

イベントディスパッチスレッド

[Java][Swing]

<http://java.sun.com/docs/books/tutorial/uiswing/concurrency/dispatch.html>

GUI スレッド

「ほとんどすべての GUI ツールキットはシングルスレッドのサブシステム (single-threaded subsystem) として実装されています。すなわちすべての GUI 活動が一つのスレッドに拘束されています。」

「特に意識的に完全にシングルスレッドのプログラムを書かないかぎり、普通の GUI アプリケーションでは、アプリケーションのスレッドと GUI のイベントスレッド、少なくともこの二つのスレッドが動きます。」

「昔は GUI のアプリケーションそのものがシングルスレッドで、GUI のイベントはアプリケーションのメインスレッドの中の "メインイベントループ (main event loop)" と呼ばれるループの中で処理されました。今日の GUI フレームワークはほんの少しだけ違って、イベントディスパッチスレッド (event dispatch thread, EDT) と呼ばれる、GUI イベントを処理する専用のスレッドを作ります。」

Java 並行処理プログラミング その「基盤」と「最新 API」を究める

Swing のイベントハンドリング

- ・ Swing のイベントハンドリングコードは、イベントディスパッチスレッドと呼ばれる特別なスレッド上で実行されます。
- ・ ほとんどのコードは、Swing のメソッドを同様にこのスレッドで、呼び出します。
- ・ これは、大抵の Swing オブジェクトのメソッドが、スレッドセーフでないため、必要です。

スレッドセーフでないメソッドの呼び出し

- ・ いくつかの Swing コンポーネントメソッドはスレッドセーフと API 仕様に記されています。
- ・ これらは、どんなスレッドからでも安全に呼び出せます。
- ・ その他の Swing コンポーネントメソッドは、イベントディスパッチスレッドから呼び出す必要があります。

プログラムが、このルールを無視すると、ほとんどの場合きちんと機能するのに、予測や再現性のないエラーが発生しやすくなります。

- ・ Java Swing 非同期処理の結果を GUI に表示する

コードがイベントディスパッチスレッド上で実行されているか

- ・ 短いタスクとして、コードがイベントディスパッチスレッド上で実行されているかを考慮することは実用的です。
- ・ ほとんどのタスクは、例えば、ActionListener.actionPerformed のような、イベントハンドリングメソッドとして呼び出されます。
- ・ 別のタスクは、invokeLater や invokeAndWait を使ってアプリケーションによりスケジュールされます。

- ・コードがイベントディスパッチ上で動いているか確認するには、`javax.swing.SwingUtilities.isEventDispatchThread` を呼び出します。

為に、イベントディスパッチスレッド上のタスクは、すばやく完了しなければなりません。

例

```
import java.awt.BorderLayout;
import java.awt.Dimension;

import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.SwingUtilities;

/**
 * @see http://java.sun.com/docs/books/tutorial/uiswing/components/frame.html
 */
public class JFrameTest {
    private JFrame frame;
    public static void main(String[] args) {
        SwingUtilities.invokeLater(
            new Runnable() {
                public void run() {
                    JFrameTest jft = new JFrameTest();
                    jft.createUI();
                }
            }
        );
    }
    private void createUI() {
        // Window の生成
        frame = new JFrame("FrameTest");
        // Window クローズ時にどうするか
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        // ダミーコンポーネント
        JLabel lbl = new JLabel("");
        lbl.setPreferredSize(new Dimension(200,100));
        // フレームにコンポーネントを乗せる
        frame.getContentPane().add(lbl, BorderLayout.CENTER);
        // サイズ調節
        frame.pack();
        // 表示
        frame.setVisible(true);

        // debug イベントディスパッチスレッドか?
        System.out.println("EventDispatchThread ? " + SwingUtilities.isEventDispatchThread());
    }
}
```