

< XSLTsearch >

VERSION 2.7

REMARKS

Version 0.1. October 2, 2007

CREATOR

Douglas J. Robar, Percipient Studios

LICENSE

CPL License

COPYRIGHT

Copyright © 2006-2008 Percipient Studios. All Rights Reserved.

OVERVIEW

XSLTsearch is the ideal Umbraco search solution for web sites with less than a thousand pages. It is fast, completely self-contained in one file, and extremely configurable.

With *XSLTsearch* you can search for words, phrases, portions of words, and portions of phrases in all document properties and attributes (even custom properties). Search results are automatically sorted by relevance with search terms highlighted in the results. To search for exact phrases, simply enclose the search term in single or double quotes.

XSLTsearch is self-maintaining, you never have to enter a debug mode to create or update an index. *XSLTsearch* works well in hosted and locked-down web environments because it does not require write access to any files on the server, nor will it interfere with an already installed Umbraco/Lucene search engine.

Hidden pages are never shown in search results, and protected pages are only returned if the user has access to the page, ensuring the safety of your private and protected content.

With *XSLTsearch*, you have complete control to customize searching behavior and output formatting (using only macro parameters, in most cases, so you don't need to even look at the XSLT code unless you want to). Semantic markup is used throughout, making it easy to style *XSLTsearch* to match the look and feel of your site. Sample CSS code is provided.

You are free to modify the source code for your umbraco site, but please honor the copyright notice and let me know of any improvements so I can incorporate them for the benefit of others.

Note: XSLTsearch is not meant to be used as the XsltLayout parameter to the Umbraco/Lucene search. XSLTsearch is a complete replacement for the umbraco/Lucene search and does not rely on any umbraco utilities.

DONATIONS

If you find *XSLTsearch* useful, please support its development by sending money via PayPal (www.paypal.com) to doug@percipientstudios.com. Any amount is helpful, from \$5 to \$500; whatever you think *XSLTsearch* is worth to you.

Make a Donation

Thank you!



INSTALLATION

Right-click on the Macros item in the Developer section of umbraco. Select Import Package. Browse the package repository to the XSLTsearch package and click the Download link. Accept the license and click Load Package.

A full umbraco package will be created in the future to simplify installation. Until then, or if you prefer to install *XSLTsearch* manually, follow these instructions.

Upgrading from an Earlier Version of XSLTsearch

Do not attempt to install XSLTsearch twice; the installation will fail. Instead, install the upgrade package for XSLTsearch, which only replaces the XSLTsearch.xslt file itself.

Note: Any modifications you have made to the XSLTsearch.xslt file will be lost by upgrading. Be sure to save a copy of your modified file before upgrading so that you can re-apply your changes to the new file.

CUSTOMIZING XSLTSEARCH

An example of *XSLTsearch*'s output, with default options is shown below.

Search Results

Your search for e matches 8 pages

[About](#)
This is a dummy website created by the umbraco Website Wizard. It's very easy to change the design of this website, most of it can be done just by changing the css. Changing the header image and the footer text Login to umbraco, click on the Frontpa...

[Contact](#)
Lorem ipsum dolor sit amet, consectetur adipiscing elit. In quis arcu vel neque vehicula dapibus. Donec quis est sit amet neque aliquet eleifend. Aenean eu nisl. Donec tincidunt eleifend diam. Maecenas feugiat tincidunt erat. Donec sapien. Quisque blandi...

[Frontpage](#)
Lorem ipsum dolor sit amet, consectetur adipiscing elit. In quis arcu vel neque vehicula dapibus. Donec quis est sit amet neque aliquet eleifend. Aenean eu nisl. Donec tincidunt eleifend diam. Maecenas feugiat tincidunt erat. Donec sapien. Quisque blandi...

[Products](#)
Our famous good are all here for your enjoyment

[Hest](#)

< Previous 1 **2** Next >

An example of *XSLTsearch*'s output, with all options enabled is shown below.

Search Results

Your search for e matches 8 pages
Showing results 1 to 5

- [About](#) (score: 210)
This is a dummy website created by the umbraco Website Wizard. It's very easy to change the design of this website, most of it can be done just by changing the css. Changing the header image and the footer text Login to umbraco, click on the Frontpa...
- [Contact](#) (score: 173)
Lorem ipsum dolor sit amet, consectetur adipiscing elit. In quis arcu vel neque vehicula dapibus. Donec quis est sit amet neque aliquet eleifend. Aenean eu nisl. Donec tincidunt eleifend diam. Maecenas feugiat tincidunt erat. Donec sapien. Quisque blandi...
- [Frontpage](#) (score: 153)
This is the main page for the site
- [Products](#) (score: 10)
Our famous good are all here for your enjoyment
- [Hest](#) (score: 4)

< Previous 1 **2** Next >

Searched 9 pages in 0.016 seconds

XSLTsearch Macro Parameters

source	The parent NodeID within which to look for the search term(s).
Behavior:	The source node and all its descendants are searched.
Default Value:	-1 (the entire site)

searchFields The document properties to search

Behavior: A comma-separated list of fields. The order of the search fields affects the search score and order of the search results! Place the more important fields first, with `bodyText` or its equivalent last. The reason is that if a search term appears in the page's title, there is a greater likelihood that page discusses the search term at length, than it simply being mentioned somewhere in the `bodyText`.

Examples: `@nodeName,bodyText`

Default Value: `@nodeName,metaKeywords,metaDescription,bodyText`

Note: Field names are case-sensitive. Document properties and attributes (preceded by `@`) are supported.

previewFields The document properties to display in the search results

Behavior: A comma-separated list of fields. The order of the preview fields is from most preferred to least preferred. Put the most appropriate fields first (typically, `bodyText`). The first field that contains text will be shown as the preview information for each search result.

Default Value: `bodyText,metaDescription`

Note: Field names are case-sensitive. Only document properties are supported (not attributes). Typically, `previewFields` will contain the same field list as `searchFields`, but in the reverse order, and without any `@` properties.

resultsPerPage The number of search results to display on each page of search results.

Default Value: 5

previewChars The maximum number of search characters to display in the preview for each search result.

Default Value: 255

previewType The format of the preview information to display.

Behavior: The preview text for each search result includes hit highlighting for search terms. You can select either a conventional display with the first words of the document shown (which may or may not contain the search term(s)), or you can display the context of as many hits as fits within the `previewChars` character count.

Example: **BEGINNING**
1. [About](#)
This is a dummy website created by the umbraco Website Wizard. It's very easy to **change** the design of this website, most of it can be done just by changing the css. Changing the header image and the footer text Login to umbraco, click on the Frontpa...

CONTEXT
1. [About](#)
Context: ... It's very easy to **change** the design of this website, ... and 103px heigh. To **change** the footer, all you ... need to do is **change** the text in the ... on textpages You can also **change** the small text info ... be language versioned. To **change** the date formatting and ...

Options: **BEGINNING**
CONTEXT

Default Value: **BEGINNING**

Note: If you select **CONTEXT**, you set the `previewFields` parameter to be the same as `searchFields` (without any `@` attributes). Otherwise, some of your searches may have no context to display.

searchBoxLocation Place a search box and button above, below, or both above and below the search results.

Options: **BOTTOM**
 TOP
 BOTH
 NONE

Default Value: **BOTTOM**

Note: If **BOTH** is set, only one search box will appear if nothing has been searched yet.

showPageRange Show the summary text at the top of the search results page, "Showing results X of Y"

Behavior: This can be useful when many results are returned, especially if **showOrdinals** is set to false.

Default Value: **0** **(FALSE)**

showOrdinals Show the number before each search result, such as 1. 2. 3. 4. etc.

Default Value: **0** **(FALSE)**

showScores Show the total score calculated for the search term within each document.

Default Value: **0** **(FALSE)**

showStats Show the summary text at the bottom of the search results page, "Search X pages in Y seconds."

Default Value: **0** **(FALSE)**

showDebug Show the version of XSLTsearch

Behavior: Set automatically when **?umbDebugShowTrace=true** is appended to the URL's querystring. The output will include the version of XSLTsearch along with the typical umbraco debugging output.

Note: This parameter is not, strictly speaking, a macro parameter since its behavior is independent of the macro properties.

SAMPLE CSS

Ample markup is provided by *XSLTsearch* to allow you to customize the look to match your site's design, while also working well in a CSS-disabled environment. Below is a sample CSS definition for *XSLTsearch* to use as a starting point for your own CSS.

```
/* XSLTsearch 2.x */
.xsltsearch_result p          {margin: 0; padding: 0; }
.xsltsearch_result:hover     {background: #f6f6f6;}
.xsltsearch_ordinal          {font-weight: bold; font-size: 75%;}
.xsltsearch_title            {font-weight: bold;}
.xsltsearch_score            {font-size: 75%;}
p.xsltsearch_result_description {padding-bottom: 10px;}
.xsltsearch_description strong {background: #ff9;} /* search term highlighting */
#xsltsearch_navigation       {text-align: center;}
#xsltsearch_navigation.disabled {color: #999;}
#xsltsearch_stats            {font-size: 75%; color: #666;}
```

TROUBLESHOOTING

If, after searching for a word and clicking the Next link, you cannot search for a new word but always get the same results back no matter what you enter, you have an `<?ASPNET_FORM>` tag surrounding the *XSLTsearch* macro. Simply turn off (close) the tag before the *XSLTsearch* macro, and then turn it on (open it) again immediately after the macro. Here is an example (note that umbraco v2.x does not need the `disablescriptmanager="true"` portion of the tag; that is for Ajax support within umbraco v3).

```
<h1><?UMBRACO_GETITEM field="pageName"/></h1>
</?ASPNET_FORM>
  <?UMBRACO_MACRO
    macroAlias="XSLTsearch"
    ...
  >
</?UMBRACO_MACRO>
<?ASPNET_FORM disablescriptmanager="true">
<?UMBRACO_GETITEM field="bodyText"/>
```

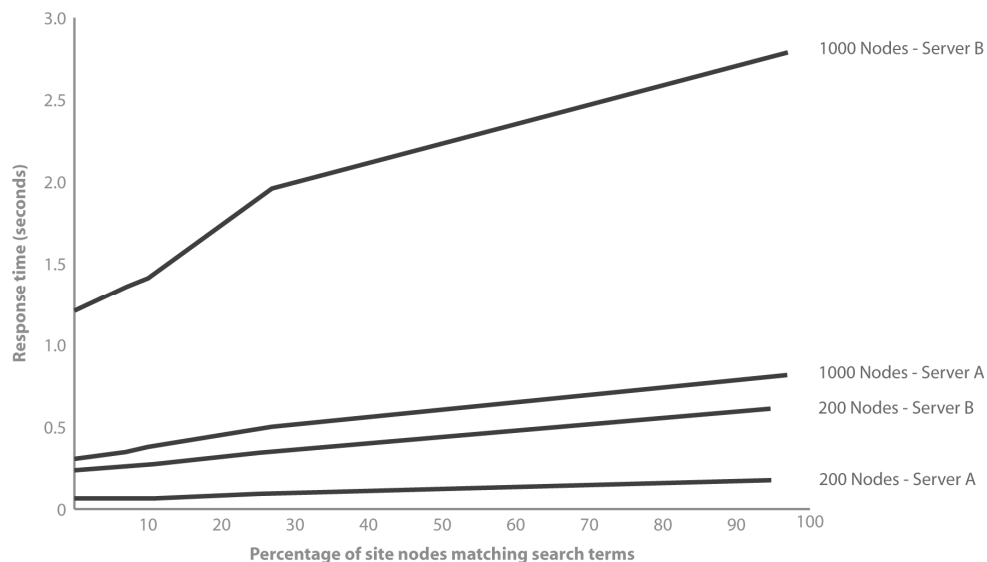
PERFORMANCE

XSLTsearch 1.3 made tremendous speed improvements over earlier versions (20X faster), making it an appropriate search engine for sites of up to a thousand nodes. *XSLTsearch 2.0* capitalizes on the advances in umbraco 3.0 to make further performance gains, especially when many pages are returned by a search. The combination of *XSLTsearch 2.0* and umbraco 3.0 will return search results nearly four times faster than before, while also adding significant search capabilities. You will find that *XSLTsearch* performs well even with modest web server hardware, and is very responsive on more powerful hardware, as the charts below show.

Search Response Time Statistics

In order to give a sense of the responsiveness of *XSLTsearch* with various sized sites, baseline tests were run on sample sites with approximately 200 and 1000 pages on two different servers, and with various versions of *XSLTsearch* and umbraco. The results are shown below. The fastest performance is achieved with *XSLTsearch 2.0* and umbraco 3.0.

TYPICAL SEARCH RESPONSE TIMES WITH XSLTSEARCH 2.0 WITH UMBRACO V3.0 (RC2)



Server B is an old, multi-processor server (700MHz) and shows its limited power with large sites, though it performs admirably on smaller sites. Server A is a newer, single-core machine that scales easily to a thousand node site, with response times typically less than half a second.

Site Size	Search Hits	Hit Percentage (of total pages)	Response Time Web Server A ¹	Response Time Web Server B ²
202 pages	0	0%	0.05 sec.	0.22 sec.
	1	0.5%	0.06 sec.	0.23 sec.
	23	11%	0.06 sec.	0.26 sec.
	50	25%	0.08 sec.	0.33 sec.
	193	95%	0.16 sec.	0.60 sec.
1,005 pages	0	0%	0.29 sec.	1.25 sec.
	1	0.1%	0.30 sec.	1.21 sec.
	72	7.1%	0.34 sec.	1.35 sec.
	102	10%	0.36 sec.	1.40 sec.
	269	27%	0.49 sec.	1.95 sec.
	975	97%	0.80 sec.	2.78 sec.

When reviewing the search response times, remember that while a search that returns a large percentage of the total pages in the site may take considerable time, such a search is not likely to be very helpful to the site visitor (it returns too many results) and should rarely occur in real-world usage. For instance, searching for the single letter 'a' will typically return a large number of results, though knowing all the pages with the letter 'a' in them is not very helpful. Rather, searches for complete words much more typical in actual practice and not only returns fewer (and more meaningful) results, but also returns those results more quickly. Therefore, though the worst-case search response time may be quite high, as shown in the response time tables, a typical real-world search that returns less than 15% of the site's pages would be typical of *XSLTsearch* responsiveness.

How to Improve Search Performance

If you find the performance of *XSLTsearch* sluggish in your environment, you can improve performance in various ways. Obviously, faster servers will help. But you can improve the performance of *XSLTsearch* with your existing hardware as well. Because the responsiveness of *XSLTsearch* is largely determined by the number of nodes being searched, searching only a portion of your site will greatly improve performance. Consider creating multiple search pages for different purposes or sections of your site. Each would include the *XSLTsearch* macro, but with differing `source` and `searchFields` parameters to create specialized search pages for the entire site, only blog entries, photos, or technical support documents, for instance. Also, searching only within the most important document type properties (via the `searchFields` macro parameter) rather than every possible property will improve search speed. The fewer `searchFields`, the faster *XSLTsearch* will respond.

¹ Dell Precision 470 – (1) 3.20GHz Pentium IV Xeon, 2GB RAM, Windows Vista Business, IIS7

² Dell PowerEdge 6400 – (4) 700MHz Pentium III Xeon processors, 2GB RAM, Windows Server 2003 (SP1), IIS6

Search Response Comparisons

For comparison, here are response times for *XSLTsearch 2.0* and 1.3 with umbraco 2.1.x. For those wishing to run umbraco 2.1.x, you will find that *XSLTsearch 2.0* is a slightly slower than *XSLTsearch 1.3*, due to its increased functionality.

TYPICAL SEARCH RESPONSE TIMES WITH XSLTSEARCH 2.0 WITH UMBRACO V2.1.6

Site Size	Search Hits	Hit Percentage (of total pages)	Response Time Web Server A ³	Response Time Web Server B ⁴
202 pages	0	0%	0.07 sec.	0.26 sec.
	1	0.5%	0.09 sec.	0.27 sec.
	23	11%	0.16 sec.	0.56 sec.
	50	25%	0.25 sec.	0.87 sec.
	193	95%	0.68 sec.	2.48 sec.
1,005 pages	0	0%	0.38 sec.	1.43 sec.
	1	0.1%	0.39 sec.	1.44 sec.
	72	7.1%	0.60 sec.	2.29 sec.
	102	10%	0.71 sec.	2.64 sec.
	269	27%	1.25 sec.	4.79 sec.
	975	97%	3.17 sec.	12.03 sec.

TYPICAL SEARCH RESPONSE TIMES WITH XSLTSEARCH 1.3 WITH UMBRACO V2.1.X

Site Size	Search Hits	Hit Percentage (of total pages)	Response Time Web Server A ⁵	Response Time Web Server B ⁶
220 pages	0	0%	0.08 sec.	0.27 sec.
	1	0.5%	0.09 sec.	0.31 sec.
	29	13%	0.16 sec.	0.45 sec.
	77	35%	0.27 sec.	0.72 sec.
	210	95%	0.59 sec.	1.56 sec.
1,000 pages	0	0%	0.38 sec.	1.33 sec.
	1	0.1%	0.36 sec.	1.40 sec.
	64	6.4%	0.51 sec.	1.69 sec.
	126	12%	0.60 sec.	2.10 sec.
	349	35%	1.08 sec.	3.39 sec.
	938	94%	2.48 sec.	7.87 sec.

³ Dell Precision 470 – (1) 3.20GHz Pentium IV Xeon, 2GB RAM, Windows Vista Business, IIS7

⁴ Dell PowerEdge 6400 – (4) 700MHz Pentium III Xeon processors, 2GB RAM, Windows Server 2003 (SP1), IIS6

⁵ Dell Precision 470 – (1) 3.20GHz Pentium IV Xeon, 1GB RAM, Windows XP(SP2) Pro, IIS5

⁶ Dell PowerEdge 6400 – (4) 700MHz Pentium III Xeon processors, 2GB RAM, Windows Server 2003 (SP1), IIS6

CHANGE HISTORY

Version	Date	Comments
2.7	January 11, 2008	Bug fix. <ul style="list-style-type: none"> searchBoxLocation="top" now shows the search box even if nothing has been searched yet... reliably
2.6	October 2, 2007	Changed the default value of the "source" parameter to search the entire site if the source parameter's value is not specified. First release for the umbraco Package Repository.
2.5	September 11, 2007	Bug fix and improved error checking. <ul style="list-style-type: none"> searchBoxLocation="top" now shows the search box even if nothing has been searched yet Only nodes with an @id are searched Added NONE option to searchBoxLocation parameter
2.4	August 18, 2007	Bug fix. <ul style="list-style-type: none"> Now returns protected pages, but only if the user has access to the page. Previously, protected pages were never returned
2.3	August 9, 2007	Bug fixes. <ul style="list-style-type: none"> Highlighted terms in search results now retain their case properly. Corrected layout bug that appeared in umbraco v3 when no results were returned
2.2	June 21, 2007	Improved error handling with CONTEXT searches when searchFields and displayFields parameters do not coincide properly.
2.1	June 13, 2007	Important bug fix. <ul style="list-style-type: none"> Boolean macro parameters should be specified as '1' or '0' rather than 'true' or 'false', in keeping with the way umbraco v3 creates Boolean values when inserting macros.
2.0	May 30, 2007	Major update. <ul style="list-style-type: none"> Supports umbraco v2.1 through v3. Documentation completed. Added phrase searching when enclosed in single or double quotes (default is 'and' searching of all terms). All configuration parameters given defaults and passed in via macro Output is semantic XHTML for easy CSS styling
1.4	March 2007	JavaScript functions converted to C# for umbraco v3. Not released.

1.3 beta 5	October 20, 2006	<p>Major update.</p> <ul style="list-style-type: none"> • Now more than 20X faster than pervious versions! • Numerous configuration options added or expanded. • Improved results paging for large sites. • Extensive commenting of XSLT code. • Can also search within document attributes (such as @nodeName) • Added displayFields for easy customization of output (defaults to bodyText (if present) then resorts to description (if present)) • Added simple boolean search (all search words are now searched using AND rather than as an implicit exact phrase). This is the first step to providing advanced search syntax capabilities. • Installation and usage documentation provided (not yet completed).
1.2	September 9, 2006	Copyright and terms of use notice added. Not released.
1.1	January 23, 2006	Minor update. Not released.
1.0	January 24, 2006	Initial release.