

User's Guide

User's Guide

Together Teamlösungen EDV-Dienstleistungen GmbH

Elmargasse 2-4

A-1190

Vienna

Austria

+43 (0) 5 04 04 - 122

+43 (0) 5 04 04 - 11 122

<office@together.at>

<http://www.together.at/together/index.html>

Copyright © 2006 Together Teamlösungen EDV-Dienstleistungen GmbH

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior written permission of the Together Teamlösungen EDV-Dienstleistungen GmbH.

Together Teamlösungen EDV-Dienstleistungen GmbH DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Table of Contents

1. About 'Together Search Server' and 'Together Document Viewer'	1
Fast execution	1
Web application	1
Flexible	1
2. 'Together Search Server' Administration/Indexing	2
Login	2
Sites	2
Site List Management	3
Editing a site	5
Site Attributes	5
Filtering files	6
Include list	7
Metadata	7
Site Indexing	8
Index all	8
Edit a Site	8
Indexing Status	8
Indexing History	9
Not indexed files	9
3. 'Together Search Server' Searching	11
Searching	11
Basic Search	11
Advanced Search	13
E-Mail Search	15
Search Results	16
XSL Transformation	17
File download	17
Presentation Object Search	18
tss.xsd	19
Preview link	21
4. 'Together Document Viewer'	22
About	22
Presentation Object Preview	22
Parameters	22
Preview.xsd	22

List of Figures

2.1. 'Together Search Server' Administrative Main page	2
2.2. Sites management main page	3
2.3. List of sites with available operations	4
2.4. Site page	5
2.5. Indexing Status	8
2.6. Sites indexing history	9
2.7. Not Indexed Files	10
3.1. Basic search	11
3.2. Advanced Search	13
3.3. Advanced Search	13
3.4. E-Mail Search	15
3.5. Search Result Page	17
3.6. File download dialog	18

List of Tables

2.1. Buttons in Site List Management4

3.1. Examples of search over text field 12

3.2. Examples of search over Date field 14

3.3. Examples of search over 'Document Properties' or 'Meta Data' field 15

Chapter 1. About 'Together Search Server' and 'Together Document Viewer'

Fast execution

'Together Search Server' and 'Together Document Viewer' are a fast and lightweight indexing and searching applications designed using 'Together application server' - platform built to handle extreme high volume web sites in an n-tier architecture.

Web application

'Together Search Server' and 'Together Document Viewer' are a web applications that can run on virtually any existing web server. The whole applications runs and is managed through a web browser which limits the system requirements (resources) need to run it on the client side. The installation and configuration on the server side as very simple as well and the whole process takes less than 5 minutes.

Flexible

'Together Search Server' and 'Together Document Viewer' are written in pure JAVA which guarantees its execution on virtually any OS currently in use. Thanks to its modular design, 'Together Search Server' and 'Together Document Viewer' can be easily extended/upgraded with new features, search and indexing engines etc.

Chapter 2. 'Together Search Server' Administration/Indexing

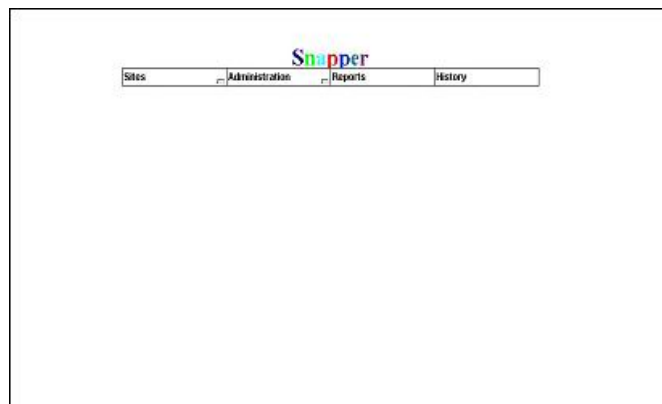
Login

Accessing the administration for the first time displays the logging dialog. By default the username is “admin” and password is “enhydra”.

Administrative part of the application consists of several modules:

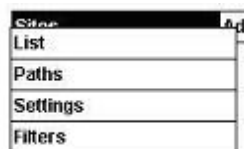
- *SITES* - site management, site path management, site settings, indexing.
- *ADMINISTRATION* - application settings, indexing status, document group, file type mapping
- *REPORTS*
- *HISTORY* – indexing history

Figure 2.1. 'Together Search Server' Administrative Main page



Sites

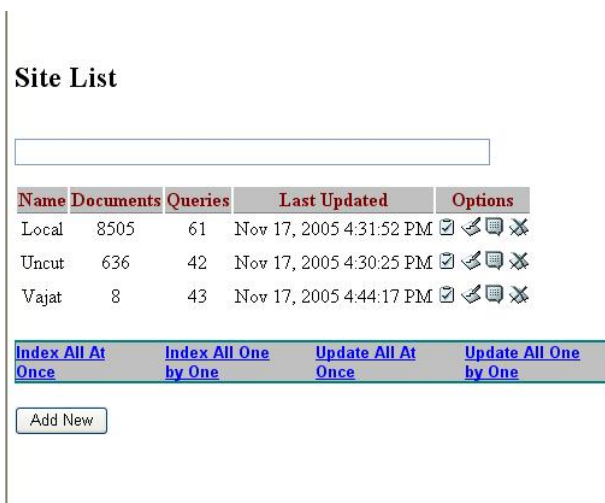
Clicking the SITES button on the Administrative main page loads the sites module and displays it in a new web page. Sites operations are listed in web pages middle frame (Sites Bar).



The following functionality is available in this module:

- *LIST* - Site Indexing and editing.
- *PATHS* - Editing of paths and path types associated with each site
- *SETTINGS* - Various site setting
- *FILTERS* - Indexing filters management

Figure 2.2. Sites management main page



By default, clicking the SITES button, displays the LIST (Site List) screen in the bottom frame.










Site List Management

Selecting the LIST button in the Site bar displays the Site List page (module) in the bottom frame. Initially, the list of sites is displayed. The following Site management operations are available on this page:

- *list of sites*
- *index*

- *index all*

Figure 2.3. List of sites with available operations



Site List				
<input type="text"/>				
Name	Documents	Queries	Last Updated	Options
Local	8505	61	Nov 17, 2005 4:31:52 PM	  
Uncut	636	42	Nov 17, 2005 4:30:25 PM	  
Vajat	8	43	Nov 17, 2005 4:44:17 PM	  
Index All At Once		Index All One by One	Update All At Once	Update All One by One
<input type="button" value="Add New"/>				

The following information can be obtained from the List of Sites:

- *Name* - document name
- *Documents* - number of documents indexed in the site
- *Queries* - number of queries (searches) made to this site
- *Last Updated* - the time of last indexing of site index
- *Options* - site management operations

Operations that can be performed over an existing site are displayed in the Options column. Each operation is represented by a button:

Table 2.1. Buttons in Site List Management

	<i>Index</i> , site indexing
	<i>Edit/View</i> site settings

Clicking the Index All button indexes all Sites. There is a choice between indexing all Sites at once, or one-by-one (after one is finished the next one is started).

Editing a site

Figure 2.4. Site page

Create New Site

Name

Language

DE

Max. Size (kB)

Max. Age (days)

Searched by default

☒

Downloadable

☒

Index directory

Use Filter list

☒

Database

Table

Column

Use Include list

☐

Include Database

Include Table

Include Column

Include Column Modified

Metadata Database

Metadata Table

Metadata File Column

Metadata Key Column

Metadata Value Column

Index content

☒

Index directories

☐

Index unknown file types

☐

Index by group

Select All

☐

Microsoft Word (* .doc)

☐

Microsoft Excel (* .xls)

☐

Microsoft Power Point (* .ppt)

☐

Microsoft Power Show (* .pps)

☐

Plain text (* .txt)

☐

Rich text documents (* .rtf)

☐

Java source file (* .java)

☐

Create Site

Site Attributes

The following attributes are listed on the Site page :

- *Name* – Site name. Must be unique (name per site).
- *Language* – Site language. Used for analyzing and indexing documents
- *Max. Size (kb)* – Maximal size of a file to be indexed in thousands of bytes. If a file is larger than the value specified in this field, it won't be indexed.
- *Max. Age (days)* - Maximal age of a file to be indexed in days. If a file is older than the value specified in this field, it won't be indexed.

The rest fields are not mandatory.

- Search by default – checked automatically includes the site in search
- Downloadable – checked allows for downloading of sites files
- Index directory – overrides the default IndexDir parameter
- Filter database – database where filtered files are stored
- Filter table - table where filtered files are stored
- Filter column - column where filtered files are stored
- Include database – database where included files are stored
- Include table - table where included files are stored
- Include column - column where included files are stored
- Include column modified – document modification time column
- Index content – should file content be indexed or just index file name and modification time.
- Index directories – directories will be indexed.
- *Index unknown file types* – all other file types. These files are not parsed for contents but rather indexed only using their file name and modification time.
- Index by group – Choose predefined file group

Filtering files

If the optional attributes Filter database, Filter table, and Filter column were entered for a Site, during indexing files to be indexed will be filtered according to the criteria (file paths) entered in the Filter column.

Filter database is a logical database name. Filter table is a table within the database. Filter column is a

column within the table. This is the column where the actual files (file paths) to be filtered are stored.

In order to use the database, a user/administrator should configure ("filterDB") in

```
<together-application-server-runtime-7.0 root>/multiserver/webapps/tssAdmin/WEB-INF/ web.xml
```

```
<together-application-server-runtime-7.0 root>/multiserver/conf/catalina/localhost/tssAdmin.xml
```

Include list

If the optional attributes Include database, Include table, Include Column and Include Column Modified were entered for a Site, during indexing files to be indexed will be indexed according to an include list in an external database.

Include database is a logical database name. *Include table* is a table within the database. *Include column* is a column within the table containing full file path. This is the column where the actual files (file paths) to be indexed are stored. *Metadata key column* is a column within the table where the file property keys are stored. *Metadata value column* is a column within the table where the file property values are stored.

Metadata

If the optional attributes Metadata database, Metadata table, Metadata File Column, Metadata Key Column and Metadata Value Column were entered for a Site, during indexing and updating files addition information about each file whose reference exists in Metadata database will be indexed, and at the end of the index/update process if the predefined file type 'NULL' is selected to be indexed, Meta Data addition information will be indexed in case that file path of the file from 'Metadata File Column' don't exists on file system and not have been indexed/updated yet.

If the parameter LogicalNameFromDatabase was set to "1" in the application configuration file, the indexing/update mechanism will try to locate the DocumentLogicalName parameter (also defined as a string value in the configuration file) in the metadata's key column for the file being indexed. If found the document title is obtained from metadata's value column corresponding to that key. This value will be then indexed along with other attributes.

If the LogicalNameFromDatabase parameter was set to "0" or no DocumentLogicalName was found among keys in the metadata database, the default mechanism for extracting document title is used.

If parameter 'DocumentUpdate' in the application configuration file was found among keys in the metadata database, modification time is obtained from metadata's value column corresponding to that key. If the key is not found current time will be used as modification time.

Parameter 'DocumentUpdatePattern' in the application configuration file is used to describe the pattern of the modification time in the metadata's value column.

Metadata database is a logical database name. *Metadata table* is a table within the database. *Metadata file column* is a column within the table. This is the column where the actual files (file paths) to be indexed are stored.

Site Indexing

Two mode of indexing are present. Normal and Re-Index mode.

In Normal mode every click on index button will start index process from the beginning.

The main functionality of Re-Index mode is to continue with index process from last indexed file. According to this, index process will create 'txt' file ('include.txt' or 'path.txt', that is placed in index directory of the 'Site', together with index files) where will be logged last indexed 'directory/file'.

After index process is finished, the 'txt' file will be deleted. According to this new index process will be started from beginning in Re-Index Mode.

Mode of Index, that application will start is defined in 'web.xml' file. Parameter is 'Indexer.ReIndexMode' and default value is 'true'.

Clicking the Indexing button



Index all

Clicking the *Index All* button indexes all Sites. There is a choice between indexing all Sites at once, or one-by-one (after one is finished the next one is started).

Edit a Site

Clicking the Edit Site button



on the Site List screen displays the Site page. This page resembles the Site page. All attributes are explained in the Create Site section.

Indexing Status

Selecting the Status option on the Administration menu displays the active/started indexing processes. The screen looks like this:

Figure 2.5. Indexing Status

Sites

Administration
Status
Document groups
File types

Reports

History

Active Indexing Processes

Name	Options
CDrive	

Indexing History

Clicking the History menu displays the history of sites indexed with the following information:

Site – site name

Document No. – number of documents in the index

Type – operation type: indexing

Started – operation start time

Finished – operation finish time

Length – length of the operation

Figure 2.6. Sites indexing history

Indexed sites history

Site	Document No.	Type	Started	Finished	Length
tomcat-docs	372	INDEX	Jun 19, 2006 11:24:42 AM	Jun 19, 2006 11:24:55 AM	13 sec.

[Clear](#)

Clicking the Clear button clears the history.

Clicking the Site name displays the page showing all not indexed files during the operation (if any).

Not indexed files

In case a file was not successfully indexed, it appears on this page displaying its full path with the possibility to download the file.

Figure 2.7. Not Indexed Files

Not indexed files for site: **tomcat-docs**



Chapter 3. 'Together Search Server' Searching

Searching

Search part of the application consists of a search module that enables:

- *Basic search*
- *Advanced search*
- *File download*

Basic Search

Clicking the *SEARCH* button on the main page loads the sites basic search and displays it in a new web page:

Figure 3.1. Basic search



Checking the checkboxes in the top of the page selects sites to be searched. Basic search is search over content of the document. According to this, query examples are listed in the table.

Table 3.1. Examples of search over text field

Query expression	Matches document that
search	Contain the term search
search enhydra	Contain the term search or enhydra, or both
search OR enhydra	Contain the term search or enhydra, or both
"search application" OR enhydra	Contain the fraze "search application" or just enhydra, or both
(search OR application) AND enhydra	Contain the term either search OR application AND enhydra.
"search application" NOT enhydra	"search application" NOT enhydra
+search +enhydra	Contain search but not contain enhydra
search AND enhydra	Contain both search and enhydra
search*	Contain terms that begin with search Note: You cannot use a * symbol as the first character of a search
search~	Contain terms that are close to the word search
sna*per	Contain terms that begin with "sna" and ends with "per"
snapp?r	Contain terms that are close to the word snapp?r Note:

	You cannot use a ? symbol as the first character of a search.
--	---

Clicking the *Search* button performs the search.

Advanced Search

Clicking the *Advanced Search* button on the basic search page loads the sites advanced search and displays it in a new web page:

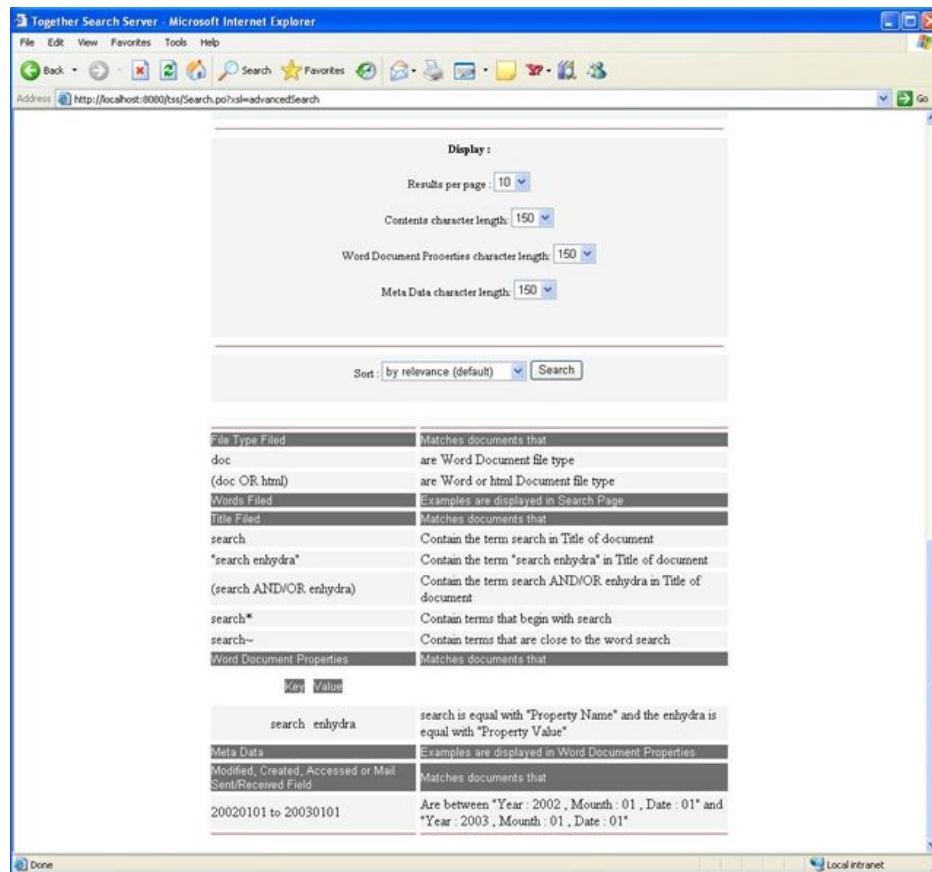
Figure 3.2. Advanced Search

The screenshot shows a web browser window titled 'Together Search Server - Microsoft Internet Explorer'. The address bar displays 'http://localhost:8080/tss/Search.do?vs=advancedSearch'. The page features the 'Together Search Server' logo and a search bar. Below the search bar, there are several sections for advanced search criteria:

- File Type:** A text input field.
- Search for:** A dropdown menu.
- Words:** A text input field.
- Title:** A text input field.
- Modified from:** A text input field followed by 'to:' and another text input field.
- OS Specific:**
 - Created from:** A text input field followed by 'to:' and another text input field.
 - Accessed from:** A text input field followed by 'to:' and another text input field.
 - Owner:** A text input field.
- Compound files:**
 - Author:** A text input field.
 - Last saved by:** A text input field.
- Document Properties:**
 - Key:** A text input field followed by 'Value:' and another text input field.

The status bar at the bottom shows 'Done' and 'Local intranet'.

Figure 3.3. Advanced Search



Checking the checkboxes in the top of the page selects sites to be searched.

Advanced search enables more complex and detailed searches like: searching over File Type, Words (contents), title, document properties, Meta Data and over other file metadata.

Available fields (document specific metadata) over whom search can be obtained are classified by field type:

'File Type', 'Words', 'Title', 'Author', 'Last saved by' are text fields, examples of query expressions over them are listed in the 'Examples of search over text field' table.

'Modified', 'Created', 'Accessed' or 'Mail Sent/Received' are date fields, examples of query expressions over them are listed in the 'Examples of search over Date field' table.

'Document Properties' and 'Meta Data' are specific text fields, consist of collection pair key/value. Examples of query expressions are listed in the 'Examples of search over 'Document Properties' or 'Meta Data' field' table.

Table 3.2. Examples of search over Date field

Modified, Created, Accessed or Mail Sent/Received Field	Matches documents that
From: 20020101 To: 20030101	Are between "Year : 2002 , Mounth : 01 , Date : 01" and "Year : 2003 , Mounth : 01 , Date : 01"
From: 20020101 To: =	Are sent/received "Year : 2002 , Mounth : 01 , Date : 01"

From: 20020101 To:	Are between "Year : 2002 , Mounth : 01 , Date : 01" and current date
From: To: 20020101	Are between "Year : 1970 , Mounth : 01 , Date : 01" and "Year : 2002 , Mounth : 01 , Date : 01"

Table 3.3. Examples of search over 'Document Properties' or 'Meta Data' field

Modified, Created, Accessed or Mail Sent/Received Field	Matches documents that
Key: search Value: enhydra	'search' is equal with "Property Name" and the 'enhydra' is equal with "Property Value". The query (frazе) will be : '("search = enhydra")'

E-Mail Search

Clicking the *Mail Search* button on the basic search page loads the sites E-Mail search and displays it in a new web page:

Figure 3.4. E-Mail Search

Content, Subject, From, To, CC, BCC Filed	Examples are displayed in Search Page
Mail Sent/Received Field (depends witch radio button is selected)	Matches e-mails that
From: 20020101 To: 20030101	Are between "Year: 2002, Mounth: 01, Date: 01" and "Year: 2003, Mounth: 01, Date: 01"
From: 20020101 To: =	Are sent/received "Year: 2002, Mounth: 01, Date: 01"
From: 20020101 To:	Are between "Year: 2002, Mounth: 01, Date: 01" and current date
From: To: 20020101	Are between "Year: 1970, Mounth: 01, Date: 01" and "Year: 2002, Mounth: 01, Date: 01"

Checking the checkboxes in the top of the page selects sites to be searched.

E-Mail search enables more complex and detailed searches through E-mail metadata like: date of sent/received e-mail, from/to/cc/bcc, subject of the e-mail and finally through e-mail content.

Available fields (e-mail specific metadata) over whom search can be obtained are clasified by field type:

'Content ', 'Subject ', 'From', 'To', 'CC' and 'BCC' are text fields, examples of query expresions over them are listed in the 'Examples of search over text field' table.

Depends which radio button is selected search can be obtained over 'Mail Sent/Received' date fields or will be ignored if selected 'Unknown', examples of query expresions over them are listed in the 'Examples of search over Date field' table.

Search Results

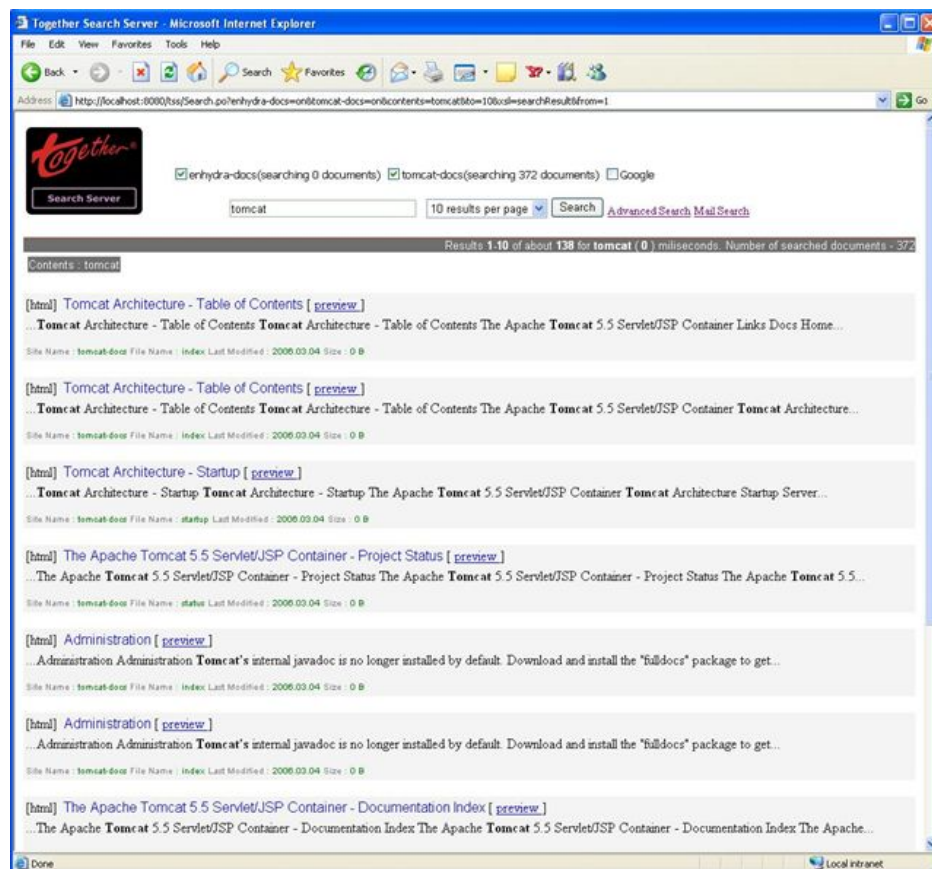
Search results representation is a standardized web page output just like any other search engine (i.e. Google). The main difference between 'Together Search Server' and other search engines is that it includes XSL transformations that are defined in an .xsl file. This way a system administrator can style the search results output in accordance with system needs/wishes.

XSL Transformation

File that needs to be edited in order to change the search output webpage in runtime is WEB_INF\classes\xls\searchResults.xls, WEB_INF\classes\xls\search.xls and WEB_INF\classes\xls\advancedSearch. For example, editing the “font” parameter and changing the font size, changes the font size (and the look of the page) as soon as the next search on the search page is performed.

The standard results page:

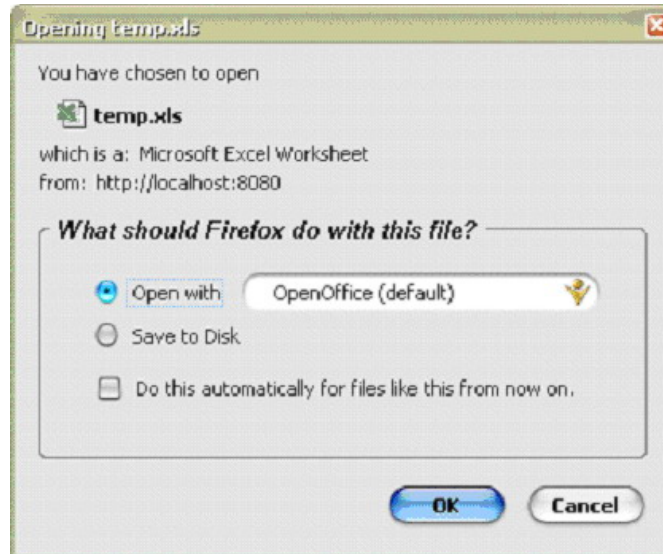
Figure 3.5. Search Result Page



File download

Clicking the file (link) in the *Search Results* page opens the selected file. This is done only if the *Download* parameter in the application configuration file is set to “ON”, and if the file path was correctly mapped while creating a new path.

Figure 3.6. File download dialog



Presentation Object Search

'Together Search Server' application is made up of one presentation object. Name of this presentation object is Search.po. It is accessed by pointing to the following address: <http://server/tss/Search.po>. [<http://server/snapper/Search.po>]

This po object uses request parameters to form an XML file defined by a search.dtd. After the XML is created the po object uses an xsl file defined in xsl request parameter (default is 'search.xsl') to transform the xml file and display the results.

NOTE : If value of xsl request parameter is 'none' the result will be XML file (XML file will not be transformed).

Request parameters:

- *xsl* – The xsl file that will be used to transform xml file. If this parameter is not defined the default xsl will be used (search.xsl)
- *from, to* – These parameters defines documents (from-to) that will be displayed in search result page.
 - If these parameters are not defined the default search result will be used ('from'=1, 'to'=10).
 - if parameter 'to' is not defined and parameter 'from' is, search result will be ('from'=from, 'to'=from+10)
 - if parameter 'from' is not defined and parameter 'to' is, search result will be ('from'=to- 10, 'to'=to)
- These parameters are compared to configuration parameter MaxRange (defined in web.xml file) to

restrict range of search result

- *type* – The type of document to search
- *contents* – The term to search in content of document
- *title* - The term to search in title of document
- *modifiedfrom* , *modifiedto* - The range of document modification time to be searched
- *propertiesKey* , *propertiesValue* – Search in Word Document properties field for key = value
- *metaDataKey* , *metaDataValue* – Search in Meta Data field for key = value
- *contentLength* - Content length in characters to be displayed in search result
- *metaDataLength* - MetaData length in characters to be displayed in search result
- *wordLength* - Word Document length in characters to be displayed in search result
- *sortType* - sorting by :relevance(by default) , newest (by newest modified files) and oldest (by oldest modified files)
- *searchDatePattern* - Overrides configuration parameter 'SearchDatePattern'.
- *resultDatePattern* - Overrides configuration parameter 'ResultDatePattern'.

tss.xsd

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xsd:element name="Summary">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="Configuration"/>
        <xsd:element ref="SearchedParameters"/>
        <xsd:element ref="Search"/>
        <xsd:element ref="GoogleSearch" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="Configuration">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="Site" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="DocumentGroup" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="Site">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="Name"/>
        <xsd:element name="IndexDir" type="xsd:string"/>
        <xsd:element name="Download" type="xsd:boolean"/>
        <xsd:element name="Searched" type="xsd:boolean"/>
        <xsd:element name="FileNR" type="xsd:int"/>
        <xsd:element name="QueryNR" type="xsd:int"/>
        <xsd:element name="LastQuery" type="xsd:string"/>
        <xsd:element name="LastUpdate" type="xsd:string"/>
        <xsd:element ref="Path" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```



```
</xsd:complexType>
</xsd:element>
<xsd:element name="DocumentGroup">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="Name" />
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="Name" type="xsd:string"/>
<xsd:element name="Path">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="Type" type="xsd:string"/>
      <xsd:element name="Root" type="xsd:string"/>
      <xsd:element name="MappingRoot" type="xsd:string"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="SearchedParameters">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="Parameter" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="Parameter">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="Key" type="xsd:string"/>
      <xsd:element name="Value" type="xsd:string"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="Search">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="SearchSummary"/>
      <xsd:element ref="SearchResult"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="SearchSummary">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="SearchedTerm"/>
      <xsd:element ref="BeginIndex"/>
      <xsd:element ref="EndIndex"/>
      <xsd:element ref="DocsNumber"/>
      <xsd:element ref="SearchedTime"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="SearchedTerm" type="xsd:string"/>
<xsd:element name="BeginIndex" type="xsd:int"/>
<xsd:element name="EndIndex" type="xsd:int"/>
<xsd:element name="DocsNumber" type="xsd:int" nillable="false"/>
<xsd:element name="SearchedTime" type="xsd:string"/>
<xsd:element name="SearchResult">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="Result" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="Result">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="SiteName"/>
      <xsd:element ref="Title"/>
      <xsd:element name="PreviewURL" type="xsd:string"/>
      <xsd:element name="FileType" type="xsd:string"/>
      <xsd:element name="FileName" type="xsd:string"/>
      <xsd:element name="AbsolutFileName" type="xsd:string"/>
      <xsd:element name="AbsolutDirPath" type="xsd:string"/>
      <xsd:element name="SearchedTerm" type="xsd:string"/>
      <xsd:element name="AbsolutFilePath" type="xsd:string"/>
      <xsd:element name="RelativeDirPath" type="xsd:string"/>
      <xsd:element name="RelativeFilePath" type="xsd:string"/>
      <xsd:element name="InsideContainerFilePath" type="xsd:string"/>
      <xsd:element name="Mapped" type="xsd:string"/>
      <xsd:element name="GrantedDownload" type="xsd:string"/>
      <xsd:element name="Score" type="xsd:string"/>
      <xsd:element name="LuceneID" type="xsd:int"/>
      <xsd:element name="Modified" type="xsd:string"/>
      <xsd:element name="Created" type="xsd:string"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

```
<xsd:element name="Accessed" type="xsd:string"/>
<xsd:element name="Owner" type="xsd:string"/>
<xsd:element name="Size" type="xsd:string"/>
<xsd:element ref="Content"/>
<xsd:element name="Properties" type="xsd:string"/>
<xsd:element name="MetaData" type="xsd:string"/>
<xsd:element name="Author" type="xsd:string"/>
<xsd:element name="LastSavedBy" type="xsd:string"/>
<xsd:element name="Subject" type="xsd:string"/>
<xsd:element name="From" type="xsd:string"/>
<xsd:element name="To" type="xsd:string"/>
<xsd:element name="CC" type="xsd:string"/>
<xsd:element name="BCC" type="xsd:string"/>
<xsd:element name="MailSentDate" type="xsd:string" minOccurs="0" maxOccurs="1"/>
<xsd:element name="MailReceivedDate" type="xsd:string" minOccurs="0" maxOccurs="1"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="GoogleSearch">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="GoogleSearchSummary"/>
      <xsd:element ref="GoogleSearchResult"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="GoogleSearchSummary">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="SearchedTerm"/>
      <xsd:element ref="BeginIndex"/>
      <xsd:element ref="EndIndex"/>
      <xsd:element ref="DocsNumber"/>
      <xsd:element ref="SearchedTime"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="GoogleSearchResult">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="GoogleResult" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="GoogleResult">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="SiteName"/>
      <xsd:element ref="Title"/>
      <xsd:element name="URL" type="xsd:string"/>
      <xsd:element ref="Content"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="SiteName" type="xsd:string"/>
<xsd:element name="Title" type="xsd:string"/>
<xsd:element name="Content" type="xsd:string"/>
</xsd:schema>
```

Preview link

Clicking the preview (link) in the *Search Results* page opens the 'Together Document Viewer'.

Chapter 4. 'Together Document Viewer'

About

'Together Document Viewer' is application for previewing files. It could be used with 'Together Search Server' or as stand alone application with parameter `parseOnFly` when file will be first parsed and result will be shown. When it is used with s'Together Search Server' it shows indexer content of the file. It has two presentation object, `Preview.po` and `ShowPicture.po`.

Presentation Object Preview

This po object uses request parameters to form an XML file defined by a `preview.dtd`. After the XML is created the po object uses an xsl file defined in xsl request parameter (default is 'preiew.xsl') to transform the xml file and display the results.

Parameters

- *xsl* – The xsl file that will be used to transform xml file. If this parameter is not defined the default xsl will be used (`preview.xsl`). If `xsl=none` is used, output will be xml tree.
- *query* - The query was used for search.
- *id* – Lucene id of the document placed in the index. It must be used with parameter `SiteName`
- *SiteName* - The name of the site.
- *fullFilePath* – Valid URL of the file for preview. If `parseOnFly` is set to true, file will be parsed and result will be displayed. This way `siteName` and `MetaData` will not be available. Without `parseOnFly` parameter 'Together Search Server' engine will be searched for file with given file path and result from indexer will be shown (no parsing will be performed)
- *parseOnFly* – Overrides searching 'Together Search Server' engine and forces preview on fly
- *parserPageLimit* – Overrides configuration parameter 'Previewer.ParserPageLimit'.
- *parserCharacterLimit* – Overrides configuration parameter 'Previewer.ParserCharacterLimit'.

Preview.xsd

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xsd:element name="Summary">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="SearchedParameters"/>
        <xsd:element ref="Preview"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="SearchedParameters">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="Parameter" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="Parameter">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="Key" type="xsd:string"/>
        <xsd:element name="Value" type="xsd:string"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="Preview">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="Title" type="xsd:string"/>
        <xsd:element name="AbsolutFileName" type="xsd:string"/>
        <xsd:element ref="AbsolutFilePath"/>
        <xsd:element name="FileType" type="xsd:string"/>
        <xsd:element name="SiteName" type="xsd:string"/>
        <xsd:element name="From" type="xsd:string"/>
        <xsd:element name="To" type="xsd:string"/>
        <xsd:element name="CC" type="xsd:string"/>
        <xsd:element name="BCC" type="xsd:string"/>
        <xsd:element name="Subject" type="xsd:string"/>
        <xsd:element name="Sent" type="xsd:string"/>
        <xsd:element name="Received" type="xsd:string"/>
        <xsd:element name="Signed" type="xsd:string"/>
        <xsd:element name="Priority" type="xsd:string"/>
        <xsd:element name="DeliveryReceiptRequested" type="xsd:string"/>
        <xsd:element name="ReadReceiptRequested" type="xsd:string"/>
        <xsd:element name="Expires" type="xsd:string"/>
        <xsd:element name="Sensitivity" type="xsd:string"/>
        <xsd:element name="Content" type="xsd:string"/>
        <xsd:element name="Properties" type="xsd:string"/>
        <xsd:element name="MetaData" type="xsd:string"/>
        <xsd:element ref="ContainerFileList" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="AbsolutFilePath">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="FilePath" type="xsd:string"/>
        <xsd:element ref="InsideContainerFilePath" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="ContainerFileList">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="ContainerFile" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="ContainerFile">
    <xsd:complexType>
      <xsd:choice>
        <xsd:sequence>
          <xsd:element ref="InsideContainerFilePath"/>
          <xsd:element name="LuceneID" type="xsd:string" minOccurs="0"/>
        </xsd:sequence>
        <xsd:element ref="Preview"/>
      </xsd:choice>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

```
</xsd:element>
<xsd:element name="InsideContainerFilePath">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="FilePath" type="xsd:string"/>
      <xsd:element name="FileType" type="xsd:string"/>
      <xsd:element ref="InsideContainerFilePath" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
</xsd:schema>
```