



the
POWER
of
JAVA™



JavaOne
Part of the Network for Business Success

Swing Threading 101: An Introduction to the Event Dispatch Thread

Scott Delap
Rich Client Consultant

Scott Gelb
Enterprise Consultant

TS-4855

About Us

- Scott Delap
 - Editor, ClientJava.com
 - Author, Desktop Java Live
 - Rich Client Java™
Technology Consultant
 - Swing
 - Eclipse RCP
 - Ajax
- Scott Gelb
 - Enterprise Java Technology
Consultant
 - Member of the Java
Community
ProcessSM services



Goal

Learn best practices for creating responsive rich client applications in Swing using threads

Agenda

Introduction

SwingUtilities

Structured Threading Solutions

Debugging Techniques

Using AOP for Safeguards

Final Thoughts

Agenda

Introduction

SwingUtilities

Structured Threading Solutions

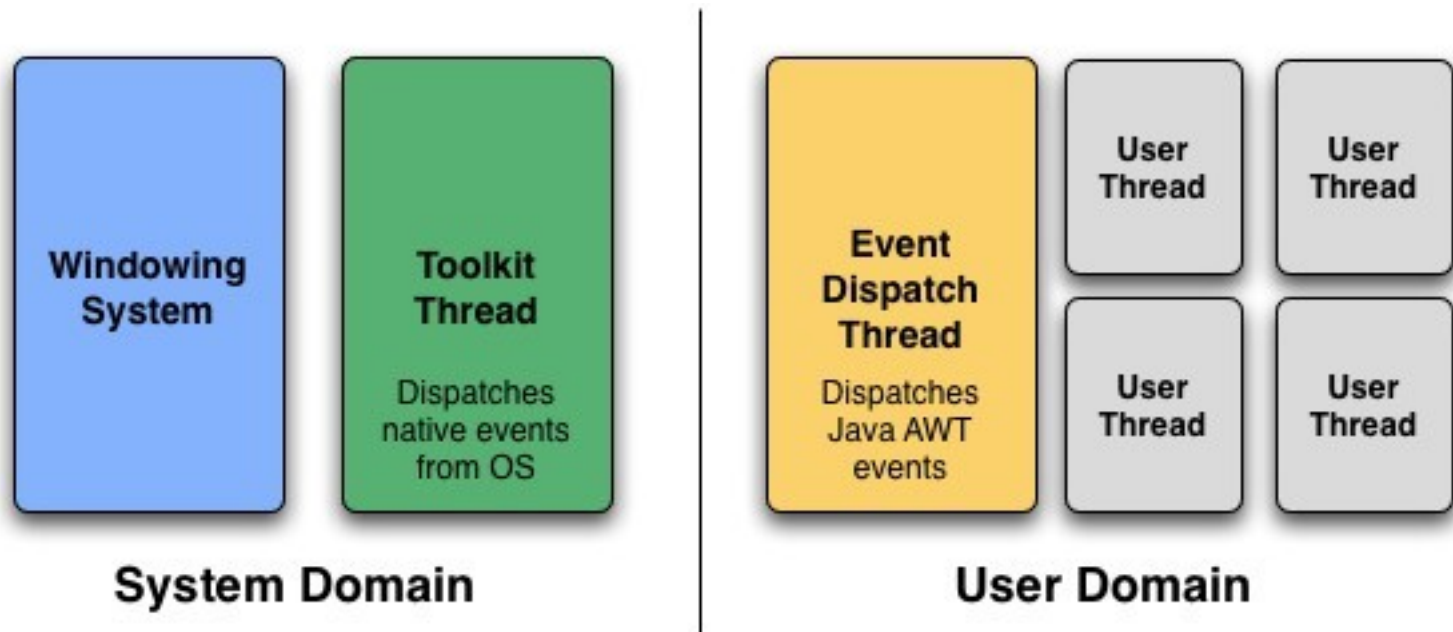
Debugging Techniques

Using AOP for Safeguards

Final Thoughts

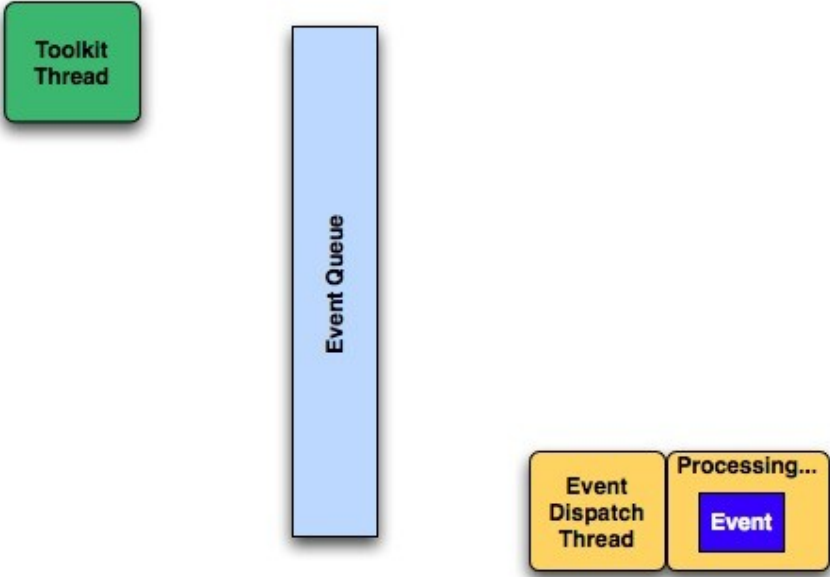
Introduction

OS, Java Technology, and Swing Threads



Introduction

Thread Interaction



Introduction

Swing Single Threading Rule

- The first version...
 - “Once a Swing component has been realized all code that might affect or depend on the state of that component should be executed in the event-dispatching thread.”

Introduction

Swing Single Threading Rule

- The rule today...
 - “All code that might affect or depend on the state of that component should be executed in the event-dispatching thread.”

Introduction

There Are a Few Thread Safe Methods...

- `invalidate()`
- `invalidate()`
- `repaint()`
- `add*Listener()`
- `remove*Listener()`
- Read the Java Docs

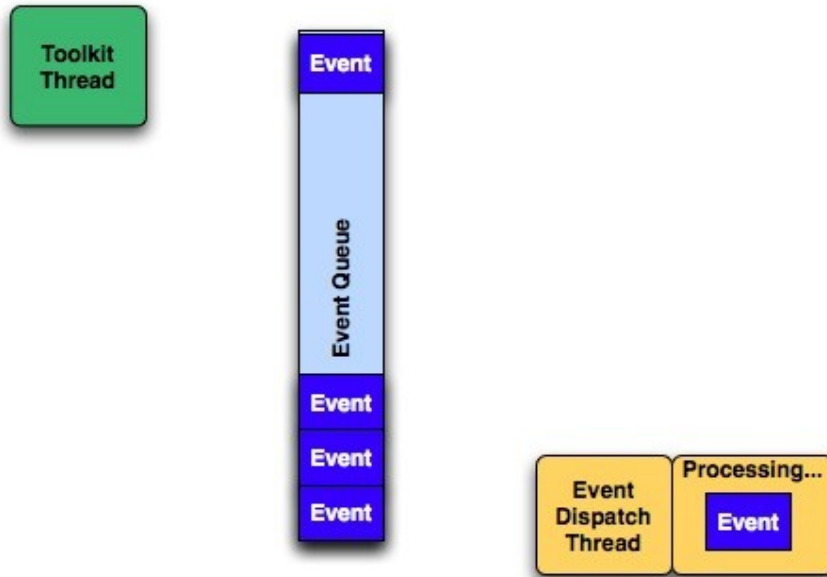
Introduction

Why Single Threading Is Good

- Developers don't need an in-depth understanding of threads
- Events are dispatched in a predictable order
- Less overhead
- Same approach used by other UI toolkits

Introduction

Why Single Threading Is Bad



DEMO

Why Single Threading Is Bad

Introduction

Common Causes of Single Threading Problems

- Long running event listeners
 - Socket communication
 - Processing expensive algorithms
 - Non UI-related work

Introduction

Solving Single Threading Problems

- Keep your event listeners lightweight
 - Multi-thread your application
 - Doesn't this violate the single threading rule?

Agenda

Introduction

SwingUtilities

Structured Threading Solutions

Debugging Techniques

Using AOP for Safeguards

Final Thoughts

SwingUtilities

Introduction

- Move work to a new thread
- Releases the Event Dispatch Thread (EDT) to paint, respond to mouse events, etc.
- What if you need to update the GUI?

SwingUtilities

Threading Helper Methods

- Swing provides two static methods to assist in multi-threading your application

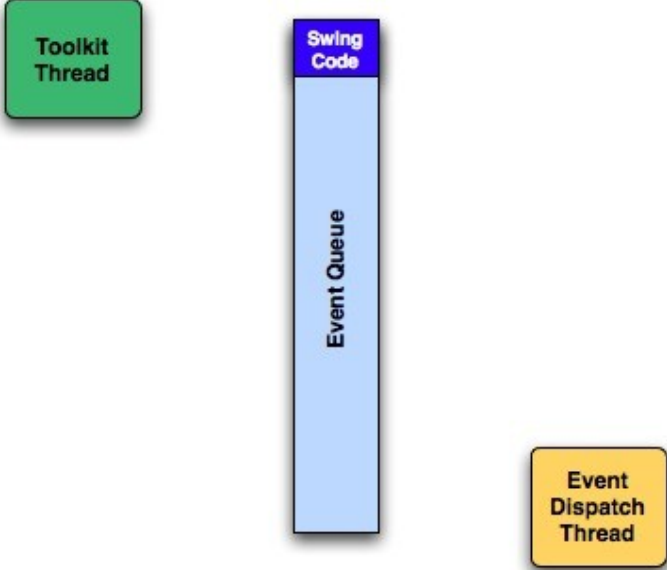
```
SwingUtilities.invokeLater( Runnable )
```

```
SwingUtilities.invokeAndWait( Runnable )
```

- These methods place the **Runnable** operations in the event queue for processing on the EDT

SwingUtilities

invokeLater()

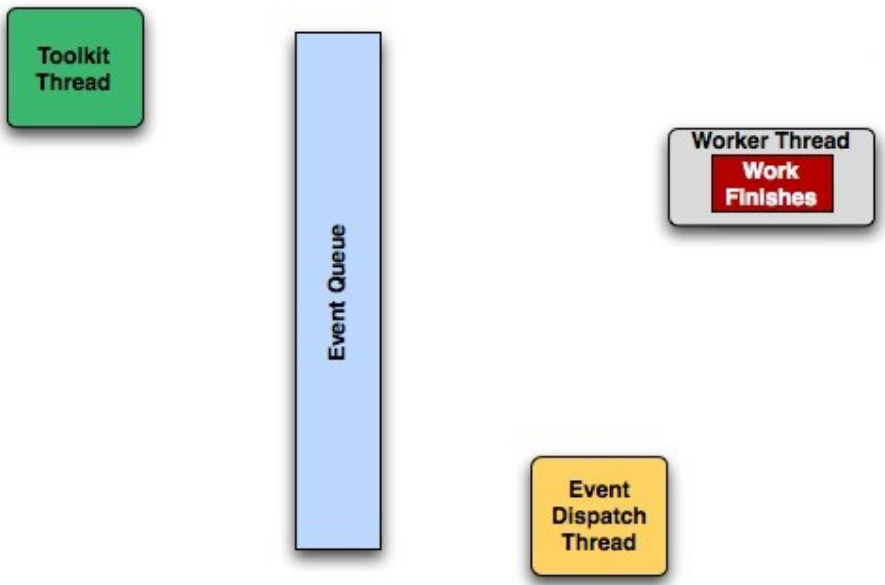


DEMO

Fixing Swing Threading Issues
With `SwingUtilities.invokeLater()`

SwingUtilities

invokeAndWait()



SwingUtilities

Using `InvokeAndWait`

- Blocks calling thread until event queue processes `Runnable`
- Useful for displaying modal dialogs
- Can cause deadlocks
- Use with caution

Agenda

Introduction

SwingUtilities

Structured Threading Solutions

Debugging Techniques

Using AOP for Safeguards

Final Thoughts

Structured Threading Solutions

- Asynchronous
 - SwingWorker
- Synchronous
 - Foxtrot

Structured Threading Solutions

SwingWorker

- Has moved frequently
 - Originally a source download of an article
 - Included as part of the JDNC Project
 - Included in Java SE 6
 - Java.net standalone project that backports to Java SE 5

Structured Threading Solutions

SwingWorker (Cont.)

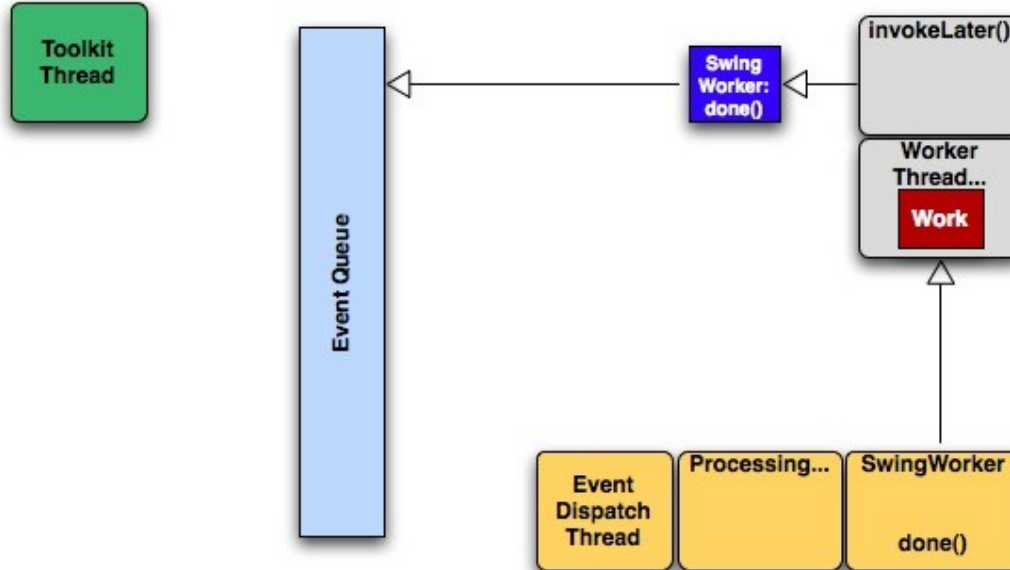
- Simplifies multi-threading process
 - Span new thread
 - Perform work
 - Update UI on EDT
 - Makes use of `java.util.concurrent`

DEMO

Fixing Swing Threading Issues
With SwingWorker

Structured Threading Solutions

SwingWorker Animation



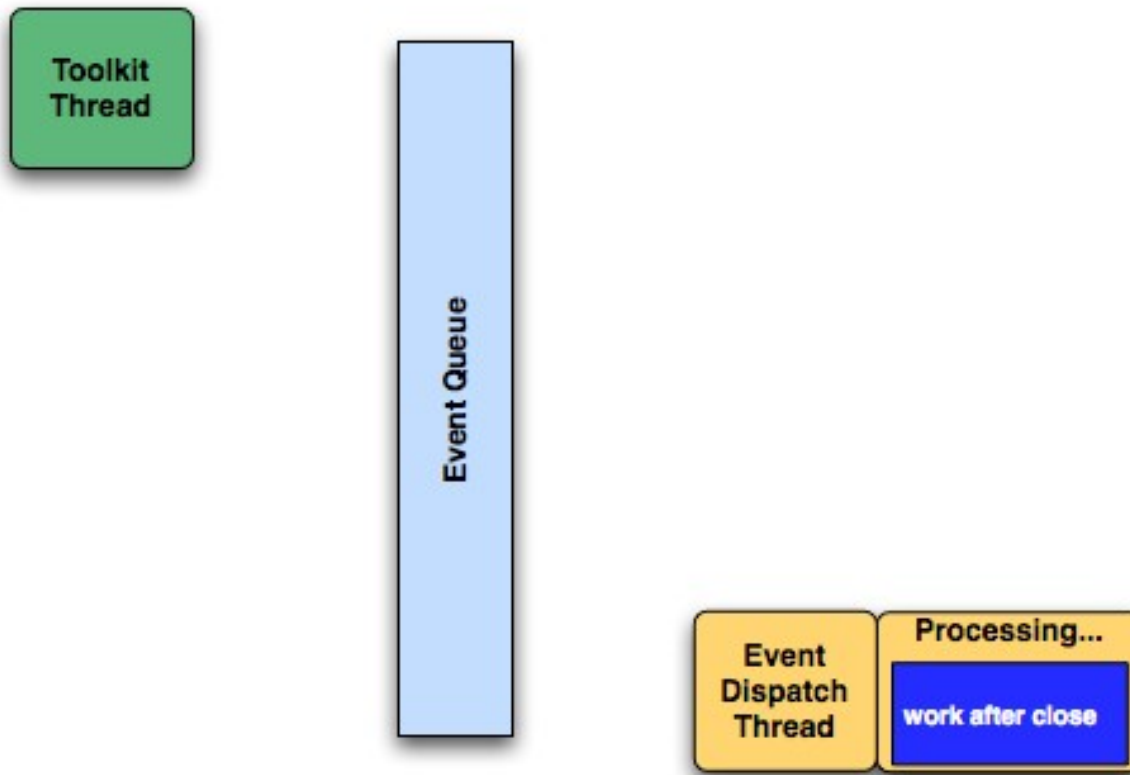
Structured Threading Solutions

Is There a Simpler Solution?

- SwingWorker is asynchronous
- Asynchronous programming can be complex
- Is there an easier way to simply “not tie up” the EDT?
- How do modal dialogs work?

Structured Threading Solutions

Modal Dialog Animation

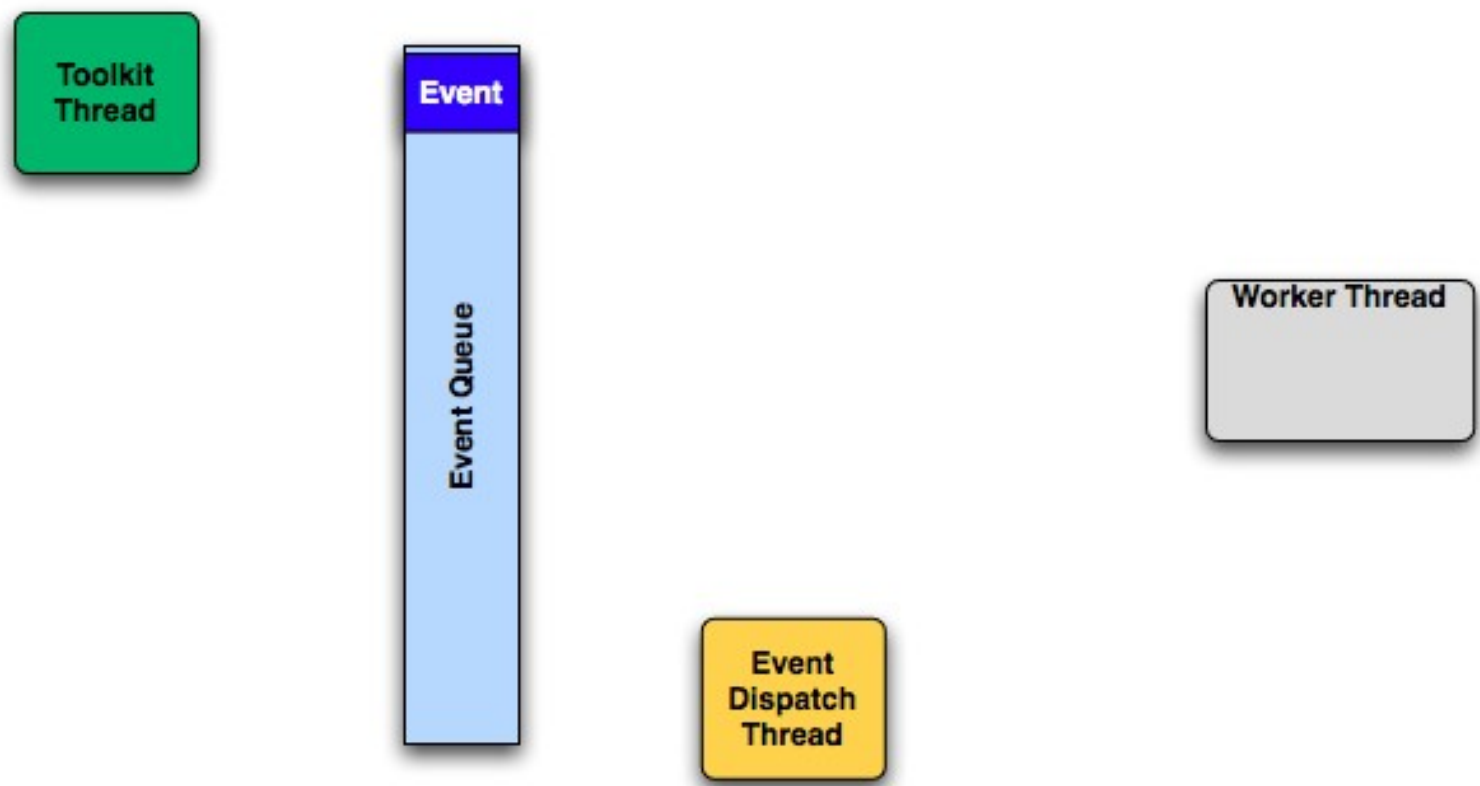


DEMO

Fixing Swing Threading Issues With Foxtrot

Structured Threading Solutions

Foxtrot Animation



Agenda

Introduction

SwingUtilities

Structured Threading Solutions

Debugging Techniques

Using AOP for Safeguards

Final Thoughts

Debugging Techniques

Is There an Easy Way to Find Violations?

- Audit all threads/runnables
- Finding the hotspots
 - Many Swing calls cause repaints
 - Have the repaint manager check the calling thread
 - **CheckThreadViolationRepaintManager**
 - Available on Java.net
 - <https://swinghelper.dev.java.net>

DEMO

Debugging Swing Threading With a Custom
RepaintManager

Agenda

Introduction

SwingUtilities

Structured Threading Solutions

Debugging Techniques

Using AOP for Safeguards

Final Thoughts

Using AOP for Safeguards

Can AOP Assist With Swing Threading?

- Aspect-oriented programming
 - Allows interception of method calls
 - Violations can be found
 - Optionally they can be automatically fixed
- **Your application is only as safe as your AOP advice!**

DEMO

Finding and Resolving Swing Threading
Issues With AOP

Agenda

Introduction

SwingUtilities

Structured Threading Solutions

Debugging Techniques

Using AOP for Safeguards

Final Thoughts

Final Thoughts

- Only multi-thread when necessary
- Observe the single threading rule and audit your Application regularly for violations
- Leverage built-in and open source tools to make threading easier
 - SwingUtilities
 - SwingWorker
 - Foxtrot

For More Information

- SwingWorker
 - <https://swingworker.dev.java.net/>
- Foxtrot
 - <http://foxtrot.sourceforge.net/>
- Spin
 - <http://spin.sourceforge.net>
- Desktop Java Live Threading Chapter
 - <http://www.sourcebeat.com/TitleAction.do?id=10>
- AspectJ and Swing Threading
 - <http://www.manning.com>
- Swing Debugging BOF
 - BOF-0204: <https://swinghelper.dev.java.net>

Q&A





the
POWER
of
JAVA™



JavaOne
Part of the Oracle and Sun Microsystems

Swing Threading 101: An Introduction to the Event Dispatch Thread

Scott Delap
Rich Client Consultant

Scott Gelb
Enterprise Consultant

TS-4855