

How to add additional fields and values to Solr discovery index

Motivation: Search for filenames and file descriptions in DSpace

Out of the box discovery in DSpace 6.0 worked quite well. But it did not include filenames and file descriptions in the search index.

As we planned to allow research data publications on our repository, there would be publications with many files (as opposed to text publications which usually consist of only one pdf) and it would be nice to be able to search for filenames and file descriptions.

Add additional fields and values to Solr index

Sometimes extending DSpace turns out to be very easy and straightforward, albeit not very well documented.

To add fields and values to the discovery index, just create a new implementation of the `SolrServiceIndexPlugin` class and add your fields and values to the document in the `additionalIndex` method:

```
package org.dspace.discovery;
import org.apache.solr.common.SolrInputDocument;
import org.dspace.content.DSpaceObject;
import org.dspace.content.Item;
import org.dspace.core.Context;
public class SolrServiceHelloWorldPlugin implements SolrServiceIndexPlugin {
    @Override
    public void additionalIndex(Context context, DSpaceObject dso, SolrInputDocument document) {
        if (dso instanceof Item) {
            Item item = (Item) dso;
            document.addField("greeting", "Hello World!");
        }
    }
}
```

Add your plugin to `discovery.xml` as a bean:

```
<bean id="solrServiceHelloWorldPlugin" class="org.dspace.discovery.SolrServiceHelloWorldPlugin"/>
```

Recompile, distribute and rebuild the discovery index:

```
[dspace]/bin/dspace index-discovery -b
```

Add multivalued fields

By default, a new field added this way will be multivalued. To add multiple values to such a field in a Solr document you can either call the `addField` method several times with the same field name:

```
document.addField("greeting", "Hello World!");
document.addField("greeting", "Bonjour Monde!");
```

Or call it once with a collection as value:

```
String[] greetings = {"Hello World!", "Bonjour Monde!"};
document.addField("greeting", greetings);
```

SolrServiceFileInfoPlugin

This is what our `SolrServiceFileInfoPlugin` looks like:

```

package org.dspace.discovery;

import org.apache.solr.common.SolrInputDocument;
import org.dspace.content.Bitstream;
import org.dspace.content.Bundle;
import org.dspace.content.DSpaceObject;
import org.dspace.content.Item;
import org.dspace.core.Context;

import java.util.List;

public class SolrServiceFileInfoPlugin implements SolrServiceIndexPlugin
{
    private static final String BUNDLE_NAME = "ORIGINAL";
    private static final String SOLR_FIELD_NAME_FOR_FILENAMES = "original_bundle_filenames";
    private static final String SOLR_FIELD_NAME_FOR_DESCRIPTIONS = "original_bundle_descriptions";

    @Override
    public void additionalIndex(Context context, DSpaceObject dso, SolrInputDocument document)
    {
        if (dso instanceof Item)
        {
            Item item = (Item) dso;
            List<Bundle> bundles = item.getBundles();
            if (bundles != null)
            {
                for (Bundle bundle : bundles)
                {
                    String bundleName = bundle.getName();
                    if ((bundleName != null) && bundleName.equals(BUNDLE_NAME))
                    {
                        List<Bitstream> bitstreams = bundle.getBitstreams();
                        if (bitstreams != null)
                        {
                            for (Bitstream bitstream : bitstreams)
                            {
                                document.addField(SOLR_FIELD_NAME_FOR_FILENAMES, bitstream.getName());

                                String description = bitstream.getDescription();
                                if ((description != null) && (!description.isEmpty()))
                                {
                                    document.addField(SOLR_FIELD_NAME_FOR_DESCRIPTIONS, description);
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}

```