

¿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 » Test de integración con Solr y el soporte de EmbeddedSolrServer.



Jose Manuel Sánchez Suárez

Consultor tecnológico de desarrollo de proyectos informáticos.

Puedes encontrarme en **Autentia**: Ofrecemos servicios de soporte a desarrollo, factoría y formación

Somos expertos en **Java/J2EE**

[Ver todos los tutoriales del autor](#)



Catálogo de servicios Autentia

JEE, JSF,
Hibernate, Spring,
Maven, SVN,
Liferay...



Fecha de publicación del tutorial: 2012-08-02 Tutorial visitado 3 veces [Descargar en PDF](#)

Test de integración con Solr y el soporte de EmbeddedSolrServer.

0. Índice de contenidos.

- 1. Introducción.
- 2. Entorno.
- 3. Configuración.
- 4. El código de nuestros tests.
 - 4.1. Parametrización en Eclipse.
- 5. Referencias.
- 6. Conclusiones.

1. Introducción

Cerrando el ciclo de tutoriales sobre Solr que hemos publicado últimamente en adictos, en este vamos a analizar cómo realizar pruebas de integración de la configuración de Solr con el soporte de EmbeddedSolrServer, un servidor embebido de Solr, y JUnit. EmbeddedSolrServer proporciona la misma interfaz de servicios que Solr pero sin necesidad de una conexión HTTP.

Tomando como referencia la configuración del tutorial sobre "Arrancar Solr desde un proyecto Maven con el soporte de Jetty" vamos a exponer qué dependencias a librerías necesitamos en el entorno de test y cómo escribiremos esos tests de integración.

El objetivo de los tests de integración contra Solr es comprobar que la definición del esquema y la configuración de filtros de recuperación e indexación se comportan como esperamos realizando consultas con el soporte de solrj.

2. Entorno.

El tutorial está escrito usando el siguiente entorno:

- Hardware: Portátil MacBook Pro 15' (2.4 GHz Intel Core i7, 8GB DDR3 SDRAM).
- Sistema Operativo: Mac OS X Lion 10.7.4
- Solr 3.5

3. Configuración.

Siguiendo con el soporte de Maven, esta sería la configuración de dependencias de nuestro pom.xml:

```

1  <dependency>
2  <groupId>org.apache.solr</groupId>
3  <artifactId>solr-test-framework</artifactId>
4  <version>3.5.0</version>
5  <scope>test</scope>
6  </dependency>
7  <dependency>
8  <groupId>javax.servlet</groupId>
9  <artifactId>servlet-api</artifactId>
10 <version>2.5</version>
11 <scope>test</scope>
12 </dependency>
13 <dependency>
14 <groupId>org.slf4j</groupId>
15 <artifactId>slf4j-simple</artifactId>
```



Síguenos a través de:

Últimas Noticias

- » La regla del Boy Scout y la Oxidación del Software
- » Autentia conquista los Alpes
- » Orientación a objetos y la importancia del "Tell, Don't Ask"
- » Autentia patrocina al Club KiteSurf Centro
- » Autentia patrocina el I Torneo Voley Playa Terrakas

[Histórico de noticias](#)

Últimos Tutoriales

- » Primeros pasos en Android.
- » Arrancar Solr desde un proyecto Maven con el soporte de Jetty.
- » Creación de un API REST protegido por OAuth2
- » Lectura y tratamiento de ficheros Excel con Talend (II): filtros y splits
- » Lectura y tratamiento de ficheros Excel con Talend (I):

```

16 <version>1.6.1</version>
17 <scope>test</scope>
18 </dependency>

```

Como vemos, todas con scope de **test**.

4. El código de nuestros tests.

Nuestros tests de integración no tendrían más que extender de la clase abstracta `AbstractSolrTestCase` y sobrecargar los métodos en los que se establece la ubicación de los ficheros de configuración.

```

1 package com.autentia.tutoriales;
2
3 import java.io.IOException;
4
5 import org.apache.solr.client.solrj.SolrQuery;
6 import org.apache.solr.client.solrj.SolrServer;
7 import org.apache.solr.client.solrj.SolrServerException;
8 import org.apache.solr.client.solrj.embedded.EmbeddedSolrServer;
9 import org.apache.solr.client.solrj.response.QueryResponse;
10 import org.apache.solr.common.SolrInputDocument;
11 import org.apache.solr.common.params.SolrParams;
12 import org.apache.solr.util.AbstractSolrTestCase;
13 import org.junit.After;
14 import org.junit.Before;
15 import org.junit.Test;
16
17 public class SolrSearchIntegrationTest extends AbstractSolrTestCase {
18
19     private SolrServer server;
20
21     @Override
22     public String getSchemaFile() {
23         return "schema.xml";
24     }
25
26     @Override
27     public String getSolrConfigFile() {
28         return "solrconfig.xml";
29     }
30
31     @Override
32     public String getSolrHome() {
33         return "";
34     }
35
36     @Before
37     @Override
38     public void setUp() throws Exception {
39         super.setUp();
40         server = new EmbeddedSolrServer(h.getCoreContainer(), h.getCore().getName())
41     }
42
43     @After
44     public void destroy() {
45         h.getCoreContainer().shutdown();
46     }
47
48     @Test
49     public void testThatDocumentExists() throws SolrServerException, IOException {
50         final SolrInputDocument document = new SolrInputDocument();
51         document.addField("id", "1");
52         document.addField("title", "Las reglas no escritas para triunfar en la empresa");
53         document.addField("type", "0");
54
55         server.add(document);
56         server.commit();
57
58         final SolrParams params = new SolrQuery("reglas");
59         final QueryResponse response = server.query(params);
60         assertEquals(1L, response.getResults().getNumFound());
61         assertEquals(1L, response.getResults().get(0).get("id"));
62     }
63 }
64

```

Con lo anterior, los ficheros de configuración de solr deberían ubicarse en el directorio `src/main/resources/`

En el código del test podemos ver que disponemos de:

- un método anotado con `@Before` que levanta el servidor embebido antes de cada test,
- un método anotado con `@After` que lo para después de cada test, y
- dentro del método de test propiamente dicho, el código que estimemos necesario para probar que podemos insertar nuevos documentos y recuperarlos.

4.1. Parametrización en Eclipse.

En este punto, el test debería pasar correctamente (con nuestro `schema.xml`) en el entorno de maven. Si bien, en el entorno de Eclipse se debería producir el siguiente error:

```

SolrSearchIntegrationTest [Runner: JUnit 4] Failure Trace
  java.lang.AssertionError: assertions are not enabled!
  at org.apache.lucene.util.LuceneTestCase.beforeClass(LuceneTestCase$4(LuceneTestCase.java:243)

```

Para solventarlo deberíamos habilitar las aserciones en el entorno de ejecución de JUnit, para ello, en preferencias > java >

nociones básicas.

Últimos Tutoriales del Autor

» Arrancar Solr desde un proyecto Maven con el soporte de Jetty.

» Lectura y tratamiento de ficheros Excel con Talend (II): filtros y splits

» Invocar a un servicio REST securizado, con el soporte de plantillas Spring.

» Indexación de documentos en Solr con el soporte de Talend.

» Configurar múltiples contextos de seguridad en Spring Security 3.1.

Últimas ofertas de empleo

2011-09-08

Comercial - Ventas - MADRID.

2011-09-03

Comercial - Ventas - VALENCIA.

2011-08-19

Comercial - Compras - ALICANTE.

2011-07-12

Otras Sin catalogar - MADRID.

2011-07-06

Otras Sin catalogar - LUGO.



Jose Manuel Sánchez
sanchezsuaresj

barrancoalberto Nuevo tutorial, Primeros pasos en Android. - [kcy.me/a6wu](#). Espero que os guste !!
12 hours ago · reply · retweet · favorite

sanchezsuaresj Arrancar #solr desde un proyecto #maven con el soporte de #jetty. - [kcy.me/a640](#) vía @adictosaltrabaj
yesterday · reply · retweet · favorite

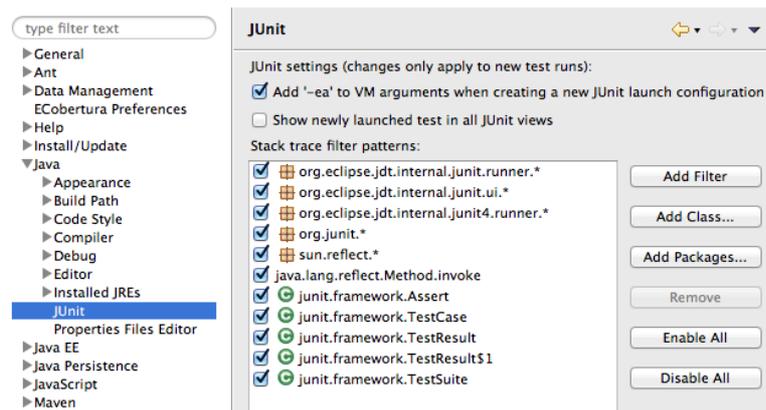
alejandropgarci Hablemos un poquito de #SOLID con: La regla del Boy Scout y la Oxidación del Software - [kcy.me/a5a9](#)
2 days ago · reply · retweet · favorite

sanchezsuaresj cerrando pendientes antes de vacaciones, estando seguro que la subida a

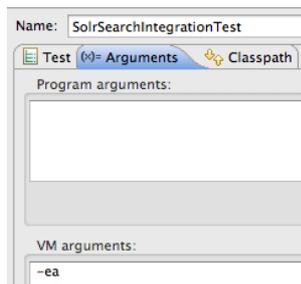


Join the conversation

jUnit debemos marcar la casilla "Add -ea to VM arguments", como sigue:



Para que aplique la nueva configuración debemos comprobar que en las propiedades de la ejecución del test > arguments > VM arguments se ha añadido la siguiente propiedad:



5. Referencias.

- <http://wiki.apache.org/solr/Solrj#EmbeddedSolrServer>
- <http://blog.synyx.de/2011/01/integration-tests-for-your-solr-config/>

6. Conclusiones.

Simple a la par que eficaz, ya podemos tener los mismos test de integración que podemos tener contra base de datos ;)

Con este tutorial damos por cubierta toda la gestión de configuración y configurabilidad arquitectónica de Solr en nuestros entornos de desarrollo.

Un saludo.

Jose

jmsanchez@autentia.com

A continuación puedes evaluarlo:

[Regístrate para evaluarlo](#)



Por favor, vota +1 o compártelo si te pareció interesante



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

» [Regístrate](#) y accede a esta y otras ventajas «



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

Copyright 2003-2012 © All Rights Reserved | [Texto legal y condiciones de uso](#) | [Banners](#) | [Powered by Autentia](#) | [Contacto](#)

