

¿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)



E-mail:
Contraseña:
Deseo registrarme
He olvidado mis datos de acceso

[Inicio](#) [Quiénes somos](#) [Tutoriales](#) [Formación](#) [Comparador de salarios](#) [Nuestro libro](#) [Charlas](#) [Más](#)

Estás en:

[Inicio](#) [Tutoriales](#) Búsquedas "facetadas" en Solr con el soporte de Solrj.



DESARROLLADO POR:
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

[Catálogo de servicios Autentia](#)



Fecha de publicación del tutorial: 2009-02-26



Share |

[Regístrate para votar](#)

Búsquedas "facetadas" en Solr con el soporte de Solrj.

0. Índice de contenidos.

- 1. Introducción.
- 2. Entorno.
- 3. Búsqueda de productos facetados a categorías.
- 4. Búsqueda de productos facetados por rango de precios.
- 5. Referencias.
- 6. Conclusiones.

1. Introducción

Después de la [introducción a Apache Solr](#), y de ver [el soporte de solrj para indexar y recuperar documentos desde java](#), en este tutorial vamos a estudiar cómo recuperar metainformación asociada a los resultados de una consulta.

Las facetes para Solr son subconjuntos de información asociados a los resultados de una consulta que nos permiten agrupar los mismos por distintos campos de los documentos.

El ejemplo más típico es el de la búsqueda de productos en el que además, recuperamos información sobre cuántos de esos productos pertenecen a cada una de las categorías de productos existentes. De este modo podremos después afinar la búsqueda de productos, restringiendo los resultados a los productos asociados a dichas categorías.

Sin el soporte de Solr es lo que, contra una base de datos relacional, tendríamos que hacer con un group by y un count(*)).

En Solr, se puede realizar esta operativa en una única consulta, de modo que el resultado de la búsqueda nos devuelva dicha información adicional, sobre el contenido de la misma.

Además, a partir de la versión 3.1 de Solr, podemos indicar en la consulta que las facetes las cree por rangos, de modo que podríamos recuperar los productos facetados por rangos de precio.

2. Entorno.

El tutorial está escrito usando el siguiente entorno:

- Hardware: Portátil MacBook Pro 17' (2.93 GHz Intel Core 2 Duo, 4GB DDR3 SDRAM).
- Sistema Operativo: Mac OS X Snow Leopard 10.6.7
- Apache Solr 3.1.
- Eclipse Helios SR2 con m2eclipse
- Junit 4.8.2

3. Búsqueda de productos facetados a categorías.

Últimas Noticias

- [theEvt 2011: Evento de Tecnología & Negocio en la Web](#)
- [Proxima charla en TheEvt: La Technicenta, de programador a empresario](#)
- [XVI Charla Autentia - Refactoring y Clean Code - Cambio de fecha](#)
- [XV Charla Autentia - web2py \(y Google App Engine\) - Vídeos y Material](#)
- [XVI Charla Autentia - Refactoring y Clean Code](#)

Histórico de NOTICIAS

Últimos Tutoriales

- [Construcción de un control personalizado en Android](#)
- [Indexación y recuperación de documentos en Apache Solr haciendo uso del api para Java.](#)
- [Creación de un portlet con Primefaces](#)
- [Introducción a Apache Solr.](#)
- [Session TimeOut en JSF2 con el soporte de Primefaces.](#)

Últimos Tutoriales del Autor

Añadir una faceta a una consulta es tan simple como invocar a los siguientes métodos:

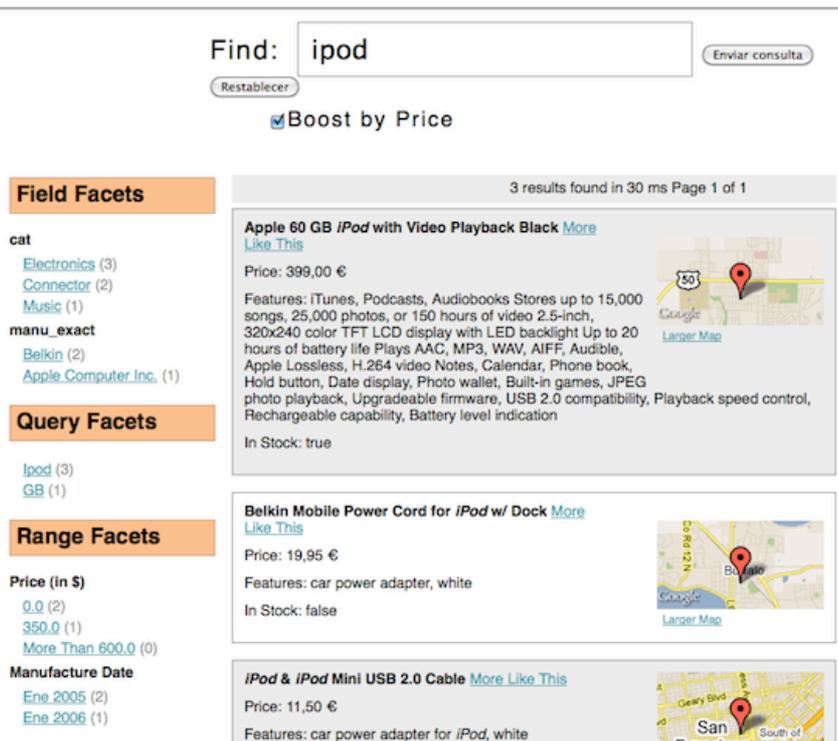
```
1 query.setFacet(true);
2 query.addFacetField("cat");
```

Primero indicamos que queremos hacer uso de facetas y después el o los campos por los que queremos agrupar esa metainformación, en el ejemplo por el campo categoría de producto.

El resultado de una consulta facetada se puede leer de la siguiente forma:

```
01 @Test
02 public void retrieveCategoryFacetedResult() throws SolrServerException{
03     final SolrQuery query = new SolrQuery();
04     query.setQuery("name:ipod");
05     query.setFacet(true);
06     query.addFacetField("cat");
07
08     final QueryResponse response = server.query(query);
09
10     final FacetField categoryFacet = response.getFacetField("cat");
11
12     assertFaceted(categoryFacet, "electronics", 0, 3);
13     assertFaceted(categoryFacet, "camera", 3, 0);
14
15 }
16
17 private void assertFaceted(FacetField facet, String name, int categoryPositionInFacet,
18     int productsInCategory) {
19     Assert.assertEquals(name, facet.getValues().get(categoryPositionInFacet).getName());
20     Assert.assertEquals(productsInCategory, facet.getValues().get(categoryPositionInFacet).getCount());
21 }
```

Con ello, lo que conseguimos es poder añadir la siguiente información al resultado de las búsquedas, de modo que el usuario pueda restringir la misma en función a subconjuntos.



4. Búsqueda de productos facetados por rango de precios.

Si quisieramos añadir rangos a una faceta, por ejemplo para recuperar productos en función a un rango de precios, podríamos hacer algo como lo que sigue:

```
01 @Test
02 public void retrievePriceFacetedResult() throws SolrServerException{
03     final SolrQuery query = new SolrQuery();
04
05     query.setQuery("name:ipod");
06     query.setFacet(true);
07
08     query.add("facet.range", "price");
09     query.add("f.price.facet.range.start", "0");
10     query.add("f.price.facet.range.gap", "100");
11     query.add("f.price.facet.range.end", "1000");
12
13     final QueryResponse response = server.query(query);
14     // response.getFacetDates() TODO: no existe un getFacetRanges ?
15     NamedList counts = (NamedList) ((SimpleOrderedMap)
16         ((SimpleOrderedMap) ((SimpleOrderedMap)
17             response.getResponse().get("facet_counts")).get("facet_ranges")).get("price")).get("counts");
18
19     Assert.assertEquals(2, counts.get("0.0"));
20     Assert.assertEquals(1, counts.get("300.0"));
21 }
```

A comentar:

- Indexación y recuperación de documentos en Apache Solr haciendo uso del api para Java.
- Introducción a Apache Solr.
- Session Timeout en JSF2 con el soporte de Primefaces.
- Primeros pasos con github: subir un proyecto al repositorio.
- Habilitar autocompletado de etiquetas para JSF en un proyecto Eclipse gestionado por Maven.

Síguenos a través de:



Últimas ofertas de empleo

- 2011-03-02 T. Información - Analista / Programador - MALAGA.
- 2011-02-24 T. Información - Especialista CRM - MADRID.
- 2011-02-16 Marketing - Experto en Marketing - CADIZ.
- 2011-02-08 Comercial - Ventas - CADIZ.
- 2011-01-28 Comercial - Ventas - SEVILLA.

Jose Manuel Sánchez sanchezsuarezj

@borjasant Stay tuned! :)
about 1 minute ago · reply · retweet · favorite

@fbricon thanks!!!, but until the next version incoming, i afraid that we'll work with IAM :(
4 minutes ago · reply · retweet · favorite

Indexación y recuperación de documentos en Apache #Solr haciendo uso del api para Java.:<http://bit.ly/jGQ9Cv>
7 hours ago · reply · retweet · favorite

Creación de un portlet con Primefaces:
<http://bit.ly/infPya>
twitter

Join the conversation

- facet.range: asigna el campo a facetar por rango,
- f.price.facet.range.start: permite establecer el valor mínimo de precio a facetar,
- f.price.facet.range.end: permite establecer el precio máximo a facetar,
- f.price.facet.range.gap: establece el salto entre cada uno de los rangos de la faceta
- llama la atención que no exista un método directo, o al menos yo no lo he encontrado, para recuperar la faceta por rango. Si bien, sí existe para las fechas (getFacetDates()) no existe un método análogo en el api de solrj para recuperar las facetas por rango, de ahí la necesidad de recuperarlo manualmente del contenido de la respuesta conociendo las claves de los mapas.

5. Referencias.

- <http://wiki.apache.org/solr/Solrj>

6. Conclusiones.

Aunque parece que el api de java aún no está preparado para soportar todo lo que si podemos hacer con xml o json, seguimos avanzando en el sorprendente mundo de Solr ;).

Un saludo.

Jose

jmsanchez@autentia.com

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)

» **Regístrate** 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

