

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

Control de autenticación y
 acceso (Spring Security)
 UDDI
 Web Services
 Rest Services
 Social SSO
 SSO (Cas)

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)

❖ Estás en:

[Inicio](#)
[Tutoriales](#)
Ejemplo de Swing Worker: ¿Por qué se me congela la interfaz?



DESARROLLADO POR:
[Rubén Aguilera Díaz-Heredero](#)

Consultor tecnológico de desarrollo de proyectos informáticos.

Ingeniero en Informática, especialidad en Ingeniería del Software

Puedes encontrarme en [Autentia](#): Ofrecemos servicios de soporte a desarrollo, factoría y formación

Somos expertos en Java/J2EE

[Catálogo de servicios](#)
[Autentia](#)

Anuncios Google

[Java](#)

[Java Thread Analysis](#)

[String Java Tutorial](#)

Fecha de publicación del tutorial: 2011-01-22



Share |

[Regístrate para votar](#)

Ejemplo de Swing Worker: ¿Por qué se me congela la interfaz?

0. Índice de contenidos.

- 1. Entorno
- 2. Introducción
- 3. Para muestra un botón... o mejor dos
- 4. Conclusiones

1. Entorno

Este tutorial está escrito usando el siguiente entorno:

- Hardware: Portátil Mac Book Pro 17" (2,6 Ghz Intel Core i7, 8 GB DDR3)
- Sistema Operativo: Mac OS X Snow Leopard 10.6.4
- Maven 2.2.1
- Eclipse 3.6 (Helios) con M2Eclipse
- jdk 1.6

2. Introducción

Una de las tareas más comunes que tenemos que hacer cuando desarrollamos con Swing es la de actualizar la interfaz en función de los procesos que se van produciendo. En muchas ocasiones nos encontramos con la sorpresa de que la interfaz no se actualiza en el momento, si no que parece congelarse y sólo acaba actualizándose cuando el proceso finaliza.

La explicación a este fenómeno es sencilla, Swing todos los eventos relacionados con la interfaz los gestiona como una cola FIFO en un hilo independiente, conocido con el nombre de EDT (Event Dispatching Thread). Por tanto, si nosotros ejecutamos tareas pesadas dentro del radio de actuación de este hilo, estaremos bloqueando el resto de peticiones de actualización de la interfaz.

Para evitar bloquear el hilo EDT, Swing nos proporciona una clase llamada SwingWorker, que lo que hace es abrir otro hilo para ejecutar la tarea pesada, y así no bloquear el EDT.

3. Para muestra un botón... o mejor dos

Últimas Noticias

[Java API 1.5](#)

XIV Charla Autentia - ZK

Informática profesional: Las reglas no escritas para triunfar en la empresa. 2ª EDICIÓN ACTUALIZADA.

Pequeño coding dojo con Carlos Ble en las oficinas de Autentia.

Disponible gratis, Autentia Comic para el iPhone y iPad,

Comentando #AID2010. Agil Industrial Day 30 Nov 2010

Histórico de NOTICIAS

Últimos Tutoriales

Nut - Network UPS Tools

Awstats - Herramienta de generación de estadísticas.

Para demostrar este funcionamiento vamos a crear un proyecto donde vamos a tener un pantalla con un campo de texto y dos botones. El primer botón va a ejecutar su acción sin la utilización de Swing Worker y el segundo si va a utilizar Swing Worker.

Este es el código para la ventana comentada:

```
view plain print ?
01. package com.autentia.ejemplo;
02.
03. import java.awt.BorderLayout;
04.
05. import javax.swing.JButton;
06. import javax.swing.JFrame;
07. import javax.swing.JPanel;
08. import javax.swing.JTextField;
09. import javax.swing.SwingUtilities;
10. import javax.swing.WindowConstants;
11.
12. @SuppressWarnings("serial")
13. public class EjemploFrame extends JFrame {
14.
15.     public static int MAX_ITE = 100000;
16.
17.     private JTextField jtfInformacion = new JTextField();
18.     private JButton jbIncrementarSinSwingWorker = new JButton();
19.     private JButton jbIncrementarConSwingWorker = new JButton();
20.     private JPanel jpMarco = new JPanel();
21.     private JPanel jpAcciones = new JPanel();
22.
23.     public EjemploFrame(){
24.         initComponents();
25.     }
26.
27.     private void initComponents(){
28.         jpMarco.setLayout(new BorderLayout());
29.         jpMarco.add(jtfInformacion, BorderLayout.NORTH);
30.
31.         jbIncrementarSinSwingWorker.setAction(new IncrementarSinSwingWorkerAct
32.         jbIncrementarSinSwingWorker.setText("Incrementar sin swing worker");
33.         jbIncrementarConSwingWorker.setAction(new IncrementarConSwingWorkerAct
34.         jbIncrementarConSwingWorker.setText("Incrementar con swing worker");
35.
36.         jpAcciones.add(jbIncrementarSinSwingWorker);
37.         jpAcciones.add(jbIncrementarConSwingWorker);
38.
39.         jpMarco.add(jpAcciones, BorderLayout.CENTER);
40.
41.         setDefaultCloseOperation(WindowConstants.EXIT_ON_CLOSE);
42.
43.         getContentPane().add(jpMarco);
44.
45.         this.setTitle("Ejemplo funcionamiento Swing Worker");
46.
47.         pack();
48.     }
49.
50.     public JTextField getTextField(){
51.         return this.jtfInformacion;
52.     }
53.
54.     public static void main(String[] args) {
55.         SwingUtilities.invokeLater(new Runnable(){
56.             public void run(){
57.                 new EjemploFrame().setVisible(true);
58.             }
59.         });
60.     }
61. }
62.
63. }
```

Si nos fijamos en el método main de esta clase, veremos la utilización de la clase SwingUtilities, más concretamente el método invokeLater. Este método precisamente lo que hace es que el proceso que implementa se encole en la lista de eventos pendiente del hilo EDT. Una buena práctica es que todos los métodos que afecten a la interfaz gráfica, se ejecuten dentro de un invokeLater, para asegurarnos de que la gestión del hilo EDT se está haciendo correctamente.

La acción que ejecutan los dos botones va a ser la misma, ambos van a mostrar en el cuadro texto la cuenta ascendente de una variable hasta que se iguale al valor de la constante MAX_ITE.

En primer lugar vamos a desarrollar la acción sin el uso de Swing Worker:

 Utilización de Commons Digester para un sistema de preferencias configurable

 Introducción a Apache James

 MongoDB, primeros pasos

Últimos Tutoriales del Autor

 Utilización de Commons Digester para un sistema de preferencias configurable

 Ejemplo básico de Spring MVC Portlet

 Trabajando con los Web Services de Liferay

 Liferay IDE

 CAS: Validador personalizado

Síguenos a través de:



Últimas ofertas de empleo

2010-10-11
 Comercial - Ventas - SEVILLA.

2010-08-30
 Otras - Electricidad - BARCELONA.

2010-08-24
 Otras Sin catalogar - LUGO.

2010-06-25
 T. Información - Analista / Programador - BARCELONA.

view plain print ?

```
01. package com.autentia.ejemplo;
02.
03. import java.awt.event.ActionEvent;
04.
05. import javax.swing.AbstractAction;
06.
07. @SuppressWarnings("serial")
08. public class IncrementarSinSwingWorkerAction extends AbstractAction{
09.
10.     private EjemploFrame ejemploFrame;
11.
12.     public IncrementarSinSwingWorkerAction(EjemploFrame ejemploSinSwingWorkerF
13. {
14.         this.ejemploFrame = ejemploSinSwingWorkerFrame;
15.     }
16.
17.     public void actionPerformed(ActionEvent arg0) {
18.         int ite = 0;
19.         while (ite < EjemploFrame.MAX_ITE){
20.             ite = ite + 1;
21.             this.ejemploFrame.getTextField().setText("" + ite);
22.         }
23.     }
24. }
```

Si ejecutamos esta acción veremos el efecto del que hablábamos en la introducción, el botón se va a quedar como congelado, hasta que el proceso finaliza y muestra directamente el valor de MAX_ITE, sin mostrar realmente el conteo ascendente. Esto es porque la tarea está bloqueando el hilo EDT.

Para evitar esto la acción del otro botón la vamos a implementar utilizando Swing Worker, el código quedaría de esta forma:

view plain print ?

```
01. package com.autentia.ejemplo;
02.
03. import java.awt.event.ActionEvent;
04.
05. import javax.swing.AbstractAction;
06. import javax.swing.SwingWorker;
07.
08. @SuppressWarnings("serial")
09. public class IncrementarConSwingWorkerAction extends AbstractAction{
10.
11.     private EjemploFrame ejemploFrame;
12.
13.     public IncrementarConSwingWorkerAction(EjemploFrame ejemploSinSwingWorkerF
14. {
15.         this.ejemploFrame = ejemploSinSwingWorkerFrame;
16.     }
17.
18.     public void actionPerformed(ActionEvent arg0) {
19.         final SwingWorker worker = new SwingWorker(){
20.
21.             @Override
22.             protected Object doInBackground() throws Exception {
23.                 int ite = 0;
24.                 while (ite < EjemploFrame.MAX_ITE){
25.                     ite = ite + 1;
26.                     ejemploFrame.getTextField().setText("" + ite);
27.                 }
28.                 return null;
29.             }
30.
31.         };
32.
33.         worker.execute();
34.
35.     }
36. }
```

Si ahora ejecutamos la segunda acción veremos que el campo se actualiza casi secuencialmente hasta que llega al valor de MAX_ITE. Con lo que el problema queda resuelto, ya que la tarea se ejecuta en un hilo independiente y no bloquea el EDT. Otra forma de resolver el problema es crear nosotros mismos el hilo con la interfaz Runnable, pero si ya existe una forma estándar de hacerlo, mejor usamos la clase Swing Worker.

4. Conclusiones

Con esto espero haber ayudado a todos aquellos a los que una vez no supimos porque nuestra aplicación Swing no se comportaba como nosotros esperábamos.

Saludos.

Anímate y coméntanos lo que pienses sobre este **TUTORIAL**:

Puedes opinar o comentar cualquier sugerencia que quieras comunicarnos sobre este tutorial; con tu ayuda, podemos ofrecerte un mejor servicio.

Enviar comentario

(Sólo para usuarios registrados)

» **Registrate** y accede a esta y otras ventajas «

COMENTARIOS



Esta obra está licenciada bajo [licencia Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5](#)

Copyright 2003-2011 © All Rights Reserved | [Texto legal y condiciones de uso](#) | [Banners](#) | [Powered by Autentia](#) | [Contacto](#)

