

Activity

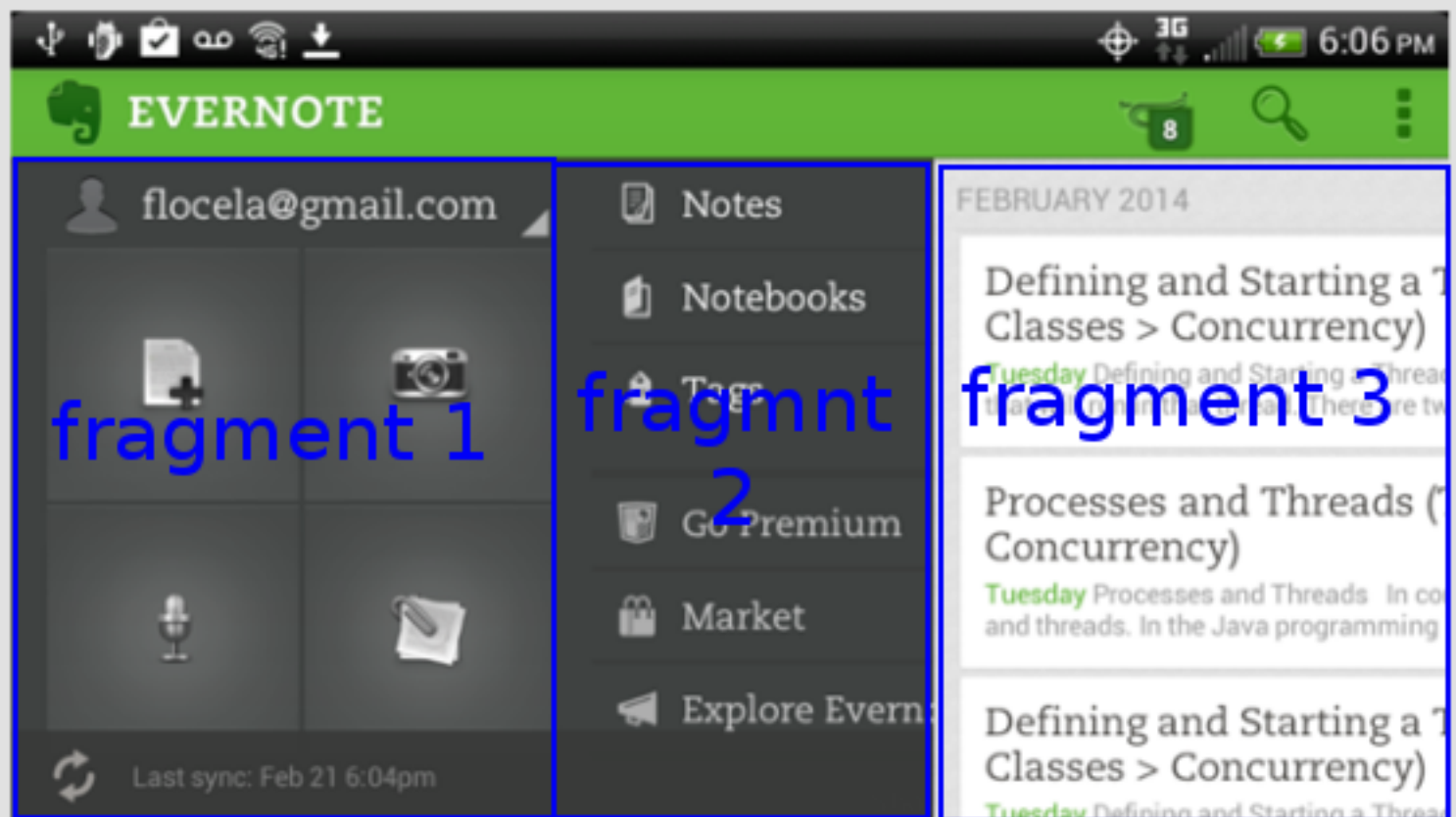


Fragment

Thread

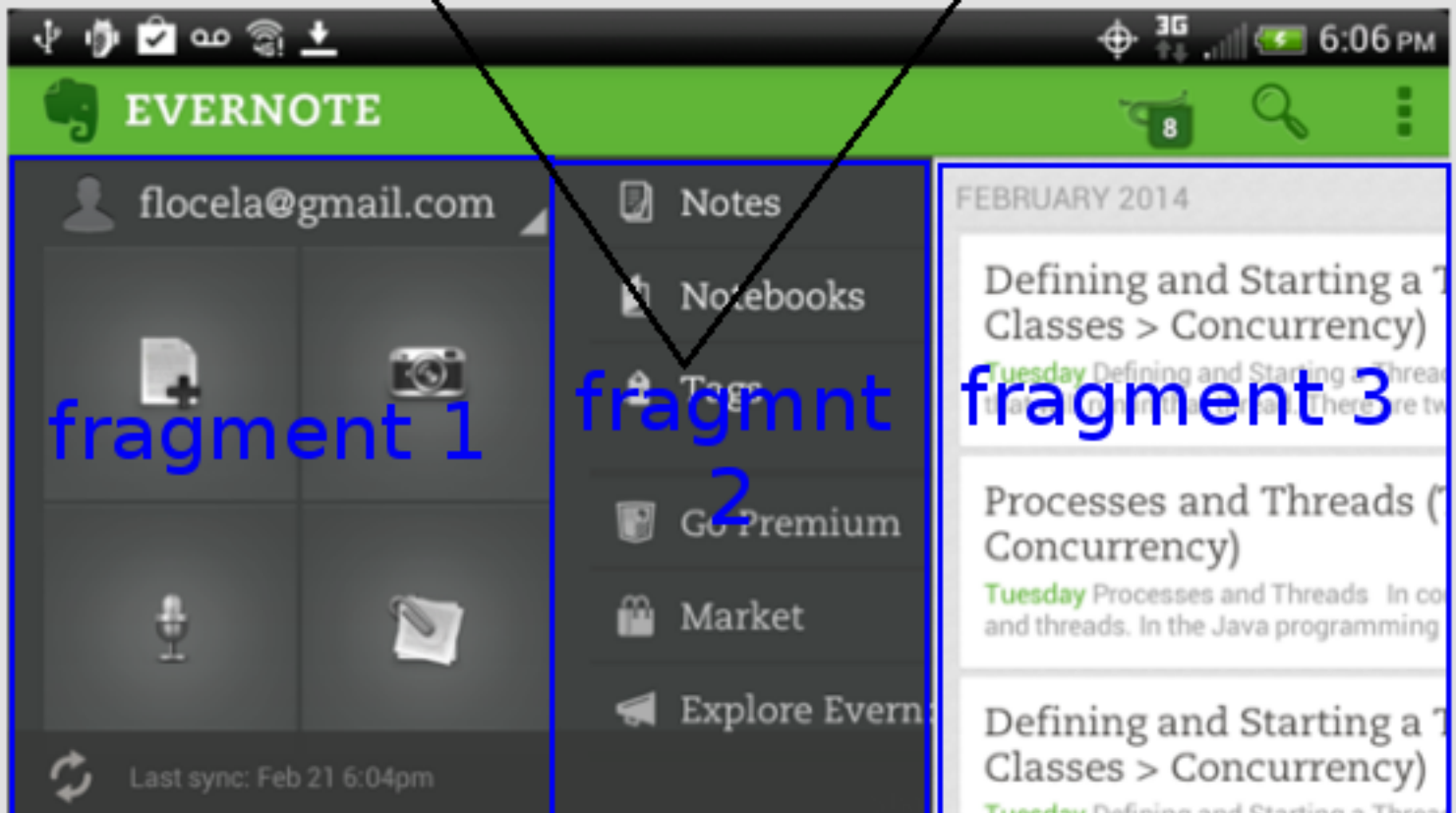
by Flo

An Activity with Viewable Fragments

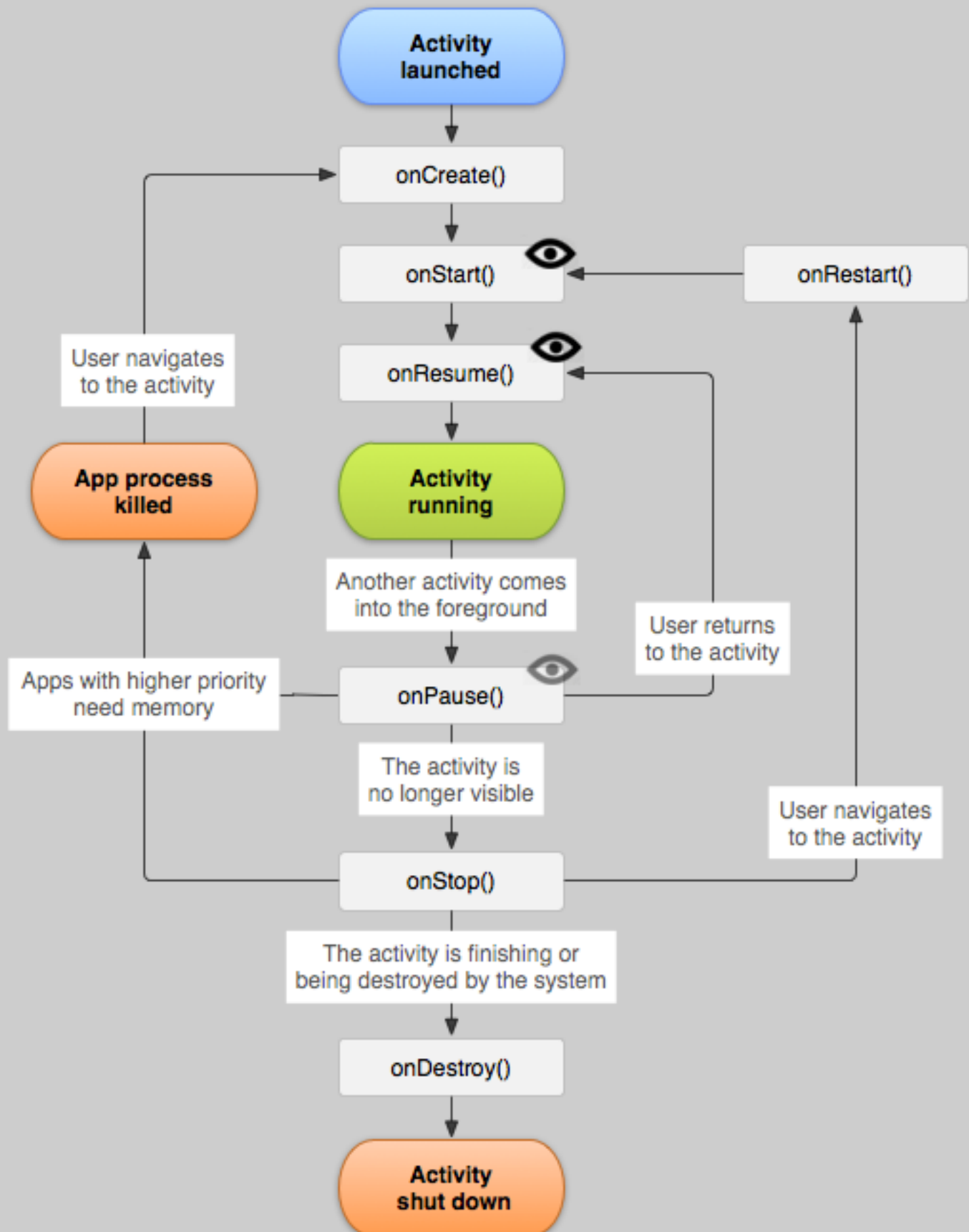


Some Fragments Work Behind the Scenes.

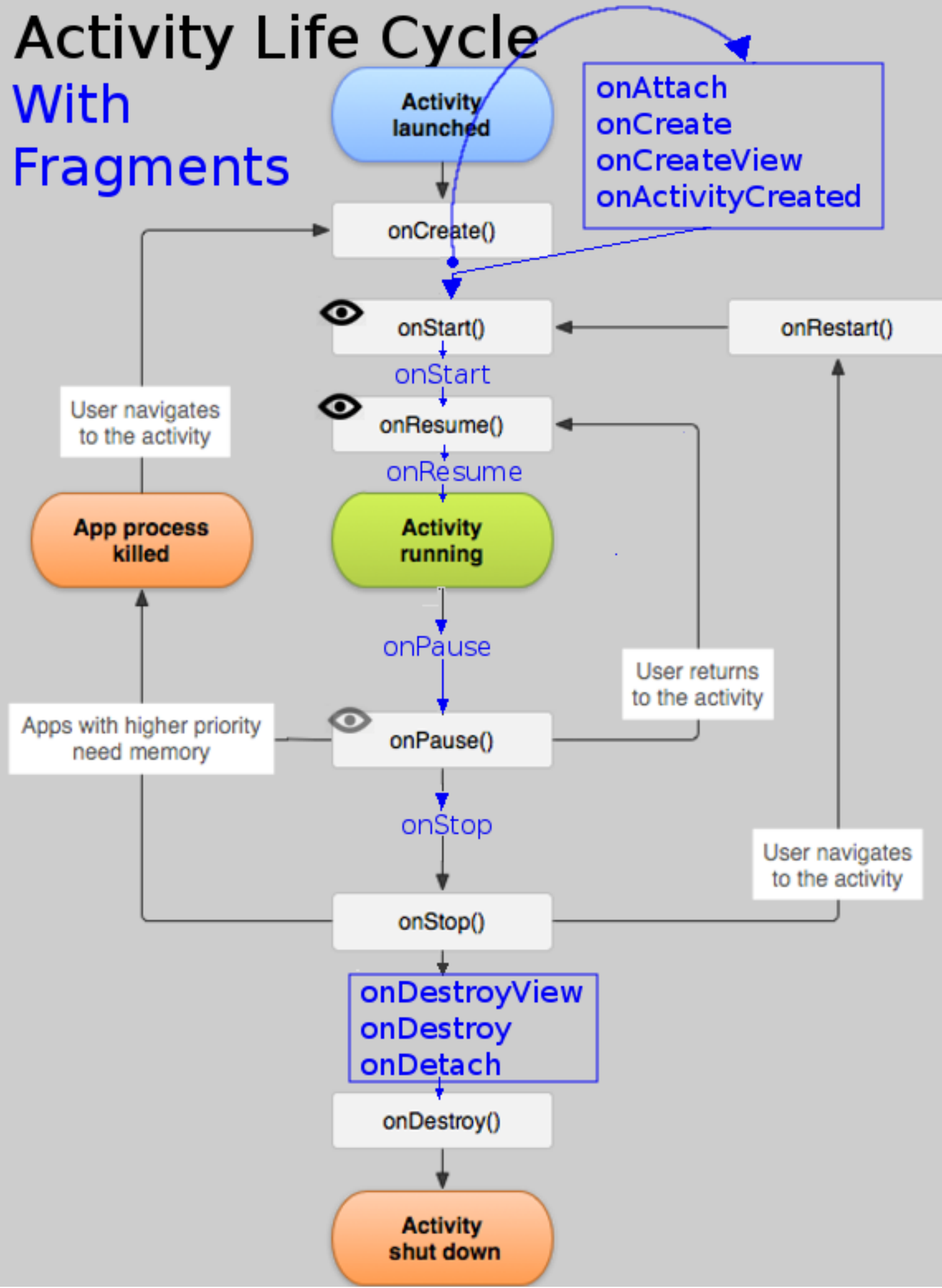
A fragment that starts a worker thread. (This particular fragment doesn't have a view.)



Activity Life Cycle



Activity Life Cycle With Fragments



Activity Life Cycle and Fragment Life Cycle

- ▶ 1. Activity.onCreate
2. Fragment.onAttach
3. Fragment.onCreate -add a thread
4. Fragment.onCreateView
5. Fragment.onActivityCreated -notify thread
6. Activity.onStart
7. Fragment.onStart
8. Activity.onResume
9. Fragment.onResume
- ROTATE PHONE**
10. Fragment.onPause
11. Activity.onPause
12. Fragment.onStop
13. Activity.onStop
14. Fragment.onDestroyView
15. Fragment.onDestroy
16. Fragment.onDetach
17. Activity.onDestroy

Life Cycle-

Non-Retained
Fragments _{vs}

Retained
Fragment

A1

A1

A1 — F1

A1 — F1

A1 — F1 — Th2

A1 — F1 — Th2

Rotate Phone

Rotate Phone

~~A1~~ — ~~F1~~ — Th2

~~A1~~ F1 — Th2

A2 — F2 — Th3

A2 — F1 — Th2

~~A1~~ — ~~F1~~  Th2

A2 — F1 — ~~Th2~~

~~A1~~ — ~~F1~~  ~~Th2~~

A2 — ~~F1~~ — ~~Th2~~

~~A2~~ — F2 — Th3

~~A2~~ — ~~F1~~ — ~~Th2~~

~~A2~~ — ~~F2~~ — Th3

~~A2~~ — ~~F2~~  Th3

~~A2~~ — ~~F2~~  ~~Th3~~

Activity Life Cycle and Fragment Life Cycle

1. Activity.onCreate
 2. Fragment.onAttach
 3. Fragment.onCreate -add a thread
 4. Fragment.onCreateView
 5. Fragment.onActivityCreated -notify thread
 6. Activity.onStart
 7. Fragment.onStart
 8. Activity.onResume
 9. Fragment.onResume
- ROTATE PHONE**
10. Fragment.onPause
 11. Activity.onPause
 12. Fragment.onStop
 13. Activity.onStop
 14. Fragment.onDestroyView
 15. Fragment.onDestroy
 16. Fragment.onDetach
 17. Activity.onDestroy

Code for Set Retain Instance

MainActivity.java

RetainedFragment.java

```
50
51 @Override
52 public void onAttach (Activity activity) {
53     super.onAttach(activity);
54     Log.i("RetainedFragment", "-"+Thread.currentThread().getId()
55         + " " + " " + RetainedFragment.this.toString()+ "onAttach");
56 }
57
58
59 @Override
60 public void onCreate(Bundle savedInstanceState) {
61     super.onCreate(savedInstanceState);
62     Log.i("RetainedFragment", "-"+Thread.currentThread().getId()
63         + " " + " " + RetainedFragment.this.toString()+ "onCreate");
64     setRetainInstance(true);
65     mThread.start();
66 }
67
68 @Override
69 public void onActivityCreated(Bundle savedInstanceState) {
70     Log.i("RetainedFragment", "-"+Thread.currentThread().getId()
71         + " " + RetainedFragment.this.toString()+ "onActivityCreated");
72     super.onActivityCreated(savedInstanceState);
73     synchronized (mThread) {
74         mReady = true;
75         mThread.notify();
76     }
77 }
78
79 @Override
80 public void onDetach() {
81     Log.i("RetainedFragment", "-"+Thread.currentThread().getId()
82         + " " + RetainedFragment.this.toString()+ "onDetach");
83     synchronized (mThread) {
84         mReady = false;
85         mThread.notify();
86     }
87     super.onDetach();
88 }
89
90 @Override
91 public void onStart () {
```

Code for Set Retain Instance

```
10 public class MainActivity extends Activity implements WorkListener {
11     RetainedFragment mWorkFragment;
12     @Override
13     protected void onCreate(Bundle savedInstanceState) {
14         Log.i("MainActivity", "-" + Thread.currentThread().getId()
15             + " " + MainActivity.this.toString() + "onCreate");
16         super.onCreate(savedInstanceState);
17         setContentView(R.layout.activity_main);
18         Button button = (Button) findViewById(R.id.closebutton);
19         button.setOnClickListener(new View.OnClickListener() {
20             if (savedInstanceState == null) {
21                 FragmentManager fm = getFragmentManager();
22                 mWorkFragment = (RetainedFragment)fm.findFragmentByTag("work");
23                 if (mWorkFragment == null) {
24                     mWorkFragment = new RetainedFragment();
25                     fm.beginTransaction().add(mWorkFragment, "work").commit();
26                 }
27             }
28         });
29     }
30     @Override
31     public void notifyAboutWork(String status) {
32         Log.i("MainActivity", "-" + Thread.currentThread().getId()
33             + " " + MainActivity.this.toString() + status);
34     }
35 }
```

Code for Set Retain Instance

MainActivity.java

RetainedFragment.java

```
8 public class RetainedFragment extends Fragment {
9     boolean mReady = false;
10    final Thread mThread = new Thread() {
11        @Override
12        public void run() {
13            synchronized (this) {
14                while (!mReady) {
15                    try {
16                        wait();
17                    } catch (InterruptedException e) {
18                    }
19                }
20            } // end synchronized
21            for (int ii = 0; ii < 101; ii++) {
22                try {
23                    this.sleep(250l);
24                } catch (InterruptedException e) {
25                    e.printStackTrace();
26                }
27                Log.i("RetainedFragment", "-" + Thread.currentThread().
28                    getId() + " " + RetainedFragment.this.toString()
29                    + " working: " + ii);
30                synchronized(this) {
31                    if(mReady){
32                        WorkListener activity = (WorkListener)
33                            RetainedFragment.this.getActivity();
34                        if (ii==1) { activity.notifyAboutWork("at 1%"); }
35                        else if (ii==50) { activity.notifyAboutWork("at 50%"); }
36                        else if (ii==75) { activity.notifyAboutWork("at 75%"); }
37                        else if (ii==100){ activity.notifyAboutWork("at 100%"); }
38                    }
39                    else {
40                        try {
41                            wait();
42                        } catch (InterruptedException e) {
43                            e.printStackTrace();
44                        }
45                    }
46                }
47            }
48        }
49    };
};
```

Fragment Retained Results

18.496: MainActivity-th01: onCreate
18.616: RetainedFragment-th01: onAttach
18.616: RetainedFragment-th01: onCreate
18.616: RetainedFragment-th01: onActivityCreated
18.616: MainActivity-th01: onStart
18.616: RetainedFragment-th01: onStart
18.616: MainActivity-th01: onResume
18.616: RetainedFragment-th01: onResume
18.866: RetainedFragment-th02: working: 0
19.116: RetainedFragment-th02: working: 1
19.116: MainActivity-th02: at 1%
19.367: RetainedFragment-th02: working: 2-50
31.398: MainActivity-th02: at 50%
31.649: RetainedFragment-th02: working: 51-59
ROTATE PHONE
33.911: RetainedFragment-th01: onPause
33.911: MainActivity-th01: onPause
33.911: RetainedFragment-th01: onStop
33.911: MainActivity-th01: onStop
33.911: RetainedFragment-th01: onDetach
33.911: MainActivity-th01: onDestroy
33.911: RetainedFragment-th02: working: 60
33.921: MainActivity-th01: onCreate
33.921: RetainedFragment-th01: onAttach
33.981: RetainedFragment-th01: onActivityCreated
33.981: MainActivity-th01: onStart

Fragment Retained Results

33.981: RetainedFragment-th01: onStart
33.991: MainActivity-th01: onResume
33.991: RetainedFragment-th01: onResume
34.221: RetainedFragment-th02: working: 61-75
37.745: MainActivity-th02: at 75%-19860
37.995: RetainedFragment-th02: working: 76-100
44.021: MainActivity-th02: at 100%

Results -

Fragment is Not Retained

```
36:35.114: MainActivity-th01: onCreate
36:35.224: Fragment-th01: onAttach
36:35.224: Fragment-th01: onCreate
36:35.224: Fragment-th01: onActivityCreated
36:35.224: MainActivity-th01: onStart
36:35.234: Fragment-th01: onStart
36:35.234: MainActivity-th01: onResume
36:35.234: Fragment-th01: onResume
36:35.484: Fragment-th02:  working: 0
36:35.735: Fragment-th02:  working: 1
36:35.735: MainActivity-th02: at 1%
36:35.985: Fragment-th02:  working: 2-50
36:48.017: MainActivity-th02: at 50%
36:48.267: Fragment-th02:  working: 51-59
      ROTATE PHONE
36:50.319: Fragment-th01: onPause
36:50.319: MainActivity-th01: onPause
36:50.339: Fragment-th01: onStop
36:50.339: MainActivity-th01: onStop
36:50.339: Fragment-th01: onDestroy
36:50.339: Fragment-th01: onDetach
36:50.339: MainActivity-th01: onDestroy
36:50.379: D/memalloc/dev/pmem: Unmapping buf
size:16433152 offset:14344192
36:50.409: MainActivity-th01: onCreate
36:50.409: Fragment-th01: onAttach
36:50.409: Fragment-th01: onCreate
36:50.429: Fragment-th01: onActivityCreated
```

Results -

Fragment is Not Retained

```
36:50.429: MainActivity-th01: onStart
36:50.429: Fragment-th01: onStart
36:50.429: MainActivity-th01: onResume
36:50.429: Fragment-th01: onResume
36:50.529: Fragment-th02: working: 60
36:50.679: Fragment-th03: working: 0
36:50.779: Fragment-th02: working: 61
36:50.929: Fragment-th03: working: 1
36:50.929: MainActivity-th03: at 1%
36:51.029: Fragment-th02: working: 62

36:51.180: Fragment-th03: working: 2
36:51.280: Fragment-th02: working: 63
36:51.430: Fragment-th03: working: 3
36:51.530: Fragment-th02: working: 64
36:51.690: Fragment-th03: working: 4
36:51.780: Fragment-th02: working: 65-74
36:51.940: Fragment-th03: working: 5-14
36:54.293: Fragment-th02: working: 75
36:54.293: W/dalvikvmthreadid=11: thread ex
exception (group=0x40abc228)
36:54.293: E/AndroidRuntimeFATAL EXCEPTION:
36:54.293: E/AndroidRuntimejava.lang.NullPo
36:54.293: E/AndroidRuntime  at
com.example.fragmentnotretained.NotRetained
nedFragment.java:36)
36:54.333: Fragment-th01: onPause
```

Results -

Fragment is Not Retained

```
36:54.333: MainActivity-th01: onPause
36:54.433: Fragment-th03: working: 15
36:54.613: D/memalloc/dev/pmem: Unmapping buff
size:16433152 offset:14344192
36:54.613: D/memalloc/dev/pmem: Unmapping buff
size:22208512 offset:20119552
36:54.683: Fragment-th03: working: 16
36:54.813: Fragment-th01: onStop
36:54.813: MainActivity-th01: onStop
36:54.813: Fragment-th01: onDestroy
36:54.813: Fragment-th01: onDetach
36:54.813: MainActivity-th01: onDestroy
36:54.943: Fragment-th03: working: 17-50
37:03.331: W/dalvikvmthreadid=12: thread exiti
exception (group=0x40abc228)
37:03.331: E/AndroidRuntime_2_crashcrash in th
Thread-th03:
37:03.331: E/AndroidRuntime_2_crashjava.lang.N
37:03.331: E/AndroidRuntime_2_crash at
com.example.fragmentnotretained.NotRetainedFra
nedFragment.java:34)
```


Code for Fragment Not Retained

```
43 @Override
44 public void onAttach (Activity activity) {
45     Log.i("Fragment", "-"+Thread.currentThread().getId()+ " " +
46         NotRetainedFragment.this.toString() +"onAttach");
47     super.onAttach(activity);
48 }
49
50
51 @Override
52 public void onCreate(Bundle savedInstanceState) {
53     super.onCreate(savedInstanceState);
54     Log.i("Fragment", "-"+Thread.currentThread().getId()+ " " +
55         NotRetainedFragment.this.toString() +"onCreate");
56     mThread.start();
57 }
58
59 @Override
60 public void onActivityCreated(Bundle savedInstanceState) {
61     Log.i("Fragment", "-"+Thread.currentThread().getId()+ " " +
62         NotRetainedFragment.this.toString() +"onActivityCreated");
63     super.onActivityCreated(savedInstanceState);
64     synchronized (mThread) {
65         mReady = true;
66         mThread.notify();
67     }
68 }
69
70 @Override
71 public void onDestroy() {
72     Log.i("Fragment", "-"+Thread.currentThread().getId()+ " " +
73         NotRetainedFragment.this.toString() +"onDestroy");
74     super.onDestroy();
75 }
76
77 @Override
78 public void onDetach() {
79     Log.i("Fragment", "-"+Thread.currentThread().getId()+ " " +
80         NotRetainedFragment.this.toString() +"onDetach");
81     super.onDetach();
82 }
```

Code for Fragment Not Retained

NotRetainedFragment.java

*MainActivity.java

```
1 package com.example.fragmentnotretained;
2
3 import android.os.Bundle;
4
5
6
7
8
9
10 public class MainActivity extends Activity implements WorkListener{
11     NotRetainedFragment mWorkFragment;
12     @Override
13     protected void onCreate(Bundle savedInstanceState) {
14         Log.i("MainActivity", "-"+Thread.currentThread().getId()+ " " +
15             MainActivity.this.toString()+ "onCreate");
16         super.onCreate(savedInstanceState);
17         setContentView(R.layout.activity_main);
18         Button button = (Button) findViewById(R.id.closebutton);
19         button.setOnClickListener(new View.OnClickListener() {
20             if (savedInstanceState == null) {
21                 FragmentManager fm = getFragmentManager();
22                 mWorkFragment = (NotRetainedFragment)fm.findFragmentByTag("work");
23                 if (mWorkFragment == null) {
24                     mWorkFragment = new NotRetainedFragment();
25                     fm.beginTransaction().add(mWorkFragment, "work").commit();
26                 }
27             }
28         });
29     }
30
31     @Override
32     protected void onStart () {
33         super.onStart();
34         Log.i("MainActivity", "-"+Thread.currentThread().getId()+ " " +
35             MainActivity.this.toString()+ "onStart");
36     }
37 }
```

Code for Fragment Not Retained

```
7
8 public class NotRetainedFragment extends Fragment {
9     boolean mReady = false;
10    final Thread mThread = new Thread() {
11        @Override
12        public void run() {
13            synchronized (this) {
14                while (!mReady) {
15                    try {
16                        wait();
17                    } catch (InterruptedException e) {
18                    }
19                }
20            } // end synchronized
21            for (int ii = 0; ii < 101; ii++) {
22                try {
23                    this.sleep(250l);
24                } catch (InterruptedException e) {
25                    e.printStackTrace();
26                }
27                Log.i("Fragment", "-"+Thread.currentThread().getId()+ " "
28                    + NotRetainedFragment.this.toString()+ " working: " + ii);
29                WorkListener activity = (WorkListener)NotRetainedFragment.
30                    this.getActivity();
31                if (ii==1)
32                    activity.notifyAboutWork("at 1%");
33                else if (ii==50)
34                    activity.notifyAboutWork("at 50%");
35                else if (ii==75)
36                    activity.notifyAboutWork("at 75%");
37                else if (ii==100)
38                    activity.notifyAboutWork("at 100%");
39            }
40        }
41    };
42
```