AdvSearch **1.0.0 pl**

Date: 25/01/2012

Dynamic content search add-on that supports results highlighting, faceted search and search in custom packages.

Author: *Coroico <coroico@wangba.fr>*

Before to install the AdvSearch package, thanks to read the installation chapter first.

AdvSearchForm

Basic usage: **[[!AdvSearchForm]]**

PARAMETERS

&asId

Purpose: Unique id for AdvSearch instance
Option: a-z, _ , 0-9 (case sensitive)
Default: ‘as0’
Note: optional. Used to distinguish several AdvSearch add-on instances on the same page. This token is used to link the snippet calls between them. Choose a short name. eg: ‘as2’.
Placeholders: -
Related: -

&clearDefault

Pur: clearing default text.
Opt:

- 0 : not activated
- 1 : activated.

Def: 1
Not: Optional; Include the clear default js function in header and add the class "cleardefault" to the input text form.

&help

Pur: whether to set or not a link to help window.
Opt: 0 (without help) | >0 (with help)
Def: 1 => with help
Not: optional.
The help is provided by an ajax request to a resource named “AdvSearchHelp handler” which call the AdvSearchHelp snippet. If you would like use an another ajax handler, specify his resource id in the help parameter (different of 0 or 1). This handler should return a html document.

Related:
[[+helpLink]] : placeholder where to set the help link image on the document.
[[$HelpLink]] : link to AdvSearch help

&addCss

Pur: to add the default css file to the web pages automatically.
Opt: 0 | 1
0: not included. You should add the css manually.
1: included before the closing HEAD tag
Def: 1
Not: optional.

&addJs

Pur: to add the javascript files to the web pages automatically.
Opt: 0 | 1 | 2
0: not included. You should add them manually.
1: included before the closing HEAD tag
2: included before the closing BODY tag
Def: 1
Not: optional.

&jsSearchForm

Pur: full path name of the javascript file to link with the form. This js file include the clearDefault function and could be modified to add custom features.
Opt: full path name under /assets
Def: ‘assets/components/advsearch/js/advSearchForm.min.js’
Not: optional

&landing

Pur: the document id that the AdvSearch snippet is called on, that will display the results of the search.
Opt: resource id
Def: 0 => current document id
Not: Mandatory if you would like display the results on another page.

&method

Pur: whether to send the search over POST or GET.
Opt: ‘POST’ | ‘GET’
Def: ‘GET’
Not: optional

&searchIndex

Pur: the name of the REQUEST parameter that the search will use.
Opt: string
Def: ‘search’
Not: optional

&searchString

Pur: to initialize the search with a default searchString. Used in conjunction with &init=`all` this allow to display search results for a specific search string.
Opt: query string
Def: '' – ‘empty string)
Not: optional

&toPlaceholder

Pur: whether to set the output to directly return, or set to a placeholder with this propertys name.
Opt: placeholder name
Def: ‘’ (empty string) => directly returned
Not: optional

&tpl

Pur: the chunk that will be used to display the search form.
Opt: string
Def: ‘AdvSearchForm’
Not: optional

&withAjax

Pur: whether to set on a new document (page reload), or set the results in div section thru ajax.
Opt: 0 (false) | 1 (true)
Def: 0 => non-ajax mode
Not: optional

Ajax mode Parameters

&ajaxResultsId

Pur: The document id which hold the AdvSearch snippet call. This document is called through ajax. It should be a plain/text document with empty template.
Opt: resource id
Not: Mandatory with the ajax mode

&addJQuery

Pur: to add the jquery library to the web pages automatically.
Opt: 0 | 1 | 2
0: not included
1: included before the closing HEAD tag
2: included before the closing BODY tag
Def: 1
Not: optional.

&jsJQuery

Pur: Location of the Jquery javascript library
Opt: path to the Jquery library
Def: ‘assets/components/advsearch/js/jquery-1.5.1.min.js’
Not: optional. change this parameter to set an newer jquery release.

&liveSearch

Pur: to use the live search (i.e. results as typing)
Opt: 0 | 1
Def: 0 - livesearch mode inactivated
Not: optional

&urlScheme

Pur: indicates in what format the URL is generated.
Opt: -1, full, abs, http, https
Def: -1
Not: optional
-1 : (default value) URL is relative to site_url

See:
http://rtfm.modx.com/display/revolution20/modX.makeUrl

Chunks

AdvSearchForm

Pur: default chunk provided to style a search form

Should contain:
- a form id: **id="[[+asId]]"**
- an action: **action="[[~[[+landing]]]]"**
- a method: **method="[[+method]]"**
- an input of type hidden named asId: **<input type="hidden" name="asId" value="[[+asId]]" />**

And possibly:
- an input text named **"[[+asId]]"_search**

HelpLink

Pur: default chunk provided to style the link to open the advSearch help window
Should contain an id: **id="[[+asId]]_helplink"**

ResultsWindow

Pur: default chunk provided to style the div section needed to set the ajax window of results.
Should contain an id: **id="[[+asId]]_reswin"**

Placeholders

These placeholders are related to templates.

[[+asId]]

advSearch identifier
This placeholder is required in the search form template to distinguish advSearch instances.

[[+landing]]

Landing resource document id where are displayed results with non-ajax mode.

[[+method]]

http request method used to send the form.

[[+searchIndex]]

http variable name used for the search string.

[[+searchValue]]

Search string value.

[[+resultsWindow]]

<div></div> section where will be attached the search results window.

[[+helpLink]]

where is displayed the link for opening the advanced search help

AdvSearchHelp

Basic usage: **[[AdvSearchHelp]]**

used to set up the help content
The help is provided by an ajax request to a resource named "AdvSearch Help" which call the AdvSearchHelp snippet.
The content of the window is provided by the lexicon file help.inc Add your own language and customize the content as you want. The default English content presents the **minimal query syntax** supported by the release for **mysql** and **zend** engines.
You could move the resource in the MODx resource tree but don't change his name.

AdvSearch

Basic usage: **[[!AdvSearch]]**

PARAMETERS

&asId

Purpose: Unique id for AdvSearch instance
Option: a-z, _ , 0-9 (case sensitive)
Default: 'as0'
Note: optional. Used to distinguish several AdvSearch add-on instances on the same page. This token is used to link the snippet calls between them. Choose a short name. eg: 'as2'.
Placeholders: -
Related: -

&containerTpl

Pur: the chunk that will be used to wrap all the search results, pagination and message.
Opt: chunk name
Def: 'AdvSearchResults'
Not: optional

&contexts

Pur: The contexts to search.
Opt: comma separated list of contexts
Def: Defaults to the current context if none are explicitly specified.
Not: optional

&currentPageTpl

Pur: The chunk to use for a pagination link..
Opt: chunk name
Def: 'CurrentPageLink'
Not: optional

&fields

Pur: The list of fields available with search results.
Opt: string (csv list of fields)
Def: 'pagetitle,longtitle,alias,description,introtext,content'
Not: optional

&docindexPath

Pur: The path under 'assets/files/' where are located Lucene document indexes.
Opt: string
Def: 'docindex/'
Not: optional

&engine

Pur: Search engine selected
Opt: string ['mysql' | 'zend' | 'all']
Def: 'mysql'
Not: optional

&extractEllipsis

Pur: ellipside to mark the start and the end of an extract when the sentence is cutting
Opt: *[string]*
Def: '…' (three dots)
Not: optional

&extractLength

Pur: length of separate extraction
Opt: 50 < Integer < 800
Def: 200

&extractTpl

Pur: The chunk that will be used to wrap each extract
Opt: chunk name
Def: 'Extract'
Not: optional

&fieldPotency

Pur: potency per field.
Opt: comma separated list of field : potency.
Default potency to 1 if not set
Def: 'createdon'
Not: optional

&highlightClass

Pur: The CSS class name to add to highlighted terms in results.
Opt: css class name
Def: 'advsea-highlight'
Not: optional

&highlightResults

Pur: create links so that search terms will be highlighted when linked page clicked
Opt: 0 (false) | 1 (true)
Def: 1
Require: Highlight plugin
Not: optional

&highlightTag

Pur: The html tag to wrap the highlighted term with in search results.
Opt: html tag name
Def: 'span'
Require: Highlight plugin
Not: optional

&hideContainers

Pur: Search in container resources.
Opt:
- 0 : search in all resources
- 1 : will not search in any resources marked as a container (is_folder).

Def: 0
Not: optional

&hideMenu

Pur: search in hidden documents from menu
Opt:
- 0 : search only in documents visible from menu
- 1 : search only in documents hidden from menu
- 2 : search in hidden or visible documents from menu

Def: 2

Not: optional

&ids

Pur: Comma-delimited list of ids to search in.
Opt: list of ids
Def: ‘’ (empty string)
Not: optional
Use **GetIds** addon to specify complex list of ids.

&includeTVs

Pur: Add TVs values to search results and set them as placeholders.
Opt:

- ‘’ : disabled the feature
- csv list of tv names

Def: ‘’ (empty string)
Not: optional
e.g:
`tv1,tv2,tv3` : added as placeholders

&init

Pur: defines if the search display all the results or none when the page is loaded at the first time
Opt: 'none' | 'all'
Def: ‘none’
Not: optional

&libraryPath

Pur: The path under ‘assets/’ where is located the Zend library.
Opt: path to libraries
Def: ‘libraries/’
Not: optional

&maxWords

Pur: maximum number of words for searching
Opt: int
Def: 20
Not: optional

&method

Pur: whether to send the search over POST or GET.
Opt: ‘POST’ | ‘GET’
Def: ‘GET’
Not: optional

&minChars

Pur: Minimum number of characters to require for a word to be valid for searching.
Opt: 2 < int < 100
Def: 2
Not: optional

&offsetIndex

Pur: The name of the offset parameter that the search will use.
Opt: string
Def: ‘offset’
Not: optional

&output

Pur: output type
Opt: ‘html’ | ‘json’
Def: ‘html’
Not: optional
'json' : Array of all results as json string
'html' : Page of results as html string

&pagingType

Pur: selection of the type of pagination
Opt: 1 | 2
Def: 1
Note: optional
1: Previous - X-Y /Z – Next
2: Results Pages 1 | 2 | 3

&pageTpl

Pur: The chunk to use for a pagination link. Used only by paging0 type.
Opt: chunk name
Def: ‘PageLink’
Not: optional

&paging1Tpl

Pur: The chunk to use for the paging type 1.
Opt: chunk name
Def: ‘Paging1’
Not: optional

&paging2Tpl

Pur: The chunk to use for the paging type 2.
Opt: chunk name
Def: ‘Paging2’
Not: optional

&pagingSeparator

Pur: String used to separate number page links. . Used only by paging0 type.
Opt: string
Def: ‘ | ’
Not: optional

&perPage

Pur: Set to the max number of results you would like on each page. Set to 0 if unlimited.
Opt: integer
Def: 10
Not: optional

&postHook

Pur: A post hook to change the displaying of results.
Opt: A snippet name to run as postHook
Def: ‘’ (empty list)
Not: optional.
e.g: To display the results as a table rather than as a list use the posthook feature.
See the QueryHook documentation, for more details.

&postHookTpls

Pur: A list of chunks used by the postHook to change the display of results
Opt: A csv list of chunks
Def: ‘’ (empty list)
Not: optional.
e.g: To display the results as a table of results, you probably provide: the table header chunk and the row header chunk.
See the QueryHook documentation, for more details.

&queryHook

Pur: A query hook to change the default query
Opt: A snippet name to run as queryHook
Def: ‘’ (empty list)
Not: optional.
See the QueryHook documentation.

&searchString

Pur: to initialize the search with a default searchString. Used in conjunction with &init=`all` this allow to display search results for a specific search string.
Opt: query string
Def: '' – ‘empty string)
Not: optional

&showExtract

Pur: show the search terms highlighted in one or several extract.
Opt: string as nb: csv list of fields
Def: '1:content' - One extract displayed from content field’
Not: optional
5 stands for 5:content
e.g: 3: introtext,content => 3 extracts displayed from introtext . content

&sortby

Pur: comma separated list of couple "field [ASC|DESC]" to sort by. field could be : field name or TV name. Field name of joined resource are prefixed by resource name.
e.g: equipComment_body
Opt: placeholder name
Def: ‘createdon DESC’
Not: optional

&toPlaceholder

Pur: whether to set the output to directly return, or set to a placeholder with this propertys name.
Opt: placeholder name
Def: ‘’ (empty string) => directly returned
Not: optional

&tpl

Pur: The chunk that will be used to display the contents of each search result.
Opt: string
Def: ‘AdvSearchResult’
Not: optional

&urlScheme

Pur: indicates in what format the URL is generated..
Opt: -1, full, abs, http, https
Def: -1
Not: optional
-1 : (default value) URL is relative to site_url

See:
http://rtfm.modx.com/display/revolution20/modX.makeUrl

&withFields

Pur: Define which fields are used for the search in fields of document resource.
Opt:

- ‘’ : disabled the feature
- csv list of fields from modResource

Def:
‘pagetitle,longtitle,alias,description,introtext,content ‘
Not: optional
e.g: `pagetitle,introtext` : the search occurs only inside these 2 fields

&withTvs

Pur: Define which Tvs are used for the search in Tvs
Opt:

- ‘’ : disabled the feature
- csv list of tv names

Def: ‘’ (empty string)

Not: optional
e.g: `tv1,tv2,tv3` : the search occurs only inside these 3 TVs.

Ajax mode Parameters

&effect

Pur: The effect to apply for the displaying of the window of results. See AdvSearch demo site.
Opt: basic | showfade | slidefade
Def: 'basic'
Not: optional

&opacity

Pur: Opacity of the advSearch_output div where are returned the ajax results.
Opt: 0. (transparent) < Float < 1. (opaque)
Def: 1.
Not: optional

&moreResults

Pur: The document id of the page you want the more results link to point to.
Opt: resource id
Def: 0 => no moreResults page

Not: Mandatory if you want more results.

&moreResultsTpl

Pur: The chunk to use for the "More results" link.
Opt: chunk name
Def: 'PageLink'
Not: optional

&withAjax

Pur: whether to set on a new document (page reload), or set the results in div section thru ajax.
Opt: 0 (false) | 1 (true)
Def: 0 => non-ajax mode
Not: Should be set to 1 with ajax mode.

Chunks

AdvSearchResult

Pur: default chunk provided to style a search result.

Should contain:
- **"[[+extracts]]"** to get the extracts of result

AdvSearchResults

Pur: default chunk provided to style a search result.

Should contain:
- **" [[+resultInfo]]"** to get the results info header

- **" [[+paging]]"** to get the pagination
- **" [[+results]]"** to get the search results

Paging1

Pur: default chunk provided to style the pagination type 1

Should contain:
- **"[[+previousLink]]"** to get the link to the previous page
- **"[[+nextLink]]"** to get the link to the next page

Paging2

Pur: default chunk provided to style the pagination type 2.

PageLink

Pur: default chunk provided to style the page number links. Used with paging type 0 only.

CurrentPageLink

Pur: default chunk provided to style the current page number link.
Used with paging type 0 only.

Placeholders

These placeholders are related to templates.

[[+asId]]

AdvSearch identifier
This placeholder is required in the search form template to distinguish advSearch instances.

[[+landing]]

Landing resource document id where are displayed results with non-ajax mode.

[[+resultsWindow]]

<div></div> section where will be attached the search results window.

[[+method]]

http request method used to send the form.

[[+searchIndex]]

Http variable name used for the search string.

[[+etime]]

Server elapsed time of the search.

[[+*fieldName*]]

Any field value from the list of fields provided by the fields parameter.
e.g: &fields=`pagetitle,introtext`
[[+pagetitle]] and [[+introtext]] available

[[+*TVName*]]

Any TV value from the list of TVs provided by the withTVs and includeTVs parameter.
e.g: &withTVs=`tv1,tv2`
[[+tv1]] and [[+tv2]] available

[[+idx]]

Number of result. could be use inside the AdvSearchResult chunk to alternate class.

[[+query]]

The search string used as query.

[[+total]]

The total number of result found.

[[+pagingType]]

Paging type. 1 or 2

[[+page]]

The current page number.

[[+totalPage]]

The total number of result pages.

[[+perPage]]

The maximum number of results par page.

[[+offset]]

Offset of the current page.

[[+first]]

Number of the first result of the current page.

[[+last]]

Number of the last result of the current page.

[[+separator]]

String used as separator between page link number for paging Type2.

Query Syntax

Find below the syntax of AdvSearch to set up powerfull search queries.

Terms
A query is broken up into terms and operators.
There are two types of terms: Single Terms and Phrases.
A Single Term is a single word such as "test" or "hello".
A Phrase is a group of words surrounded by double quotes such as "hello dolly".
Multiple terms can be combined together with Boolean operators to form a more complex query (see below).

Wildcard Searches
AdvSearch supports single and multiple character wildcard searches within single terms(not within phrase queries).
To perform a single character wildcard search use the "?" symbol.
To perform a multiple character wildcard search use the "*" symbol.
The single character wildcard search looks for terms that match that with the single character replaced. For example, to search for "text" or "test" you can use the search:

te?t

Multiple character wildcard searches looks for 0 or more characters. For example, to search for test, tests or tester, you can use the search:

test*

You can also use the wildcard searches in the middle of a term.

te*t

Note: You cannot use a * or ? symbol as the first character of a search.

Boolean Operators

Boolean operators allow terms to be combined through logic operators. AdvSearch supports AND, OR, and NOT as Boolean operators (Note: Boolean operators must be ALL CAPS).

OR

The OR operator is the default conjunction operator. This means that if there is no Boolean operator between two terms, the OR operator is used. The OR operator links two terms and finds a matching document if either of the terms exist in a document. This is equivalent to a union using sets. The symbol **||** can be used in place of the word OR.

To search for documents that contain either "jakarta apache" or just "jakarta" use the query:

"jakarta apache" jakarta

or

"jakarta apache" OR Jakarta

AND

The AND operator matches documents where both terms exist anywhere in the text of a single document. This is equivalent to an intersection using sets. The symbol **&&** can be used in place of the word AND.

To search for documents that contain "jakarta apache" and "Apache Lucene" use the query:

"jakarta apache" AND "Apache Lucene"

NOT

The NOT operator excludes documents that contain the term after NOT. This is equivalent to a difference using sets. The symbol **!** can be used in place of the word NOT.

To search for documents that contain "jakarta apache" but not "Apache Lucene" use the query:

"jakarta apache" NOT "Apache Lucene"

Note: The NOT operator cannot be used with just one term. For example, the following search will return no results:

NOT "jakarta apache"

Grouping

AdvSearch supports using parentheses to group clauses to form sub queries. This can be very useful if you want to control the boolean logic for a query.

To search for either "jakarta" or "apache" and "website" use the query:

(jakarta OR apache) AND website

This eliminates any confusion and makes sure you that website must exist and either term jakarta or apache may exist.

Post Hook

With a post hook you can modify the results before to displayed them !

See the AdvSearch Hooks documentation.
http://www.revo2.wangba.fr/advsearch-documentation.html

Query Hook

Build your faceted search with a query hook !

See the AdvSearch Hooks documentation.
http://www.revo2.wangba.fr/advsearch-documentation.html

Technical requirements

The technical prerequisites for this release of advSearch are the followings:

Revo 2.1 and upper (Php and MySQL versions)
UTF-8 charset
php multi-bytes **ON**
Jquery 1.7.1 (provided with advSearch)
Zend library 1.11 (See installation chapter)

Installation

Whatever the search engine used, you need to install the Zend lucene search classes first**.** This installation doesn't modify your Php environment. This is only a copy of a set of php files. This installation is required **only** for the first installation of AdvSearch.

1/ Go to the zend Framework download page at http://framework.zend.com/download/current/

2/ register you and download the free Zend Framework. Minimal release is enough.

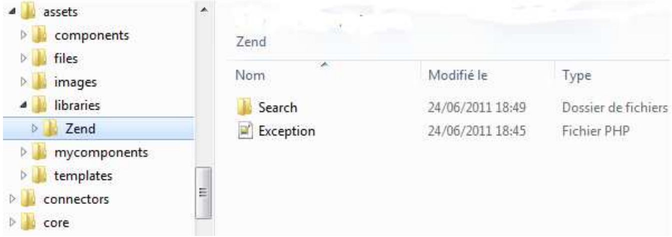
AdvSearch has been tested with Zend Framework 1.11

3/ if you haven't a folder for libraries under assets/ create a folder libraries/ under assets/

4/ unzip you zend package under the assets/libraries/ folder. This should create a subdirectory named 'Zend' (assets/libraries/Zend)

5/ in this directory only the 'search' directory and the 'Exception.php' files are required. You could remove all the others files and directories.

At the end you should have something like this:



- assets
  - components
  - files
  - images
  - libraries
    - Zend
  - mycomponents
  - templates
- connectors
- core

Zend

Nom	Modifié le	Type
Search	24/06/2011 18:49	Dossier de fichiers
Exception	24/06/2011 18:45	Fichier PHP

AdvSearch Glossary

Ajax mode

- Search results are displayed in the current page through AJAX request.

Non ajax mode

- Search results are displayed in a new page (page reloading).

Search engine

- AdvSearch could runs with two different engine: mysql and zend+mysql. Zend engine allows to do a search in an index repository first before to get the results from the mysql database. This needs to index documents first. For performance reasons, this engine should be reserved for search in dynamic contents.

Search in dynamic contents

Mysql database store only static contents. If a document has one or several MODx tags (chunk, tvs, snippet calls …) the parsed results are not stored in the database. To do a search in parsed resources, the parsed documents should be first processed and search terms found in these documents stored in an additional repository (a folder). Then this repository is used in conjunction with MySQL to provide search results.

Search term

- The entry term in the search form

Extract

- Part of a document extracted and added in the results page

Search result

- Title (as a link to the document), description and extract(s)

Search results page

- All the search results with optionaly a more results link to a showMoreResult page.
- or all the search results paginated with the perPage parameter

Highlighted term

- In the search results page, the search terms found could be (or not) highlighted.
- needs the plugin searchHighlight

PostHook

- A snippet which allow to modify on-fly the displayed results

QueryHook

- A snippet which, starting http variables, modify on-fly the default query

Faceted search

- Search with filters based on document fields and Tvs. By using the appropriate search form and queryHook you could set up all kind of faceted search you want. With or without search term input fields.

Search in custom packages

- By default advSearch runs with the modResource package (documents + possibly template variables)
- By using the “Main” and “Joined” declarations of the queryHook, you could add joined resources to the modResource package or replace modResource by an another package (+ possibly joined resources). See the query hook documentation for futher information.

AdvSearch Resources Links

Demo site: http://www.revo.wangba.fr

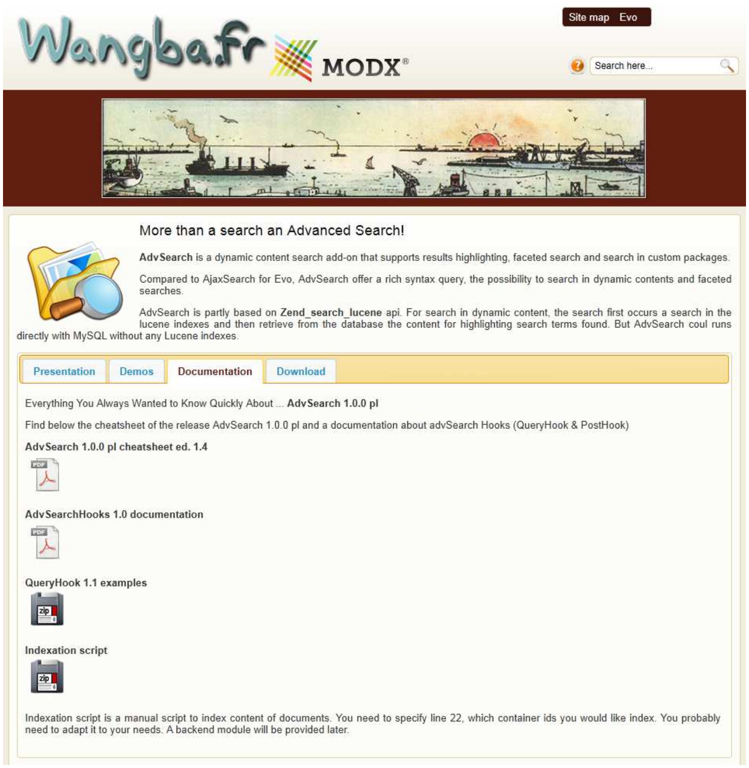
Forum board: http://forums.modx.com/board?board=93

Download: http://modx.com/extras/package/advsearch

Bugs and new feature requests: https://github.com/Coroico/AdvSearch

Wiki page: http://rtfm.modx.com/display/ADDON/AdvSearch

To get the last demos and the updated documentation go to the demo site:
http://www.revo.wangba.fr



More than a search an Advanced Search!

AdvSearch is a dynamic content search add-on that supports results highlighting, faceted search and search in custom packages.

Compared to AjaxSearch for Evo, AdvSearch offer a rich syntax query, the possibility to search in dynamic contents and faceted searches.

AdvSearch is partly based on **Zend_search_lucene** api. For search in dynamic content, the search first occurs a search in the lucene indexes and then retrieve from the database the content for highlighting search terms found. But AdvSearch coul runs directly with MySQL without any Lucene indexes.

Everything You Always Wanted to Know Quickly About ... **AdvSearch 1.0.0 pl**

Find below the cheatsheet of the release AdvSearch 1.0.0 pl and a documentation about advSearch Hooks (QueryHook & PostHook)

AdvSearch 1.0.0 pl cheatsheet ed. 1.4

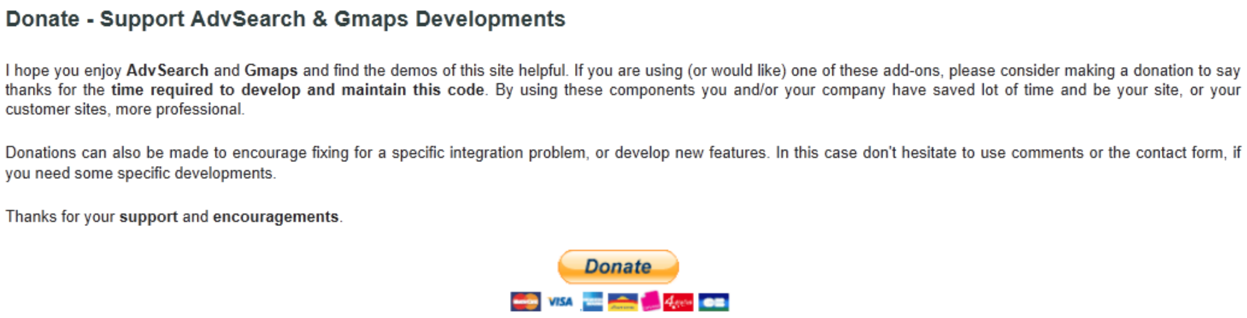
AdvSearchHooks 1.0 documentation

QueryHook 1.1 examples

Indexation script

Indexation script is a manual script to index content of documents. You need to specify line 22, which container ids you would like index. You probably need to adapt it to your needs. A backend module will be provided later.

Thanks to support AdvSearch : http://www.revo.wangba.fr/donate.html



Donate - Support AdvSearch & Gmaps Developments

I hope you enjoy **AdvSearch** and **Gmaps** and find the demos of this site helpful. If you are using (or would like) one of these add-ons, please consider making a donation to say thanks for the **time required to develop and maintain this code**. By using these components you and/or your company have saved lot of time and be your site, or your customer sites, more professional.

Donations can also be made to encourage fixing for a specific integration problem, or develop new features. In this case don't hesitate to use comments or the contact form, if you need some specific developments.

Thanks for your **support** and **encouragements**.