Aquaforest

Tabula DX
Reference Guide

Version 1.30

CONTENTS

1   INTRODUCTION

Designed specifically for search-enabling large collections of PDF files via a browser interface, Tabula DX offers the following benefits and features:

Complete PDF Searchability - Search on PDF bookmarks, annotations, and metadata including XMP with no limit on the number of PDF pages that can be indexed with the "Unlimited" license.

Ease of Use - Present users with a familiar search interface and document thumbnails.

Performance and Scalability - Tabula DX is based on the Lucene search API which has been proven to robustly support collections of millions of documents.

Customizable - Simple user interface customization via XSL.

Integration Support - Search results can be returned as pure XML from any web-based method.

Designed for IIS and ASP.Net - Tabula DX is built using C# and ASP.Net for simple integration into a Microsoft-based environment.

License Model - Tabula DX is licensed per server and has no limits on the number of documents that can be indexed with the "Unlimited" license. A limited 50,000 document license is also available. Furthermore, for collections of up to 1,000 documents the product may be used free of charge.

Lucene Compatible - The Tabula DX Lucene indexes are compatible with tools supporting Lucene 1.4 or later.

Simple Web-Based Administration - The administration module allows creation of PDF collections, settings and index scheduling.

2.1   System Requirements

- Windows 7, Windows 8, Windows Server 2008, Windows Server 2012.

- Version 3.5 of the .NET Framework.

- Note: To run batch indexing jobs via the web interface a suitable user id and password will be required, this can be set using the Settings section of the Tabula DX web interface.

- Web Server: Tabula DX includes the lightweight UltiDev Cassini web server and the product is initially configured to use this. For production use IIS is recommended as a web server. See section 2.7 below for configuration details.

2.2   Installing Tabula DX

The installer tabuladx130.msi will install Tabula DX along with the Cassini lightweight web server. Please contact support@aquaforest.com should you require any assistance.

2.3   Testing the Installation

To get started with Tabula DX, access the Tabula DX Administration shortcut that is installed on the desktop and under the Programs menu.   A good initial test is to run the installation test from the Help page and to run a search on the sample PDF collection.
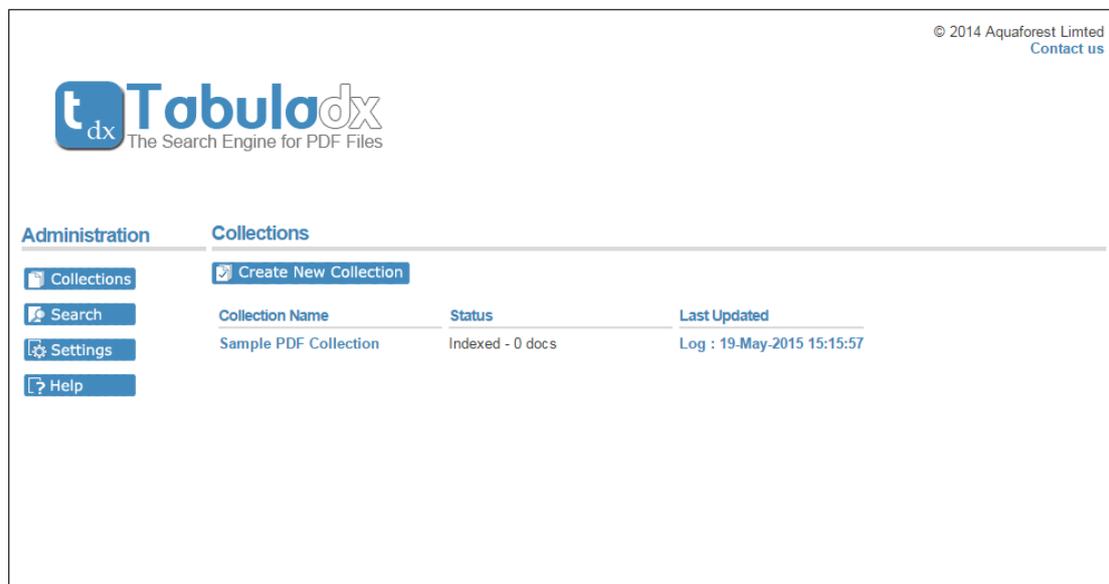
The main Tabula DX administration page

2.4   Trial License Restrictions

The trial license does not expire but limits the size of document collections to 1,000 documents. Please contact sales@aquaforest.com for additional assistance with trial licensing.

2.5   Uninstalling the Product

Tabula DX can be uninstalled via Windows Add / Remove Programs. UltiDev Cassini Web Server Explorer and UltiDev Cassini Web Server for ASP.Net 2.0 can be removed in the same way.

2.6   The Sample / Demo Collections

The product comes installed with a small demonstration collection of around 20 documents which can be searched as soon as the product has been installed.
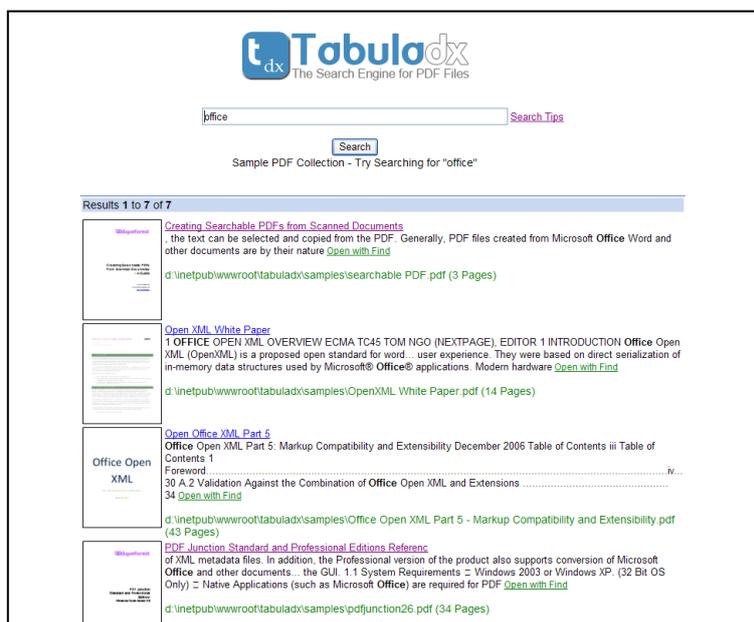
Demo Collection – Sample Search Results

The PDF file can be opened in a new browser window by clicking either the document thumbnail or the title link. In addition clicking on "Open with Find" will open the PDF document with the search string passed to Adobe Reader. This will automatically open the find interface within Adobe Reader using the same query parameters. See below for an example.
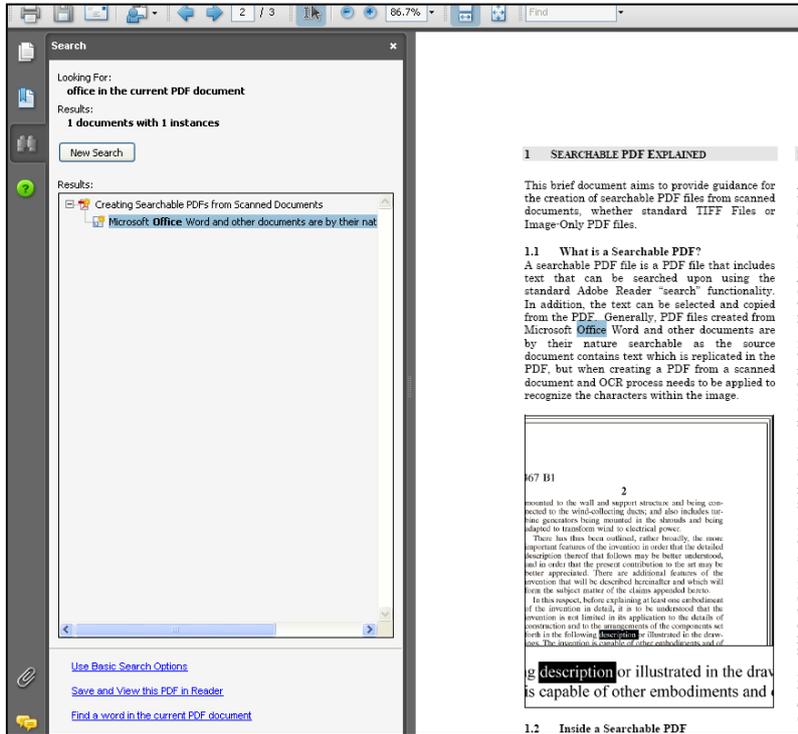
Opening a PDF Document "With Find"

2.7   Using Tabula DX with IIS

Tabula DX 1.30 is supported with the following versions of IIS.

- IIS 7.0
- IIS 7.5
- IIS 8.0
- IIS 8.5

Please note the following important system requirements:

- Tabula DX needs to be running under ASP.Net 4.0

- Tabula DX does require certain file system privileges which are unlikely to be satisfied by using the default IUSR account. Therefore the Tabula DX web application should be run using Integrated Authentication or with an anonymous user configured with sufficient privileges.

- Running under IIS7 will require use of the Classic ASP.Net application pool.

2.7.1   Using Tabula DX with IIS 7

Create a new web application directory called TabulaDX that points to the Tabula DX install location (by default C:\Program Files\Aquaforest\Tabula DX) and uses the Classic ASP.Net AppPool The screen shots below illustrate the process.

Set the default document to be main.aspx:

Ensure that the application runs with a suitably privileged identity. Not necessarily as administrator but with enough privilege to write to the Tabula DX collections folder.

Tabula DX administration can then be accessed via http://server/tabuladx :

2.7.2 Using Tabula DX with IIS 8

Create a new web site called TabulaDX that points to the Tabula DX install location (by default C:\Program Files\Aquaforest\Tabula DX) and uses the Classic ASP.Net AppPool.
The screen shots below illustrate the process.
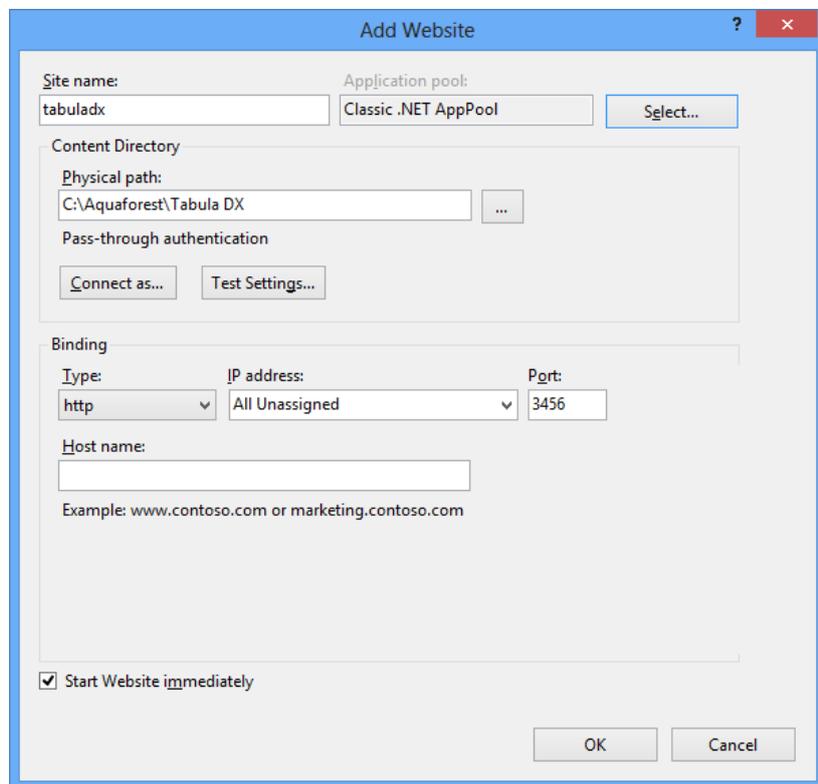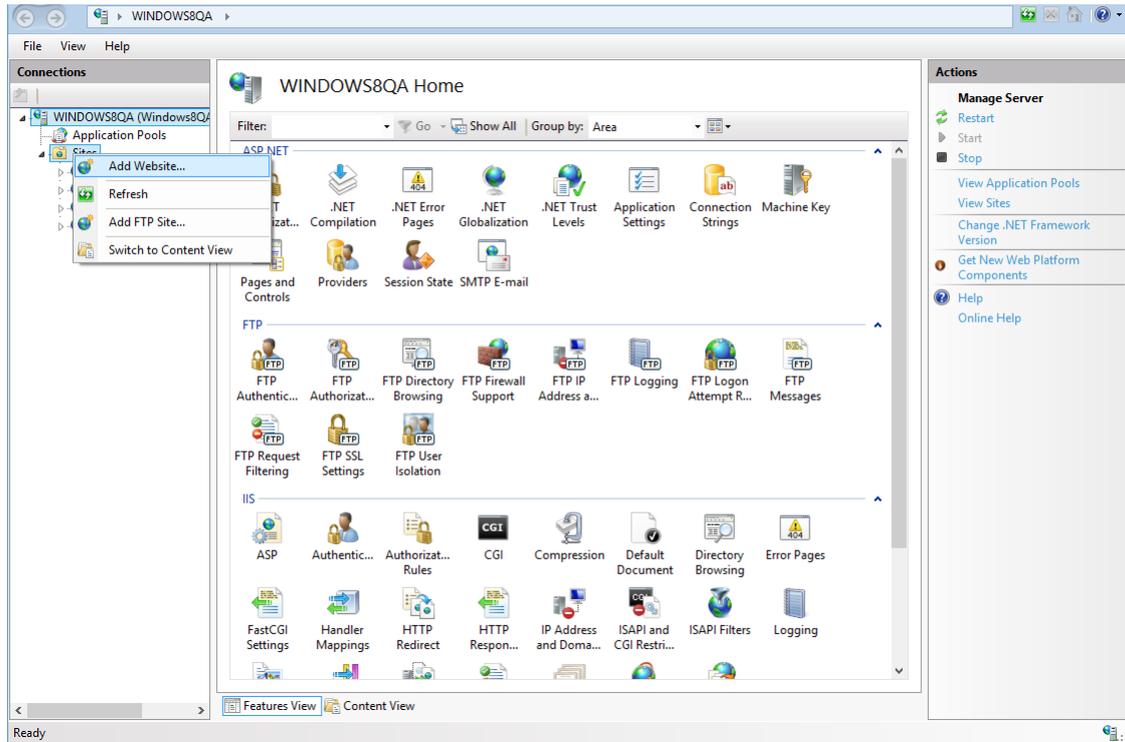
Go to the advance settings of the Classic ASP.Net AppPool and set the **Enable 32 bit support** option to true.
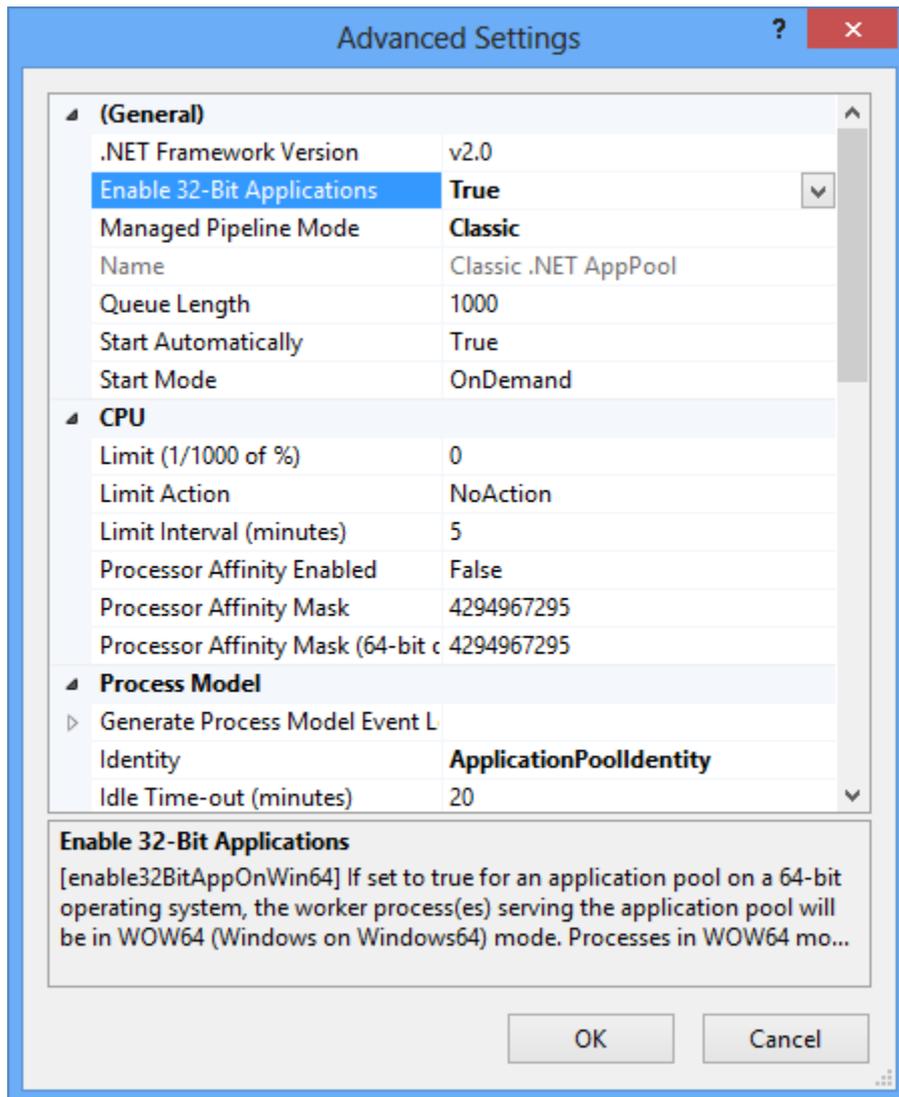
Set the default document to be main.aspx:
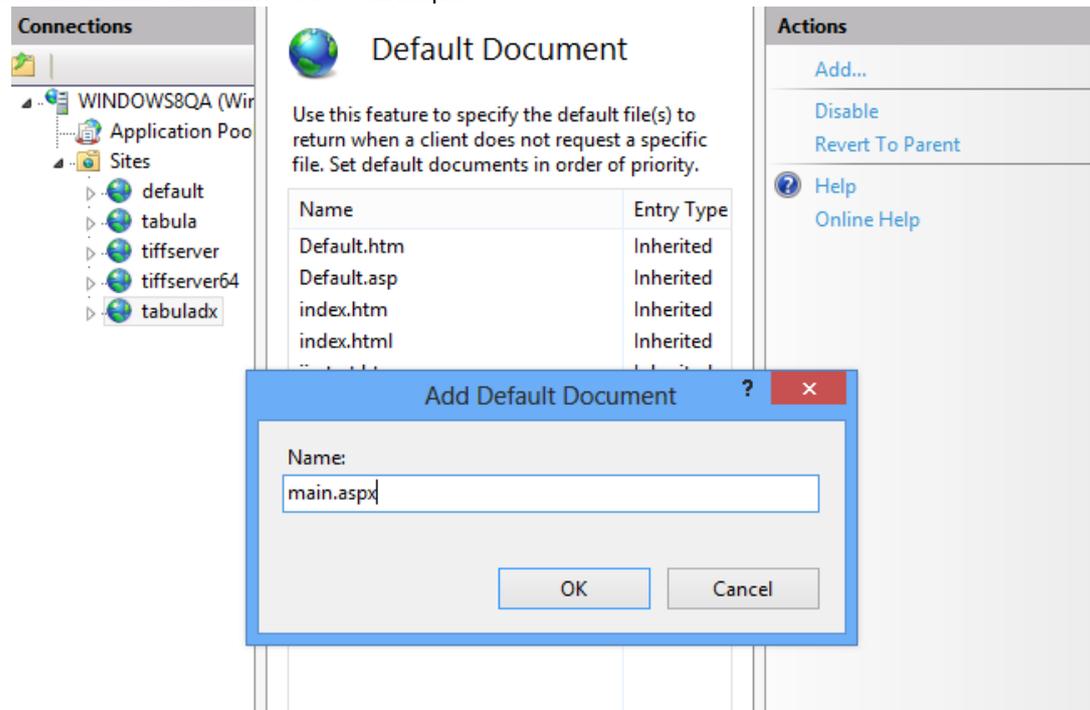
Ensure that the application runs with a suitably privileged identity. Not necessarily as administrator but with enough privilege to write to the Tabula DX collections folder.

Tabula DX administration can then be accessed via http://server/tabuladx :
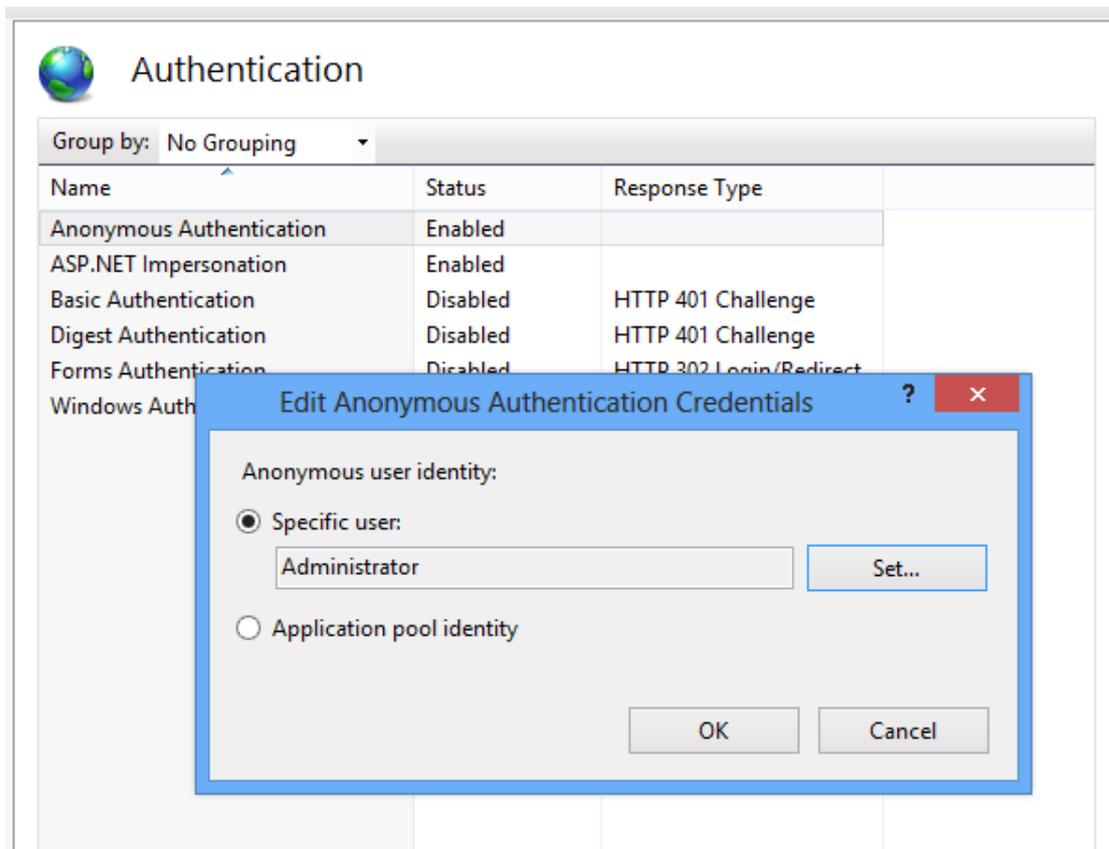
3 SEARCH QUERY EXPRESSIONS

3.1 Search Fields

When documents are indexed, a number of different search index fields are populated depending upon the collection configuration.

Field	Contains
Contents	The text of the document. This is the default field.
Bookmarks	
Annotations	
PDF Doc Info Fields (or XMP equivalents) Title Author Subject Keywords Producer Creator Creationdate Moddate In addition Custom metadata fields that also form part of the Document Information Dictionary can also be searched.	The value of these fields can be set via the Acrobat Document Properties tab.
xmpfield	The value of the XMP metadata field configured with the search name *xmpfield* see section 4.3 for details of how this is configured.
Collectionid	The collection ID of the document.
Indextime	Time stamp in YYYYMMDDHHMMSS format
Path	PDF file path
Thumbnailpath	Path of the thumbnail image (or blank if thumbnails are not used in the collection)
Pages	The number of pages in the PDF document
Filesize	The size in bytes of the file
Wincreated	Time stamp in YYYYMMDDHHMMSS format
Winmodified	Time stamp in YYYYMMDDHHMMSS format

3.2 Query Expressions

Search Query	Matches documents ...
Pdf	Contains the word "pdf" in the contents of the document.
Pdf search Pdf AND search +pdf +search	Each of these expressions will find documents with both of the words "pdf" and "search" in the contents of the document.
Pdf OR search	This will find documents with either (or both) of the words "pdf" and "search" in the contents of the document.
search AND NOT pdf	Documents that contain the term search but do not contain the term PDF.
Title:china AND –title:india	The title field contains the word china but not india
(pdf or wordperfect) AND search	Documents contain the word "search" and either "pdf" or "wordperfect".
Title: "search engines"	The title field contains the phrase search engines
Search*	Contains terms that begin with search, such as searchable, searching and search.
Search~	Contains terms that are close to the word search such as Starch (fuzzy searching)
winmodified:[20070701000000 TO 20070731235959]	Contains winmodified values in the range specified
"jakarta apache"~10	To do a proximity search use the tilde, "~", symbol at the end of a Phrase. For example to search for a "apache" and "jakarta" within 10 words of each other in a document use the search:

4.1   System-Wide Settings

The "Settings" tab allows a number of system-wide settings to be maintained. These settings are shown below.

Attribute	Description
License Key	Once purchased, a permanent license key may be entered here.
Administrator Password	If a password is entered, then access to the Autobahn DX admin pages will require entry of the specified password.
Index Username and Password	To run indexing jobs via the web interface a suitable user id and password will be required which is used to create the job using Windows Scheduled Tasks.
Default Collection ID	This defines which collection is selected by default in the "Search" function.

4.2   Collection Settings

Individual collections can be configured by clicking on the collection name link in the main collections page:

4.3   Collection Attributes

Attribute	Description
Folder Paths	One or more folders containing documents to be indexed. Multiple folders should be specified one per line.
Excluded Folders	One or more folders containing documents to be excluded from the Indexing. Multiple folders should be specified one per line.
Collection Name	The descriptive name of the document collection
Description	The description is shown on the search page for the collection under the search box
Index Batch Size	The maximum number of documents that will be indexed in a single run of the indexer.
Results per Page	Default number of results per search results page.
XSL File	Default XSL file.
Always Optimize	If checked, the indexes will be optimized at the end of each run of the indexer.
Check for Deleted	If checked, the index process will check each file in the index. If the related PDF file has been deleted, the index entry will be removed.
Thumbnails	If checked a thumbnail image of the first page of each PDF document indexed will be produced.
Thumbnail Folder	Thumbnail images will be stored in this folder. Subfolders will be automatically created to mirror the structure of the source PDF folders.
Index Logging	If checked, a log file will be created (or appended to) detailing indexing activity each time the indexer is run on this collection.
Default Sort Order	Comma separated list of sort fields e.g. field1, field2. If left blank results will be returned in descending relevance order. Can be overridden by URL parameters when using search.aspx directly (see section 6).
Default Sort Type	Comma separated list of sort field datatypes (string, int or float) corresponding to each sortfield e.g. string, int. If left blank all types will be assumed to be strings. Can be overridden by URL parameters when using search.aspx directly (see section 6).
Default Sort Order Asc	Comma separated of true / false values corresponding to each sortfield used to indicate ascending or descending sort order. If left blank ascending order will be used. Can be overridden by URL parameters when using search.aspx directly (see section 6).
Index Doc Info	If checked, PDF Doc Info metadata (title, author etc.) will be indexed and can be searched using query expressions such as author:Shakespeare

	In addition Custom metadata fields that also form part of the Document Information Dictionary can also be will also be indexed.
Index Bookmarks	If checked, the text of bookmarks will be indexed and can be searched using query expressions such as bookmarks:shakespeare
Index Annotations	If checked, PDF annotations will be indexed and can be searched using query expressions such as annotations:shakespeare
Index XMP	If checked, XMP metadata will be indexed in accordance with the "XMP Fields" instructions.
XMP Fields	This defines which XMP fields should be indexed, and each line defines a property of the form : **namespace**:*propertyname*,<u>indexname</u> For example : http://www.aiim.org/pdfa/ns/id/:conformance,pdfaconformance The above instructs Tabula DX to index the property "conformance" in the PDF/A namespace (http://www.aiim.org/pdfa/ns/id/) and index it under the name pdfaconformance. The field may be searched to find pdf/a conformant files with a query such as pdfaconformance:B For further information about XMP refer to the Adobe resources here : http://www.adobe.com/devnet/xmp/pdfs/xmp_specification.pdf NOTE: Standard PDF custom metadata items are automatically indexed. The XMP fields only need to be specified for custom schema.
Index Folder	The folder to contain the index files.
File pattern	A pattern to be used to match the files to be indexed. Default *.pdf
Index Logging File	The index log file name. The file will be placed in AUTOBAHN/collections/*collectionid*/indexlog. The string %TIMESTAMP% can be included in the file name to create a unique file for each indexer run; the timestamp is of the format YYYYMMDDHHMMSS.

4.4   Configuring Collection Indexing

The configuring collection indexing page allows a collection to be indexed according to a set schedule or immediately. The indexing process analyzes the current collection index and the set of file system files, determining which files need to be indexed or re-indexed.

Tabula DX collections can be indexed by using the command line interface. The tabuladx.exe executable can be found in the product bin folder.

---

tabuladx.exe /id=*collectionid* /op=*operation* [/debug]

Parameter	Description
Id	The collection ID, e.g. 1002
Op	The operation : Index – Index the collection. The indexing process analyzes the current collection index and the set of file system files, determining which files need to be indexed or reindexed. clear – Clear the index collection The executable can is also used to set up the demo collection, this should be performed automatically by the setup process. Setupdemo – Sets up the demo collection from the template. Adjustdemo – Adjust the collection for the local location
Debug	Optional. If specified verbose output is produced.

5.1   Indexing Configuration

The tabuladx.exe configuration file is *tabuladx.exe.config* located in the product bin folder. The parameters described below can be set in this file.

```
<setting name="ExtractEngineA" serializeAs="String">
        <value>3</value>
</setting>
<setting name="ExtractEngineB" serializeAs="String">
        <value>2</value>
</setting>
<setting name="ExtractEngineC" serializeAs="String">
        <value>1</value>
</setting>

<setting name="ThumbnailWidth" serializeAs="String">
    <value>100</value>
</setting>
<setting name="Debug" serializeAs="String">
    <value>False</value>
</setting>
```

ExtractEngineA/B/C
This controls the order in which Text Extraction engines are called by the product to extract text from PDF files. There are three engines used – engine "1", engine "2" and engine "3". In general there should be no need to edit this file unless asked to by technical support, or you wish to compare the text results from the different extraction engines.

"ExtractEngineC" is run first (by default this is engine "1") and if this doesn't return any text then "ExtractEngineB" is run next (by default this is engine "2") and if this doesn't return any text then "ExtractEngineA" is run (by default this is engine "3"). These settings can be changed by changing the values.

Thumbnail Width
This defines the image width of the thumbnail generated from the first page of the PDF file.

Debug
If set to true then the indexing log will contain very detailed indexing information which may be useful to technical support.

6 CUSTOMIZATION AND INTEGRATION

Tabula DX is designed to be customized and can be easily integrated within larger solutions. The sample URLs below assume that Tabula DX has been installed against IIS.

It is possible to give end users a direct URL to enable search, even using the default Cassini web server using the URL below, with *servername* replaced accordingly.

http://servername:7756/GoToApplication.aspx?AppID=106ECFD5-15AC-4AC6-84B4-C805469DB510

6.1 Examples of Calling Search Pages

The search page offering a choice of collections can be called directly via the search.aspx page:
http://localhost/tabuladx/search.aspx?collectionlist=1

An individual collection search page can be called as shown in the following example:
http://localhost/tabuladx/search.aspx?collectionid=1003

A search can be directly issued via a URL such as:
http://localhost/tabuladx/search.aspx?collectionid=1001&query=office&xslfile=xml

6.2 Search.aspx Parameters

The full list of search.aspx parameters is detailed below.

Parameter	Description	Default Value if Unspecified
Collectionlist	If set to 1, a drop down is shown allowing the user to select the collection that they wish to search.	0
Logo	If set to 0, the Tabula DX logo is not shown.	1
Collectionid	The numeric collection id to be searched.	N/A
Query	The query string	N/A
Resultfields	The list of fields to be returned	highlight,path,title, thumbnailpath,pages
Resultstart	The result set document number of the first document to be returned.	1
Resultsperpage	The maximum number of results to be returned.	From collection configuration.
Xslfile	Set to XML to have pure XML returned (see 6.4 below) or alternatively specify an alternate XSL file.	From collection configuration.
Thumbnails	True or false. If set to false thumbnails are not to be displayed.	From collection configuration.
Indexdirectory	Index files directory	From collection configuration.
Sortorder	Comma separated list of sort fields e.g. field1,field2	Results will be returned in descending relevance order.
Sorttype	Comma separated list of sort field datatypes (string, int or float) corresponding to each	All types will be assumed to be strings.

	sortfield e.g. string,int	
Sortasc	Comma separated of true / false values corresponding to each sortfield used to indicate ascending or descending sort order.	Ascending order will be used.

6.3   Customizing the search interface

When running a search, Tabula DX generates an XML file with details of the search results. This is passed to the browser with default of the XSL file to be used to transform the output into the search results page.

By default the style/results.xsl file is used. This can be customized to suit specific needs. Alternative stylesheets can be used by specifying the XSL stylesheet path in the Collection Settings page.

Two example alternative files are provided

6.3.1   resultslinklocal.xsl

To support easily editing PDF files, this can be used to launch Adobe Acrobat directly on the PDF file in question by using the green file path link. This can only be used locally (i.e. on the same machine that file files are located) and users need to have permission to run Active X controls on the web page. To ensure that the application has the correct path to Adobe Acrobat you can either edit the xsl file so that instead of acrobat.exe it has the full path (you'll need to use \\ instead of \ as \ is a special character) or perhaps better, add the folder that contains acrobat.exe to the system path (Control Panel | System | Advanced Settings | Environment Variables | System Variables | PATH.

6.3.2   resultscustom.xsl

This demonstrates the use of custom search fields and manipulation of the search string. In this example, the title field allows search of the title without having to specify title:…. in the query. Similarly the creation date fields allow for a convenient date input.
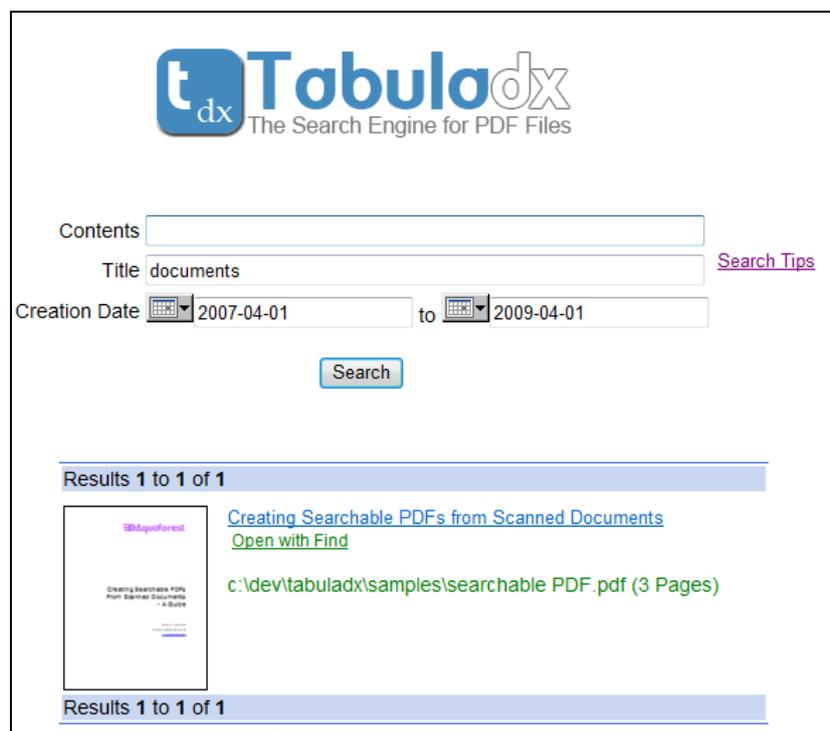
To review how this is accomplished, analyze the xsl file and in particular the field definitions near the top of the file. The field name must be of the form **_add_name._**

```xsl
<xsl:comment>  Search Fields </xsl:comment>
        <table>
         <tr>
          <td  align="right">Contents</td>
          <td>
           <input name="add_contents" size="60" >
             <xsl:attribute name="value">
               <xsl:value-of select="searchresults/search/add_contents"/>
             </xsl:attribute>
           </input>
          </td>
         </tr>
         <tr>
…

<xsl:comment> End Search Fields </xsl:comment>
```

The JavaScript function *adjustQuery*() which appears near the end of the XSL file. This constructs the query as it would have been if the end user had typed in the full syntax for the query by placing the full query in the query field of the query form.

```xsl
 <xsl:comment> The adjustQuery function constructs the necessary query from the individual fields </xsl:comment>


    function adjustQuery()
    {
    var q="";
    var contents=document.query.add_contents.value;
    var title=document.query.add_title.value;
    var modstartdate=document.query.add_modstartdate.value;
    var modenddate=document.query.add_modenddate.value;

    if(contents!="")
    q+=contents;

    if(title!="")
    q+=" title:"+title;

    if(modstartdate!="")
    {
    modstartdate=""+modstartdate.replace(/-/g,"")+"000000";
    modenddate=""+modenddate.replace(/-/g,"")+"235959";
    q+=" winmodified:["+modstartdate+" TO "+modenddate+"]";
    }

    document.query.query.value=q;
        }
```

6.4 XML Output

Some applications may wish to directly consume the XML output from search.aspx. An example file segment and definition of each of the XML attributes is provided below.

```xml
<?xml version="1.0" encoding="utf-8"?>
<searchresults>
 <search>
  <collectionid>1001</collectionid>
  <query>document guidelines</query>
  <resultfields>highlight,path,title,thumbnailpath,pages</resultfields>
  <resultstart>1</resultstart>
  <resultend>9</resultend>
  <resultsperpage>10</resultsperpage>
  <sortorder />
  <thumbnails>true</thumbnails>
  <indexdirectory>c:\dev\scrii\index\</indexdirectory>
  <xslfile />
 </search>
 <hits>9</hits>
 <searchstatus>success</searchstatus>
 <results>
  <result>
   <row>1</row>
   <highlight>
    <B>Guidelines</B>for Creating Archival ……
   </highlight>
   <path>c:\docs2\PDFGuideline.pdf</path>
   <title>PDF Guideline for FDA</title>
   <thumbnailpath>c:\thumbnails\PDFG.pdf.1.gif</thumbnailpath>
   <pages>6</pages>
  </result>
  <result>
   <row>2</row>
  </result>
 </results>
</searchresults>
```

Attribute	Description
<search>	This contains the search attributes
<collectionid>	The collection ID
<query>	The search query
<resultfields>	The fields that will be included in <results>
<resultstart>	The result number of the first document in <results>
<resultend>	The result number of the last document in <results>
<resultsperpage>	The maximum number of results
<thumbnails>	True or False (see <thumbnailpath> if true).
<indexdirectory>	The location of the collection index directory
<hits>	The total number of results from the search
<searchstatus>	Can be : success, blank (if query is blank), nomoreresults , error
<results>	The result document set
<result>	An individual result
<row>	The sequence in the result set
<field>	The contents of *field* for the document
<highlight>	Document fragment, highlighted with search terms
<path>	The path of the document
<title>	Document title. If the PDF document does not have a title, the first 60 characters of the document content is used.
<thumbnailpath>	The path of the thumbnail image
<pages>	The number of pages in the PDF document.

6.5 Web.config Parameters

The web.config file contains a number of appSettings that can be adjusted. These parameters are defined below.

Setting	Description
generatedTitleLength	For PDF documents with no title, Tabula DX generates a title for search results purposes using the first *generatedTitleLength* characters in the file. Initial value : 60.
highlighMaxDocBytesToAnaylze	When constructing the "highlight" text fragment for results display, this parameter determines how many characters of text will be examined. Initial value : 100000.
highlightNumFragments	Determines how many text fragments will be used in the highlight text. Initial value : 2.
highlightDelimiter	Specifies a character string to be used to separate the highlight text fragments. Initial value : "…"
highlightLength	Defines the length of highlight text. Initial value 150.
defaultOperator	Defines whether "AND" or "OR" is the default search operator. Initial value : "AND".
resizeThumbnails	By default thumbnails are a width of 100 pixels. They can be *resized* on the fly if this parameters is set to true. Initial value false. Note that if a different original thumbnail size is required this can be achieved by modifying the parameters in tabuladx.exe.config – see section 5.1.
resizeThumbnailWidth	Defines the image width if resizeThumbnails is set to true.

7   TABULA DX DIRECTORIES

The Tabula DX folder structure is explained below. The root folder is typically c:\inetpub\wwwroot\tabuladx

Folder	Description
Bin	Contains the DLLs and executables, including tabuladx.exe
Collections	The root folder for the search collections.
Collections/9999	The root folder for the search collections 9999. Includes the config_9999.xml file which holds the collection configuration.
Collections/9999/index	The default location for the collection index files.
Collections/9999/indexlog	The default location for the collection index log files.
Collections/9999/temp	The location for the collection index log files.
Collections/9999/thumbnails	The default location for the collection thumbnail files.
Config	Contains the Tabula DX config file and the template collection template file.
Docs	Contains the reference guide and license file.
Img	Contains the images used in the web interface.
Samples	Contains the sample collection documents.
Style	Default location for XSL documents. Includes results.xsl.
Template	The template for the sample collection.

The indexes created by Tabula DX are compatible with Lucene 1.4 or later. More information regarding Lucene can be found here: http://lucene.apache.org/java/docs/

Included with the product is Luke – the Lucene Index Toolbox. This can be a useful tool for analyzing the contents of indexes and running queries. Luke can be launched by running lukeall-0.7.1.jar in the product bin folder. Luke is Licensed under the Apache License, Version 2.0 (the "License"); you may you may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0

9 ACKNOWLEDGEMENTS

This product includes Luke - Lucene Index Toolbox (http://www.getopt.org/luke), Copyright 2008 Andrzej Bialecki.

This product includes iText Copyright (C) 1999-2009 by Bruno Lowagie and Paulo Soares et al. All Rights Reserved. Binaries distributed under the Mozilla Public License.