Evaluation of Concurrency & Parallelism Mechanisms in Java

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

Professor of Computer Science
Institute for Software Integrated Systems
Vanderbilt University
Nashville, Tennessee, USA
Learning Objectives in this Part of the Lesson

• Know which Java mechanism(s) to understand & apply
Which Java Mechanism(s) to Understand & Apply
Which Java Mechanism(s) to Understand & Apply

- Java’s concurrency & parallelism mechanisms span multiple layers in the software stack.
Which Java Mechanism(s) to Understand & Apply

- Java’s concurrency & parallelism mechanisms span multiple layers in the software stack
- Choosing best mechanism(s) depend on various factors

<table>
<thead>
<tr>
<th>Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Additional Application Frameworks</td>
</tr>
<tr>
<td>Concurrency/Parallelism Frameworks</td>
</tr>
<tr>
<td>Java Threads &amp; Synchronizers</td>
</tr>
<tr>
<td>Execution Environment (JVM, Dalvik/ART, etc.)</td>
</tr>
<tr>
<td>System Libraries</td>
</tr>
<tr>
<td>Operating System Kernel</td>
</tr>
</tbody>
</table>

| Java/JNI |
| C++/C |
| C |

5
Additional Application Frameworks

Operating System Kernel

Applications

System Libraries

Execution Environment (JVM, Dalvik/ART, etc.)

Concurrency/Parallelism Frameworks

Java/JNI

C++/C

Java Threads & Synchronizers

Which Java Mechanism(s) to Understand & Apply

• Developers of low-level classes & performance-sensitive apps may prefer shared object mechanisms e.g., `java.util.concurrent` as per [www.youtube.com/watch?v=sq0MX3fHkro](http://www.youtube.com/watch?v=sq0MX3fHkro)
Which Java Mechanism(s) to Understand & Apply

- Developers of low-level classes & performance-sensitive apps may prefer shared object mechanisms
  - **Pros:** Efficient & lightweight
  - **Cons:** Tedious & error-prone

Shared objects are often best used by infrastructure vs. app developers
Additional Application Frameworks

Operating System Kernel

Applications

System Libraries

Concurrency/Parallelism Frameworks

Java Threads & Synchronizers

Execution Environment (JVM, Dalvik/ART, etc.)

Which Java Mechanism(s) to Understand & Apply

• Framework developers may want to use the Java message passing mechanisms

e.g., Android AsyncTask/HaMeR frameworks or Java ExecutorCompetitionService
Which Java Mechanism(s) to Understand & Apply

• Framework developers may want to use the Java message passing mechanisms
  • **Pros:** Flexible & decoupled
  • **Cons:** Time/space overhead

May incur higher context switching, synchronization, & data movement overhead
Which Java Mechanism(s) to Understand & Apply

- Mobile app developers may want to program with higher-level frameworks

  - e.g., Java 8 parallel streams & completable futures, RxJava, etc.
Which Java Mechanism(s) to Understand & Apply

- Mobile app developers may want to program w/higher-level frameworks
  - **Pros**: Productivity & robustness
  - **Cons**: Time/space overhead & overly prescriptive
Which Java Mechanism(s) to Understand & Apply

“Full stack” developers should understand concepts & mechanisms at each layer
End of Evaluation of Concurrency & Parallelism in Java