Mapping Classes to Patterns in Java & Android’s Concurrency Frameworks

Douglas C. Schmidt
d.schmidt@vanderbilt.edu
www.dre.vanderbilt.edu/~schmidt

Institute for Software Integrated Systems
Vanderbilt University
Nashville, Tennessee, USA
### Learning Objectives in this Lesson

- Understand the mapping of Android concurrency classes to software patterns

<table>
<thead>
<tr>
<th>Android/Java Mechanism</th>
<th>Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>Looper</td>
<td><em>Thread-Specific Storage</em></td>
</tr>
<tr>
<td>HaMeR framework &amp; Java Executor framework</td>
<td><em>Command Processor &amp; Active Object</em></td>
</tr>
<tr>
<td>AsyncTask framework</td>
<td><em>Half-Sync/Half-Async</em></td>
</tr>
<tr>
<td>Java LinkedBlockingQueue</td>
<td><em>Monitor Object</em></td>
</tr>
<tr>
<td>Java Streams framework</td>
<td><em>Pipes &amp; Filters</em></td>
</tr>
</tbody>
</table>
Elements of Android Concurrency Frameworks
Elements of Android Concurrency Frameworks

• Android’s concurrency frameworks are built using reusable classes
Elements of Android Concurrency Frameworks

- Android’s concurrency frameworks are built using reusable classes

Earlier we covered classes used to write concurrent Android programs
Elements of Android Concurrency Frameworks

- Android’s concurrency frameworks are built using reusable classes

We now explore the implementation of Android’s concurrency frameworks
Elements of Android Concurrency Frameworks

- Android’s concurrency frameworks are built using reusable classes

See upcoming section on “Android Concurrency Frameworks: Patterns”
Elements of Android Concurrency Frameworks

- Android’s concurrency frameworks are built using reusable classes

We cover the most important classes in the Android concurrency frameworks
Elements of Android Concurrency Frameworks

- Android’s concurrency frameworks are built using reusable classes

Some classes are used by both the HaMeR & AsyncTask concurrency frameworks
Elements of Android Concurrency Frameworks

- Android’s concurrency frameworks are built using reusable classes
- Looper – Run a message loop for a thread

See developer.android.com/reference/android/os/Looper.html
Elements of Android Concurrency Frameworks

- Android’s concurrency frameworks are built using reusable classes
- Looper – Run a message loop for a thread
  - Applies *Thread-Specific Storage* pattern to ensure only one Looper is allowed per Thread

See [www.dre.vanderbilt.edu/~schmidt/PDF/TSS-pattern.pdf](http://www.dre.vanderbilt.edu/~schmidt/PDF/TSS-pattern.pdf)
Elements of Android Concurrency Frameworks

- Android’s concurrency frameworks are built using reusable classes
  - Looper
  - MessageQueue – Holds the list of messages to be dispatched by a Looper

**Elements of Android Concurrency Frameworks**

- Android’s concurrency frameworks are built using reusable classes
  - Looper
  - MessageQueue – Holds the list of messages to be dispatched by a Looper
    - Applies *Monitor Object* pattern to enqueue/dequeue Messages concurrently & efficiently

See [www.dre.vanderbilt.edu/~schmidt/PDF/monitor.pdf](http://www.dre.vanderbilt.edu/~schmidt/PDF/monitor.pdf)
Elements of Android Concurrency Frameworks

- Android’s concurrency frameworks are built using reusable classes
  - Looper
  - MessageQueue
- Message – Contains data & type information that can be sent to a Handler via a MessageQueue

See developer.android.com/reference/android/os/Message.html
Elements of Android Concurrency Frameworks

- Android’s concurrency frameworks are built using reusable classes
  - Looper
  - MessageQueue
- Message – Contains data & type information that can be sent to a Handler via a MessageQueue
  - Messages are created via Factory Method pattern

See [en.wikipedia.org/wiki/Factory_method_pattern](en.wikipedia.org/wiki/Factory_method_pattern)
Android’s concurrency frameworks are built using reusable classes:

- Looper
- MessageQueue
- Message

Handler – Allows the sending & processing of Message & Runnable objects in the MessageQueue associated with a Thread’sLooper.

See developer.android.com/reference/android/os/Handler.html
Elements of Android Concurrency Frameworks

- Android’s concurrency frameworks are built using reusable classes
  - Looper
  - MessageQueue
  - Message
- Handler – Allows the sending & processing of Message & Runnable objects in the MessageQueue associated with a Thread’s Looper
  - Handlers support Active Object & Command Processor patterns to allow sender & receiver Threads to run concurrently

See [www.dre.vanderbilt.edu/~schmidt/PDF/{Act-Obj,CommandProcessor}.pdf](http://www.dre.vanderbilt.edu/~schmidt/PDF/{Act-Obj,CommandProcessor}.pdf)
Elements of Android Concurrency Frameworks

- Android’s concurrency frameworks are built using reusable classes
  - Looper
  - MessageQueue
  - Message
  - Handler
- Runnable – Represents a command that can be executed

See developer.android.com/reference/java/lang/Runnable.html
Elements of Android Concurrency Frameworks

- Android’s concurrency frameworks are built using reusable classes
  - Looper
  - MessageQueue
  - Message
  - Handler
- Runnable – Represents a command that can be executed
  - Supports Command Processor pattern to pass commands from one Thread to another

See [www.dre.vanderbilt.edu/~schmidt/CommandProcessor.pdf](http://www.dre.vanderbilt.edu/~schmidt/CommandProcessor.pdf)
Elements of Android Concurrency Frameworks

- Android’s concurrency frameworks are built using reusable classes
  - Looper
  - MessageQueue
  - Message
  - Handler
  - Runnable

Some classes are just used by the AsyncTask concurrency framework
Elements of Android Concurrency Frameworks

- Android’s concurrency frameworks are built using reusable classes
  - Looper
  - MessageQueue
  - Message
  - Handler
  - Runnable
- FutureTask – Start & cancel an asynchronous computation, query to see if the computation is complete, & retrieve the result of the computation

See developer.android.com/reference/java/util/concurrent/FutureTask.html
Elements of Android Concurrency Frameworks

- Android’s concurrency frameworks are built using reusable classes
  - Looper
  - MessageQueue
  - Message
  - Handler
  - Runnable
- FutureTask – Start & cancel an asynchronous computation, query to see if the computation is complete, & retrieve the result of the computation
  - Supports the Future pattern to block client only if computation isn’t finished

See [en.wikipedia.org/wiki/Futures_and_promises](en.wikipedia.org/wiki/Futures_and_promises)
Elements of Android Concurrency Frameworks

- Android’s concurrency frameworks are built using reusable classes
  - Looper
  - MessageQueue
  - Message
  - Handler
  - Runnable
  - FutureTask
- Executor framework – Execute submitted Runnable tasks either sequentially or in a pool of threads

See developer.android.com/reference/java/util/concurrent/Executor.html
Elements of Android Concurrency Frameworks

- Android’s concurrency frameworks are built using reusable classes
  - Looper
  - MessageQueue
  - Message
  - Handler
  - Runnable
  - FutureTask
- Executor framework – Execute submitted Runnable tasks either sequentially or in a pool of threads
  - Implements *Proactor*, *Active Object*, & *Pooling* patterns

See [en.wikipedia.org/wiki/Proactor_pattern](en.wikipedia.org/wiki/Proactor_pattern), [en.wikipedia.org/wiki/Active_object](en.wikipedia.org/wiki/Active_object), & [kircher-schwanninger.de/michael/publications/Pooling.pdf](kircher-schwanninger.de/michael/publications/Pooling.pdf)
End of Mapping Classes to Patterns in Java & Android’s Concurrency Frameworks