Efficient Android Threading

Anders Göransson

Beijing • Cambridge • Farnham • Köln • Sebastopol • Tokyo
Table of Contents

Preface xi

1. Android Components and the Need for Multiprocessing 1
   Android Software Stack 1
   Application Architecture 2
     Application 3
     Components 3
   Application Execution 5
     Linux Process 6
     Lifecycle 6
   Structuring Applications for Performance 9
     Creating Responsive Applications Through Threads 9
   Summary 11

Part I: Fundamentals

2. Multithreading in Java 15
   Thread Basics 15
     Execution 15
   Single-Threaded Application 17
   Multithreaded Application 17
   Thread Safety 19
     Intrinsic Lock and Java Monitor 20
     Synchronize Access to Shared Resources 22
     Example: Consumer and Producer 24
   Task Execution Strategies 26
     Concurrent Execution Design 27
3. **Threads on Android.**

   Android Application Threads
   - UI Thread
   - Binder Threads
   - Background Threads

   The Linux Process and Threads
   - Scheduling

   Summary

4. **Thread Communication.**

   Pipes
   - Basic Pipe Use
   - Example: Text Processing on a Worker Thread

   Shared Memory
   - Signaling

   BlockingQueue

   Android Message Passing
   - Example: Basic Message Passing
   - Classes Used in Message Passing

   Message
   - Looper
   - Handler
   - Removing Messages from the Queue
   - Observing the Message Queue

   Communicating with the UI Thread

   Summary

5. **Interprocess Communication.**

   Android RPC
   - Binder

   AIDL
   - Synchronous RPC
   - Asynchronous RPC

   Message Passing Using the Binder
   - One-Way Communication
   - Two-Way Communication

   Summary

6. **Memory Management.**

   Garbage Collection

---

Summary

27

3. **Threads on Android.**

   Android Application Threads
   - UI Thread
   - Binder Threads
   - Background Threads

   The Linux Process and Threads
   - Scheduling

   Summary

4. **Thread Communication.**

   Pipes
   - Basic Pipe Use
   - Example: Text Processing on a Worker Thread

   Shared Memory
   - Signaling

   BlockingQueue

   Android Message Passing
   - Example: Basic Message Passing
   - Classes Used in Message Passing

   Message
   - Looper
   - Handler
   - Removing Messages from the Queue
   - Observing the Message Queue

   Communicating with the UI Thread

   Summary

5. **Interprocess Communication.**

   Android RPC
   - Binder

   AIDL
   - Synchronous RPC
   - Asynchronous RPC

   Message Passing Using the Binder
   - One-Way Communication
   - Two-Way Communication

   Summary

6. **Memory Management.**

   Garbage Collection

---

Summary

27

3. **Threads on Android.**

   Android Application Threads
   - UI Thread
   - Binder Threads
   - Background Threads

   The Linux Process and Threads
   - Scheduling

   Summary

4. **Thread Communication.**

   Pipes
   - Basic Pipe Use
   - Example: Text Processing on a Worker Thread

   Shared Memory
   - Signaling

   BlockingQueue

   Android Message Passing
   - Example: Basic Message Passing
   - Classes Used in Message Passing

   Message
   - Looper
   - Handler
   - Removing Messages from the Queue
   - Observing the Message Queue

   Communicating with the UI Thread

   Summary

5. **Interprocess Communication.**

   Android RPC
   - Binder

   AIDL
   - Synchronous RPC
   - Asynchronous RPC

   Message Passing Using the Binder
   - One-Way Communication
   - Two-Way Communication

   Summary

6. **Memory Management.**

   Garbage Collection

---

Summary

27

3. **Threads on Android.**

   Android Application Threads
   - UI Thread
   - Binder Threads
   - Background Threads

   The Linux Process and Threads
   - Scheduling

   Summary

4. **Thread Communication.**

   Pipes
   - Basic Pipe Use
   - Example: Text Processing on a Worker Thread

   Shared Memory
   - Signaling

   BlockingQueue

   Android Message Passing
   - Example: Basic Message Passing
   - Classes Used in Message Passing

   Message
   - Looper
   - Handler
   - Removing Messages from the Queue
   - Observing the Message Queue

   Communicating with the UI Thread

   Summary

5. **Interprocess Communication.**

   Android RPC
   - Binder

   AIDL
   - Synchronous RPC
   - Asynchronous RPC

   Message Passing Using the Binder
   - One-Way Communication
   - Two-Way Communication

   Summary

6. **Memory Management.**

   Garbage Collection

---

Summary

27
Thread-Related Memory Leaks 91
Thread Execution 92
Thread Communication 98
Avoiding Memory Leaks 101
Use Static Inner Classes 101
Use Weak References 101
Stop Worker Thread Execution 102
Retain Worker Threads 102
Clean Up the Message Queue 102
Summary 103

Part II. Asynchronous Techniques

7. Managing the Lifecycle of a Basic Thread ........................................ 107
   Basics 107
   Lifecycle 107
   Interruptions 108
   Uncaught Exceptions 110
   Thread Management 112
   Definition and Start 112
   Retention 114
   Summary 119

8. HandlerThread: A High-Level Queueing Mechanism .......................... 121
   Fundamentals 121
   Lifecycle 123
   Use Cases 124
      Repeated Task Execution 125
      Related Tasks 125
      Task Chaining 128
      Conditional Task Insertion 131
   Summary 131

9. Control over Thread Execution Through the Executor Framework .......... 133
   Executor 133
   Thread Pools 136
      Predefined Thread Pools 136
      Custom Thread Pools 137
   Designing a Thread Pool 138
   Lifecycle 142
   Shutting Down the Thread Pool 143
Thread Pool Uses Cases and Pitfalls

Task Management

Task Representation

Submitting Tasks

Rejecting Tasks

ExecutorCompletionService

Summary

10. **Tying a Background Task to the UI Thread with AsyncTask**

Fundamentals

Creation and Start

Cancellation

States

Implementing the AsyncTask

Example: Downloading Images

Background Task Execution

Application Global Execution

Execution Across Platform Versions

Custom Execution

AsyncTask Alternatives

When an AsyncTask Is Trivially Implemented

Background Tasks That Need a Looper

Local Service

Using execute(Runnable)

Summary

11. **Services**

Why Use a Service for Asynchronous Execution?

Local, Remote, and Global Services

Creation and Execution

Lifecycle

Started Service

Implementing onStartCommand

Options for Restarting

User-Controlled Service

Task-Controlled Service

Bound Service

Local Binding

Choosing an Asynchronous Technique

Summary

12. **IntentService**

Summary

viii | Table of Contents