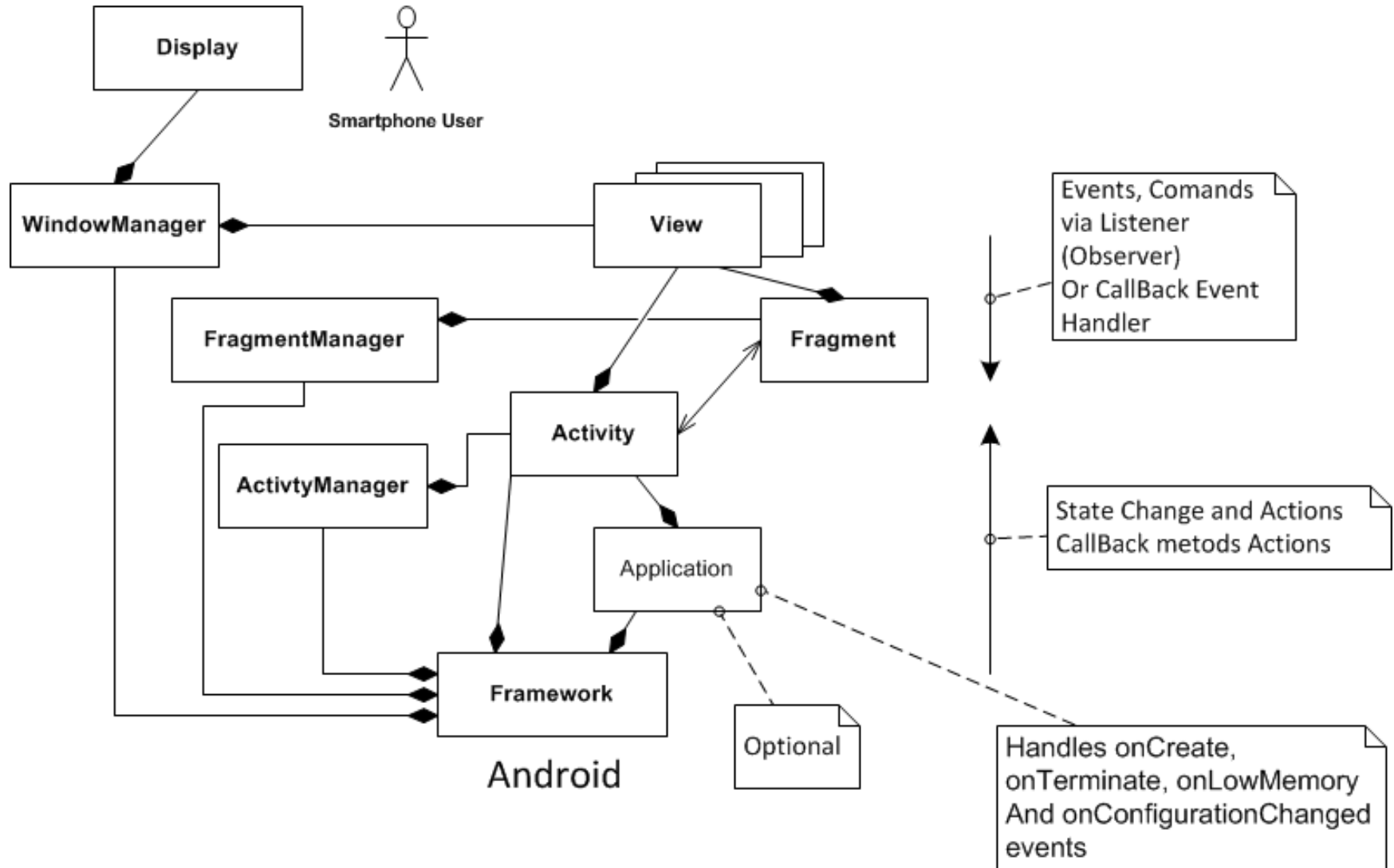# Android Java View Class Hierarchy
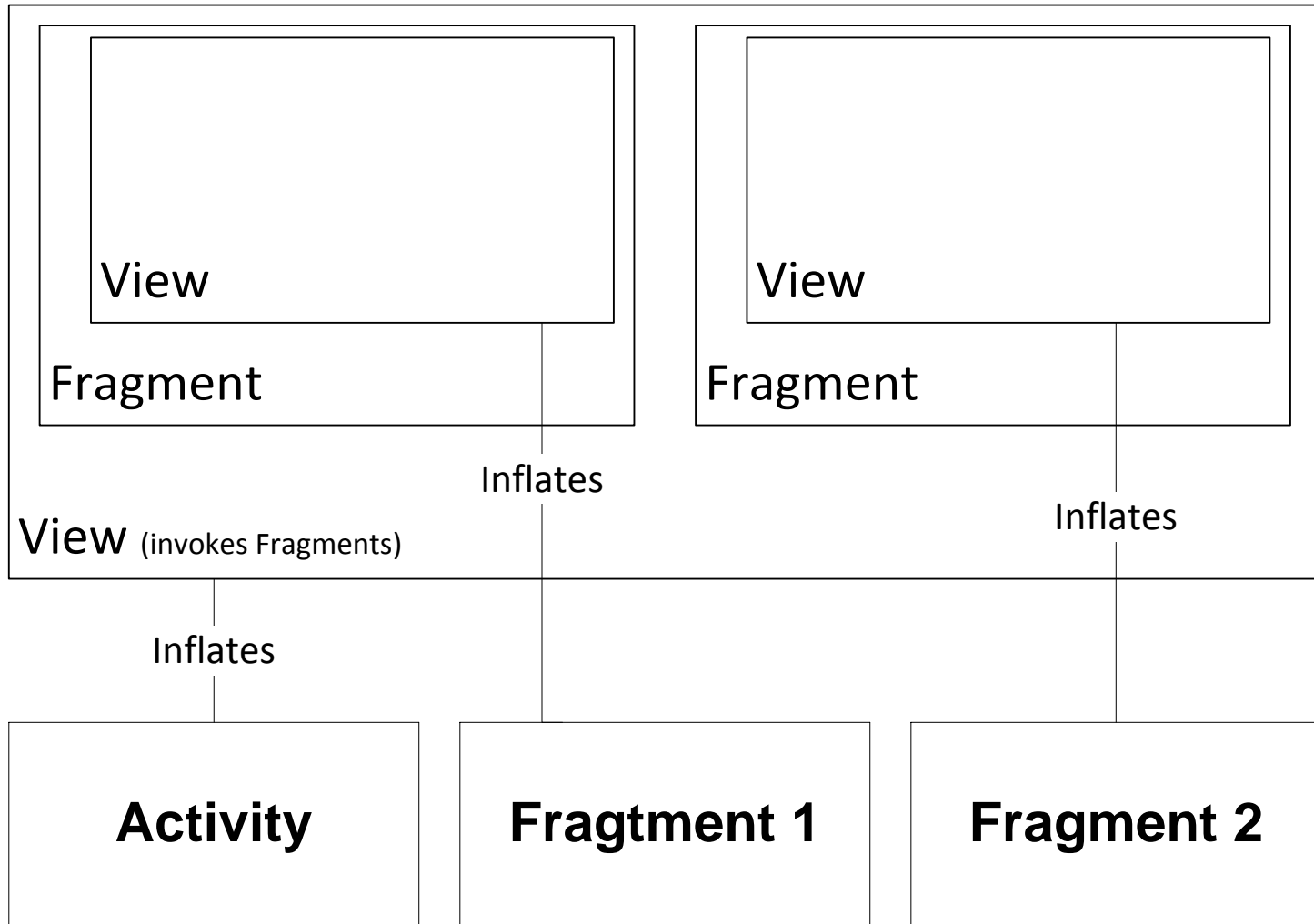
Fragment
From Android 3.0 ->

- Enable you to divide your Activities into fully encapsulated reusable components, each with its own lifecycle and UI.
- Allows you to adapt Views to different screen sizes and orientations etc.
- Tightly bound to the Activity into which is placed. Do not need to registered in the Android Manifest
- Can be reused across your Activities
- Activity can add, exchange and moved Fragments
- **Available for platforms API level 4 to 10 using the support library**

App Overall Architecture:
User Interface by View, with Fragments (part 2)

Display

Smartphone User

WindowManager

View

FragmentManager

Fragment

Activity

ActivtyManager

Application

Framework

Optional

Android

Events, Comands via Listener (Observer) Or CallBack Event Handler

State Change and Actions CallBack metods Actions

Handles onCreate, onTerminate, onLowMemory And onConfigurationChanged events

View and Fragment

Fragment Lifecycle
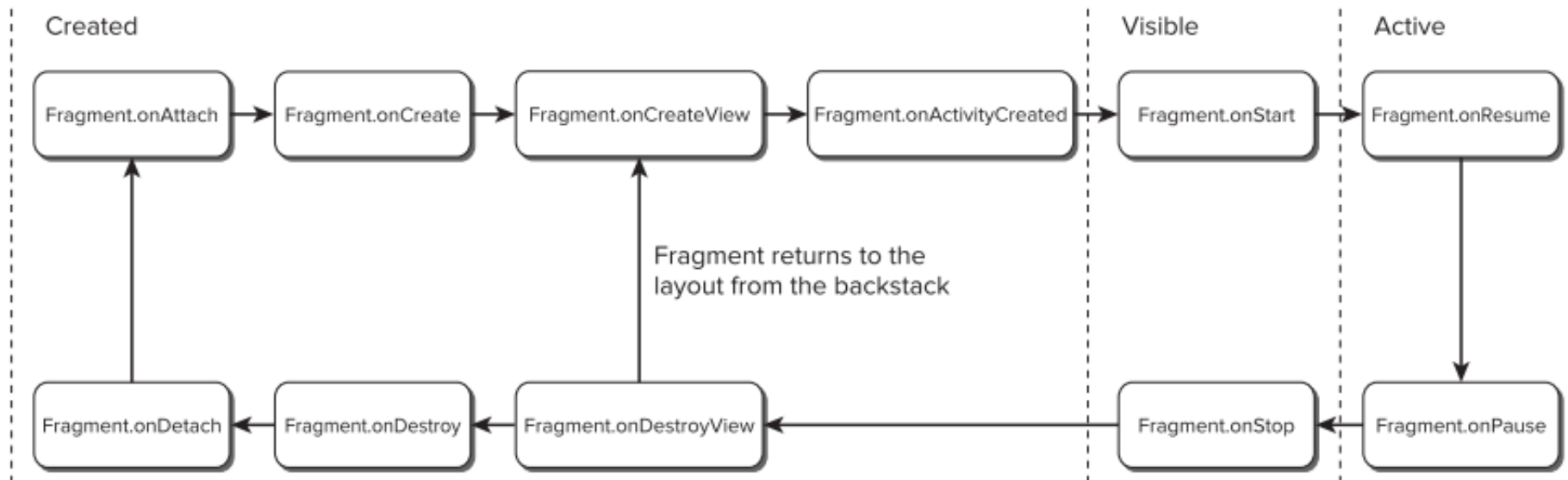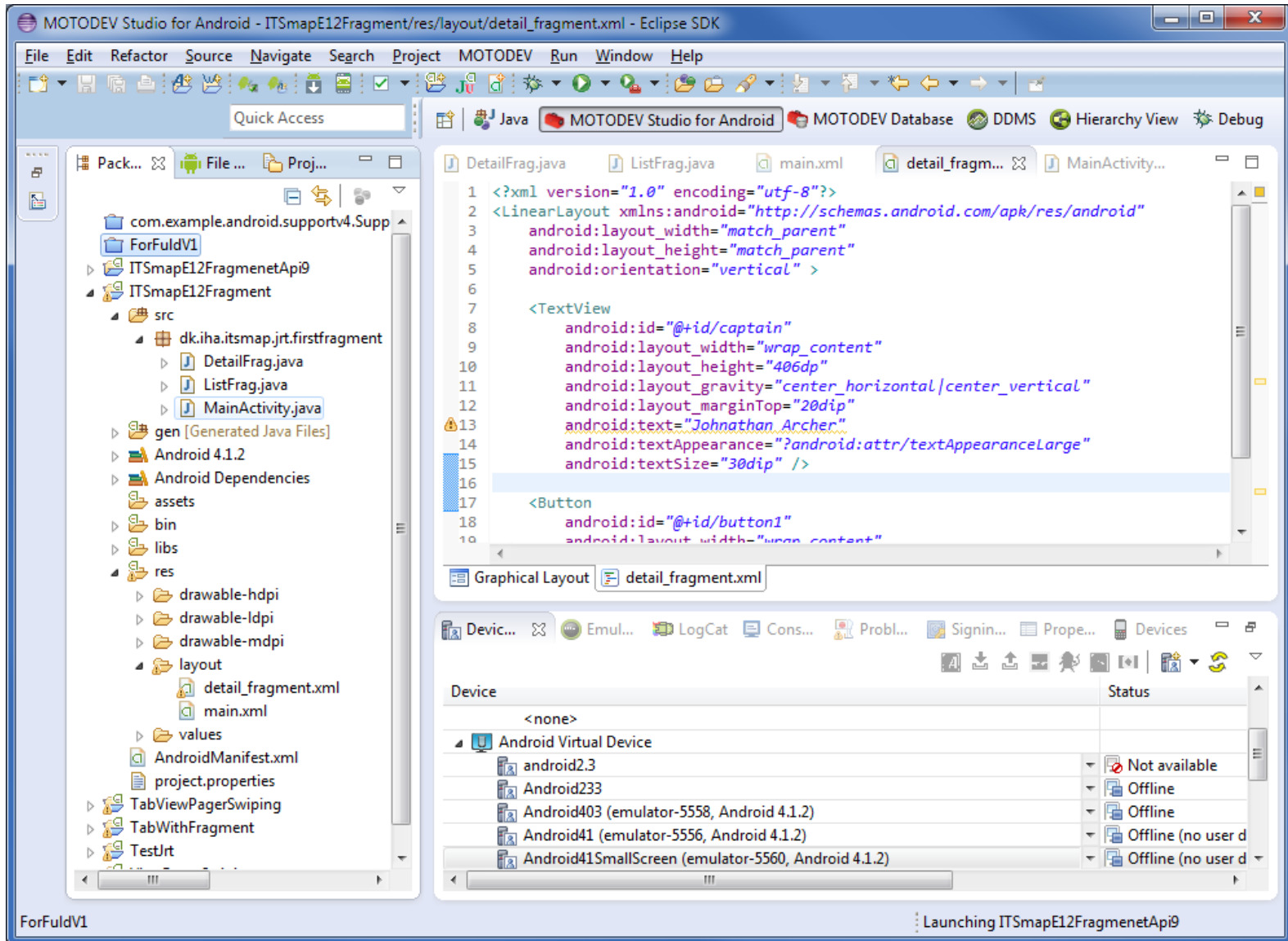# Controlled by FragmentManager

FIGURE 4-9

Get a Fragment Skeleton from [Meier] code snippets
PA4AD_Ch04_Fragments/src/MySkeletonFragment.java

Fragment Example

Parent Activity and View

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="horizontal" >

  <fragment
    android:id="@+id/frag_series"
    android:layout_width="200dip"
    android:layout_height="match_parent"
    android:layout_marginTop="?android:attr/actionBarSize"
    class="dk.iha.itsmap.jrt.firstfragment.ListFrag" />

    <fragment
        android:id="@+id/frag_capt"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        class="dk.iha.itsmap.jrt.firstfragment.DetailFrag"
        />
</LinearLayout>
```

```java
package dk.iha.itsmap.jrt.firstfragment;

import android.app.Activity;
import android.os.Bundle;

public class MainActivity extends Activity {
    /** Called when the activity is
        first created. */
    @Override
    public void onCreate(Bundle
                            savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```

Fragment 1 View

```xml
<?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" >

    <TextView
        android:id="@+id/captain"
        android:layout_width="wrap_content"
        android:layout_height="406dp"
        android:layout_gravity="center_horizontal|center_vertical"
        android:layout_marginTop="20dip"
        android:text="Johnathan Archer"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:textSize="30dip" />

    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Button" />

</LinearLayout>
```

Fragment 1 code

```java
public class DetailFrag extends Fragment {

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,Bundle savedInstanceState)
{
View view = inflater
.inflate(R.layout.detail_fragment, container, false);
Button mButton = (Button) view.findViewById(R.id.button1);
    mButton.setOnClickListener(new OnClickListener() {
//          @Override
        public void onClick(View v) {
            // here you set what you want to do when user clicks your button,
            // e.g. launch a new activity
        Log.i("JrtFragmentTest","DetailViewButton");

        }
    });

return view;
}

public void setText(String item) {
TextView view = (TextView) getView().findViewById(R.id.captain);
view.setText(item);
}

}
```

Fragment 2 code part 1
Uses a ListFragment (Gives the View)

```java
package dk.iha.itsmap.jrt.firstfragment;

import android.app.ListFragment;
import android.os.Bundle;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.ListView;

public class ListFrag extends ListFragment {

        @Override
        public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        }
        @Override
        public void onActivityCreated(Bundle savedInstanceState) {
        super.onActivityCreated(savedInstanceState);
        String[] values = new String[] { "Enterprise", "Star Trek", "Next Generation", "Deep Space 9", "Voyager"};
        ArrayAdapter<String> adapter = new ArrayAdapter<String>(getActivity(), android.R.layout.simple_list_item_1,
values);

        setListAdapter(adapter);
        }
```

Fragment 2 code part 2
Use a ListFragment (Gives View)

```java
public void onListItemClick(ListView l, View v, int position, long id) {
String item = (String) getListAdapter().getItem(position);
DetailFrag frag = (DetailFrag) getFragmentManager().findFragmentById(
R.id.frag_capt);
if (frag != null && frag.isInLayout()) {
frag.setText(getCapt(item));
}
}

private String getCapt(String ship) {
if (ship.toLowerCase().contains("enterprise")) {
return "Johnathan Archer";
}
if (ship.toLowerCase().contains("star trek")) {
return "James T. Kirk";
}
if (ship.toLowerCase().contains("next generation")) {
return "Jean-Luc Picard";
}
if (ship.toLowerCase().contains("deep space 9")) {
return "Benjamin Sisko";
}
if (ship.toLowerCase().contains("voyager")) {
return "Kathryn Janeway";
}
return "???";
}
```

```xml
<fragment
    android:id="@+id/frag_capt"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    class="dk.iha.itsmap.jrt.firstfragm
    />
```

```java
public class DetailFrag extends Fragment {

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,Bundle savedInstanceState)
{
View view = inflater
.inflate(R.layout.detail_fragment, container, false);
Button mButton = (Button) view.findViewById(R.id.button1);
    mButton.setOnClickListener(new OnClickListener() {
//        @Override
        public void onClick(View v) {
            // here you set what you want to do when user clicks your button,
            // e.g. launch a new activity
        Log.i("JrtFragmentTest","DetailViewButton");
        }
    });

return view;
}

public void setText(String item) {
TextView view = (TextView) getView().findViewById(R.id.captain);
view.setText(item);
}

}
```

Handling/Changing Fragments must be done in Transactions

```
FragmentTransaction fragmentTransaction = fragmentManager.beginTransaction();
fragmentTransaction.add(R.id.ui_container, new MyListFragment());
fragmentTransaction.commit();
```

```
FragmentTransaction fragmentTransaction = fragmentManager.beginTransaction();
Fragment fragment = fragmentManager.findFragmentById(R.id.details_fragment);
fragmentTransaction.remove(fragment);
fragmentTransaction.commit();
```

```
FragmentTransaction fragmentTransaction = fragmentManager.beginTransaction();
fragmentTransaction.replace(R.id.details_fragment,
                            new DetailFragment(selected_index));
fragmentTransaction.commit();
```

Lesson 4 Exercise 3

- [Meier] ch 4 tutorial 2 "Using Fragments for Your To-Do List" (ch 4 excerpt on Campusnet.)