Android Concurrency Frameworks: Introduction

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 Part of the Lesson

- Know the motivations for Android concurrency & concurrency frameworks
- Recognize the two types of Android concurrency frameworks
Learning Objectives in this Part of the Lesson

• Know the motivations for Android concurrency & concurrency frameworks

• Recognize the two types of Android concurrency frameworks, e.g.
  • Handler, Messages, & Runnables (HaMeR) framework

See [code.tutsplus.com/tutorials/concurrency-on-android-using-hamer-framework--cms-27129]
Learning Objectives in this Part of the Lesson

- Know the motivations for Android concurrency & concurrency frameworks
- Recognize the two types of Android concurrency frameworks, e.g.
  - Handler, Messages, & Runnables (HaMeR) framework
  - AsyncTask framework

See developer.android.com/reference/android/os/AsyncTask.html
Overview of Android Concurrency Frameworks
Overview of Android Concurrency Frameworks

- Android defines two primary concurrency frameworks
Overview of Android Concurrency Frameworks

- Android defines two primary concurrency frameworks
- Handlers, Messages, & Runnables (HaMeR)

Operations running in one or more background threads can post/send their results to the UI thread

See developer.android.com/training/multiple-threads/communicate-ui.html
Overview of Android Concurrency Frameworks

- Android defines two primary concurrency frameworks
  - Handlers, Messages, & Runnables (HaMeR)
  - AsyncTask

Operations run in one or more background threads & publish results to UI thread without manipulating threads, handlers, messages, or runnables

Overview of Android Concurrency Frameworks

- Both frameworks have pros & cons & are used extensively throughout Android

<table>
<thead>
<tr>
<th></th>
<th>Async Task</th>
<th>Posting Runnables</th>
<th>Sending Messages</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Usability (Simple)</strong></td>
<td>+++</td>
<td>+++</td>
<td>++</td>
</tr>
<tr>
<td><strong>Usability (Complex)</strong></td>
<td>+++</td>
<td>+</td>
<td>++</td>
</tr>
<tr>
<td><strong>Scalability</strong></td>
<td>+++</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td><strong>Flexibility</strong></td>
<td>++</td>
<td>+</td>
<td>+++</td>
</tr>
<tr>
<td><strong>Efficiency</strong></td>
<td>++</td>
<td>+++</td>
<td>+++</td>
</tr>
</tbody>
</table>

See upcoming part on “Evaluating Android’s Concurrency Frameworks”
Overview of Android Concurrency Frameworks

- Android’s concurrency frameworks are often used to decouple user interactions from computation & communication

Long-duration & (potentially) blocking operations run in background thread(s)
Overview of Android Concurrency Frameworks

- Android’s concurrency frameworks are often used to decouple user interactions from computation & communication
Overview of Android Concurrency Frameworks

- Android’s concurrency frameworks are often used to decouple user interactions from computation & communication.

See [github.com/douglascraigschmidt/POSA/tree/master/ex/M4/SimpleImageDownloads](github.com/douglascraigschmidt/POSA/tree/master/ex/M4/SimpleImageDownloads)
Overview of Android Concurrency Frameworks

- Android’s concurrency frameworks are often used to decouple user interactions from computation & communication.

Background threads perform long-duration image downloads.
Android’s concurrency frameworks are often used to decouple user interactions from computation & communication.

Synchronized message queue passes results from background thread(s) to UI thread.
Overview of Android Concurrency Frameworks

- Android’s concurrency frameworks are often used to decouple user interactions from computation & communication.
End of Android Concurrency Frameworks: Introduction