

Datasource Indexer

User Manual

Contents

Datasource Indexer	1
Introduction.....	2
References.....	2
Installation.....	2
Usage	2
To know	3

Introduction

This document describes how to use the Datasource Indexer Modules. It gives information to how to install the modules and how to use it.

The modules provides you a way to index your Datasource item by adding them on the `_content` field of the referred item.

References

Blog, projects, persons that help during the development of the module.

Content Usage Tool : <https://github.com/team-orange/sitecore-content-usage-tools>

Techopria414 : <http://www.techphoria414.com/Blog/2013/November/Sitecore-7-Computed-Fields-All-Templates-and-Datasource-Content>

Jeremy Coste, for the brainstorming and the idea

Robin Hermanussen: <http://hermanussen.eu/sitecore/wordpress> , for his help during the development (zDatasourceIndexer.config !!)

Installation

Simply install the packages via the Sitecore Installation Wizard.

Usage

To define how to index a datasource, a new section is added on each sublayout. Some fields are dependent of the Datasource Template of the Sublayout.

The new section (Datasource Indexer) is composed by:

- Is Indexed : If the checkbox is check, he will index the sublayout datasource.
- Index Class: If you have some custom field that you want to index, or index a different way, you can create a class which inherit from `DatasourceComputed` and override the method `string Run(Item)`, then simply add the Namespace, and the assembly in the single line text.

- Index All Field: If this checkbox is checked, you will index all the field of the datasource template set on the sublayout.
- Multi List Field: Choose which field to index, take care that if you use that multi list, index all field should not be checked.

When you save a datasource item, he will trigger the GetDependencies pipelines which will reindex the page referred to the datasource.

To know

The module write the result of his indexation into the `_content` field, I let the default behavior of `MediaItemContentExtractor`, if he matches a media Item.

This module has his limitation and they are the following:

- You cannot specify for each Rendering which field to index (Maybe in a future version)
- If you play with the BasePage on the Datasource Template, he will not fetch all the field of a more complex datasource.