SITECORE DATASOURCE INDEXER

by Johann Baziret (Sidewalk)

CONTENTS

INTRODUCTION

This document describes what is the Sitecore Datasource Indexer module, how to install and use it.

The Datasource Indexer module extends Sitecore Content Search by allowing to index the actual content of your pages and not only the items. You can then implement a true free text search on your website without needing a third party search engine to crawl your pages.

This modules indexes not only the page item itself but also the content of the items referenced as datasource of the sublayouts on your web pages. Moreover it allows for the developer to fine-tune what is exactly indexed in the pages.

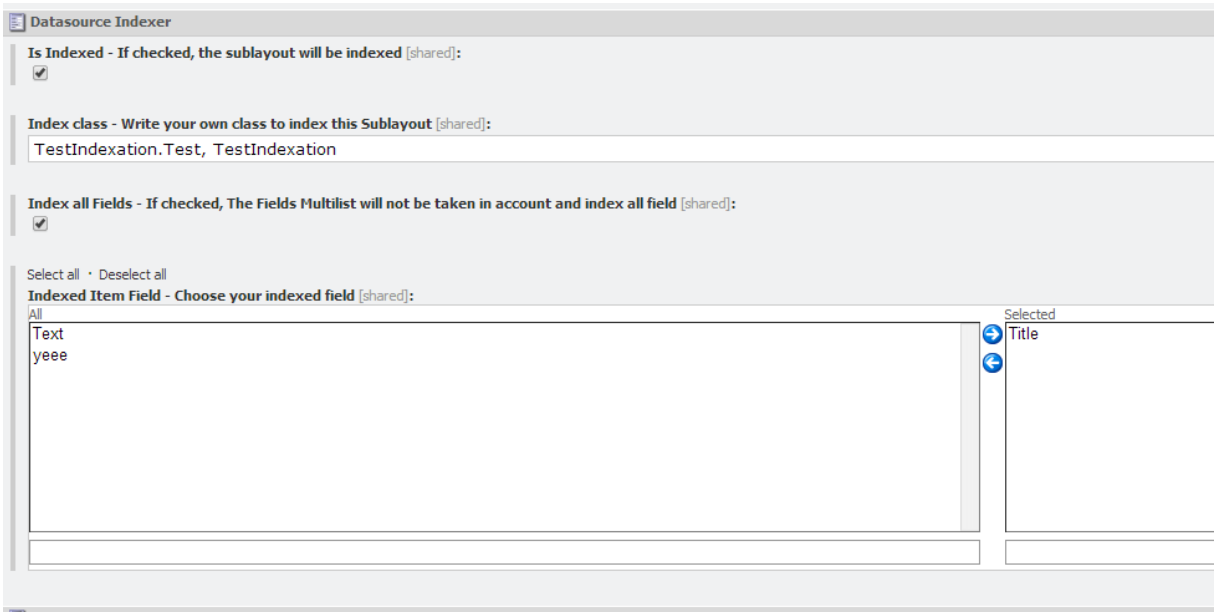
INSTALLATION

Simply install the package via the Sitecore Installation Wizard.

USAGE

Sublayouts Configuration

In order to define how to index a sublayout datasource, new fields have been added to Sitecore sublayouts. You can set those options on a sublayout to fine-tune how the datasource items, if any, will be indexed.



The new section – **Datasource Indexer** – includes the following fields:

- **Is Indexed** – If checked, it will index the sublayout datasource item.
- **Index Class** – If you want to have the hands on the content to be indexed for a specific sublayout, you can create a class which inherit from `DatasourceComputed` and overrides the method `public string Run(Item,RenderingSettings)` that returns the text of the item to be indexed. Specify your class with namespace in this field separated by a comma. Be aware that specifying a value for this field disables all the options hereafter.

- **Index All Fields** – If checked, it will index all the fields of the datasource item without taking into account the 'Indexed Item Field'.
- **Indexed Item Field** – Specifies the fields of the datasource item to index. The available fields are the fields of the template specified as **Datasource Template** and so they are only available if the datasource template is specified. Be aware that **Index All Fields** must be unchecked in order for this option to be taken into account.

Search on Page Content

Here is an example to how to search into the Content field:

```
using (var sC = ContentSearchManager.GetIndex("your indexname").CreateSearchContext())
 {
    var r = sC.GetQueryable<SearchResultItem>().Where(i =>
i.Content.Contains("Datasource Item Text")).Select(i => i.GetItem());
 }
```

IMPLEMENTATION DETAILS

When saving a datasource item, the item is reindexed and the `GetDependencies` pipeline is triggered which gathers all the pages referencing this item and which have to be reindexed.

The module writes the result of his indexation into the `_content` field of the Sitecore Lucene index. In Sitecore index configuration, by default, `_content` is a calculated field handled by the class `MediaItemContentExtractor`. This class indexes the content of the document from a media item. This module overrides this setting but still calls the `MediaItemContentExtractor` class if the item indexed is a media item.

KNOWN LIMITATIONS AND FUTURE IMPROVEMENTS

Here are some of the know limitations and possible improvements for this module:

- The author cannot specify for each rendering component which field to index (Maybe in a future version)
- The Build query feature on the Datasource is not supported. (LinkDatabase does not contains the result of a query)

SPECIAL THANKS & REFERENCES

Blog, projects, persons who helped during the development of the module:

- **Content Usage Tool** – https://github.com/team-orange/sitecore-content-usage-tools
- **Techopria414 blog** – http://www.techphoria414.com/Blog/2013/November/Sitecore-7-Computed-Fields-All-Templates-and-Datasource-Content
- **Jeremy Coste**, for the brainstorming, the idea and the review
- **Robin Hermanussen** (http://hermanussen.eu/sitecore/wordpress), for his help during the development (**z**DatasourceIndexer.config !!)