

## OpenText™ AppEnhancer Indexing Service

### **User Guide**

This guide provides instructions about installing, configuring, and using AppEnhancer Indexing Service.

EAXCORE250200-AIS-EN-1

---

## **OpenText™ AppEnhancer Indexing Service User Guide**

EAXCORE250200-AIS-EN-1

Rev.: 2025-Apr-08

**This documentation has been created for OpenText™ AppEnhancer Indexing Service CE 25.2.**

It is also valid for subsequent software releases unless OpenText has made newer documentation available with the product, on an OpenText website, or by any other means.

### **Open Text Corporation**

275 Frank Tompa Drive, Waterloo, Ontario, Canada, N2L 0A1

Tel: +1-519-888-7111

Toll Free Canada/USA: 1-800-499-6544 International: +800-4996-5440

Fax: +1-519-888-0677

Support: <https://support.opentext.com>

For more information, visit <https://www.opentext.com>

### **© 2025 Open Text**

Patents may cover this product, see <https://www.opentext.com/patents>.

### **Disclaimer**

No Warranties and Limitation of Liability

Every effort has been made to ensure the accuracy of the features and techniques presented in this publication. However, Open Text Corporation and its affiliates accept no responsibility and offer no warranty whether expressed or implied, for the accuracy of this publication.

---

# Table of Contents

<b>1</b>	<b>Starting Indexing Service implementation</b>	<b>5</b>
1.1	Introduction to Indexing Service	5
1.2	Configuring AppEnhancer to Use the Indexing Service	6
1.2.1	AppEnhancer Full-Text Engine	6
1.2.1.1	Full-Text Indexing	6
1.2.1.2	Full-Text Searching	7
1.2.1.3	Building Full-Text Search Expressions	8
1.3	Creating the Impersonation Account	10
1.3.1	Creating the Indexing Service Impersonation Account	11
1.3.2	Adding Advanced Rights to the Indexing Service Impersonation Account	12
<b>2</b>	<b>Installing, Configuring, and Using AppEnhancer Full-text Server</b>	<b>13</b>
2.1	Installing AppEnhancer Full-Text Server	13
2.2	Configure idol.common.cfg for AppEnhancer Full-Text Server	14
2.3	Creating Full-Text Database	15
2.4	Configuring Applications to use AppEnhancer Full-text Engine in AppEnhancer Administrator	15
<b>3</b>	<b>Registering Indexing Service via Component Registration Wizard (CRW) and Configuration Indexing Service</b>	<b>17</b>
3.1	Registration Overview	17
3.2	Configuring Indexing Service Overview	17
3.2.1	Configuration through Administrator	18
3.2.1.1	Service Credentials tab	18
3.2.1.2	Settings tab	18
3.2.1.3	Queues	19
3.2.2	Configuration through Configuration file (XtenderSolutions.IndexingService.exe.config)	19
3.3	Managing Queues	21
3.3.1	Adding a New Queue	21
3.3.2	Modifying a Queue	22
3.3.3	Deleting a Queue	22
3.3.4	Moving Queues between Available Queues and Processing Queues lists	23
<b>4</b>	<b>Managing the Indexing Service</b>	<b>25</b>
4.1	Resubmitting Documents to the Indexing Service	25
4.2	Controlling the Indexing Service	25
4.3	Monitoring the Indexing Service	25

4.3.1	Viewing Indexing Service Activities .....	25
4.3.2	Viewing the Indexing Service Footprint in Registered or Running Components .....	26
4.4	Logging .....	26
<b>5</b>	<b>Indexing documents with Full-Text server .....</b>	<b>29</b>
5.1	Massive Indexing mode .....	29
5.2	Multiple Indexing Service instances and Full-Text servers .....	31
5.2.1	Multiple Indexing Services .....	31
5.2.2	Full-Text servers .....	31

## Chapter 1

# Starting Indexing Service implementation

The Indexing Service, previously known as Index Agent, is an optional module that adds full-text indexing and OCR functionality to AppEnhancer systems.

This chapter describes implementation of the Indexing Service. It covers the following topics:

- [“Introduction to Indexing Service” on page 5](#)
- [“Configuring AppEnhancer to Use the Indexing Service” on page 6](#)
- [“Creating the Impersonation Account” on page 10](#)

## 1.1 Introduction to Indexing Service

Users can submit text to Indexing service for full-text indexing. With full-text indexing, users can search for documents in AppEnhancer Web Access .NET by index values and/or keyword content within the document.

To submit text to the Indexing Service, users can submit entire AppEnhancer documents to the server for indexing. They can also submit scanned documents and pages to the server to be processed using OCR and have the resultant text added to Indexing Service automatically. Users can also submit documents to Indexing Service for OCR processing only.

The mechanism in which documents are submitted is in the form of job queuing. This is the method used for both full-text indexing and OCR processing. The user either uses Web Access .NET to submit the full-text queues and OCR queues. The Indexing Service polls the queues for the next job (document or page) to process. Queues are set up to accommodate jobs from multiple workstations, and can be configured at any Indexing Service computer. The queues are database table rows (in the AppEnhancer database) that track which documents are processed by Indexing Service.



**Note:** Full-text queues and OCR queues are created and monitored using AppEnhancer Administrator. Once you create a queue, you must designate the queue as a processing queue. This can only be done using the AppEnhancer Administrator. For more information on designating a queue, see [“Adding a New Queue” on page 21](#) and [“Moving Queues between Available Queues and Processing Queues lists” on page 23](#).

## 1.2 Configuring AppEnhancer to Use the Indexing Service

The following instructions describes how to configure the AppEnhancer system to use Indexing Service.

### 1.2.1 AppEnhancer Full-Text Engine

OpenText AppEnhancer full-text server is a multi-instance, scalable, high-performance, full-text index server that can be configured for high availability and disaster recovery. AppEnhancer full-text engine is the new full-text engine for AppEnhancer.



**Note:** AppEnhancer full-text server 24.2 or later is required.

The following tables list out full-text indexing, searching, and using multiple processors in AppEnhancer full-text server:

- [“Full-Text Indexing” on page 6](#)
- [“Full-Text Searching” on page 7](#)
- [“Building Full-Text Search Expressions” on page 8](#)

#### 1.2.1.1 Full-Text Indexing

The following table describes how AppEnhancer full-text processing engine processes each page type:

Page Type	AppEnhancer Full-Text Server
Image	Indexing Service uses OCR on each image file to generate a text file for AppEnhancer full-text server to index.
COLD/Text	AppEnhancer full-text server indexes the COLD/Text file. Areas normally covered by redactions and non-transparent text annotations are replaced with blank spaces before indexing.
PDF	Indexing Service does not convert PDF files into text files. PDF files are converted to a text representation for indexing in AppEnhancer full-text server. If a PDF file is an image-only PDF file, Indexing Service uses OCR to generate a text file for AppEnhancer full-text server to index.

Page Type	AppEnhancer Full-Text Server
RTF	Indexing Service does not convert RTF files to text files. RTF files are converted to a text representation for indexing in AppEnhancer full-text server.
HTML	Indexing Service does not convert HTML files to text files. HTML files are converted to a text representation for indexing in AppEnhancer full-text server.
Foreign	Indexing Service does not convert foreign files to text files. Foreign files are converted to a text representation for indexing in AppEnhancer full-text server.
OLE Embedded files	Indexing Service does not process OLE files.

### 1.2.1.2 Full-Text Searching


The following table describes the appearance of full-text search results in Web Access .NET:

Page Type	xPlore
Image	AppEnhancer highlights the hit results using the text view of the image in the native AppEnhancer viewer.
COLD/Text	AppEnhancer highlights the hit results in the native AppEnhancer viewer. Areas normally covered by redactions and non-transparent text annotations are replaced with blank spaces. Annotations and form overlay do not apply. You can use the text view toolbar button or menu item to switch to normal text display mode with annotations and form overlays.
PDF	AppEnhancer highlights the hit results in the native AppEnhancer viewer.
PDF Image	AppEnhancer highlights the hit results in the native AppEnhancer viewer.
RTF	AppEnhancer highlights the hit results in the Foreign File Viewer.
HTML	AppEnhancer highlights the hit results in the Foreign File Viewer.
Foreign	AppEnhancer highlights the hit results in the Foreign File Viewer.

### 1.2.1.3 Building Full-Text Search Expressions

You can select one of the full-text search options for your query. The options include: All Words, Any Words, Exact Phrase, or Expression.

This section describes how to build full-text search expressions with AppEnhancer full-text engine.

 **Note:** You must enclose the words and text strings in single quotation marks.

The following is a list of expression operators supported by the AppEnhancer full-text engine:

Operator	Example	Within the Documents	Highlighted Occurrences	Detail
ftand	'AEX' ftand 'Simon'	Both AEX and Simon	AEX and Simon	When you use the ftand operator between two words, AppEnhancer searches for documents that contain both words.
ftor	'AEX' ftor 'Simon'	Either AEX or Simon	AEX and Simon	When you use the ftor operator between two words, AppEnhancer searches for documents that contain at least one of the words.
ftnot	'AEX' ftand ftnot 'Simon'	AEX but not Simon	AEX	When you use the ftnot operator before a word, AppEnhancer searches for documents that do not contain the word.

Operator	Example	Within the Documents	Highlighted Occurrences	Detail
?	'A?X' with wildcards	A string of 3 consecutive characters, in which the first must be A, the second may be any character, and the last must be X	Matching words	When you use a question mark (?) in a string of text with wildcards, AppEnhancer searches for documents that contain that string of text, except when other any single character replaces the question mark. (If you are searching for multiple words, make sure that another operator is used between each word.)
*	'S*' with wildcards	Any string that begins with S	Matching words	When you use an asterisk (*) in a string of text with wildcards, AppEnhancer searches for documents that contain that string of text, except when any other characters or even lack of a character replaces the asterisk. (If you are searching for multiple words, make sure that another operator is used between each word.)

Operator	Example	Within the Documents	Highlighted Occurrences	Detail
()	'AEX' ftand ('Simon' ftor 'Schiff')	Either Simon or Schiff first, and of those documents, AppEnhancer finds ones that also have AEX	AEX, Simon and Schiff	By default, AppEnhancer interprets a search expression from left to right, interpreting each search operator one at a time, and narrowing the results with each subsequent search. You can use parentheses to group search expressions to control the order in which AppEnhancer performs these searches. When you use parentheses in an expression, AppEnhancer searches first for documents that contain the expression in parentheses.

### 1.3 Creating the Impersonation Account

The Indexing Service impersonation account grants security privileges to Indexing Service and is essential for Indexing Service to operate properly. Indexing Service uses the impersonation account to access remote and server resources, including AppEnhancer document paths, the License Server, the registry, and the image repository.

Because no login is required for the Indexing Service to run, some security context needs to be established, and this is done with the Indexing Service impersonation account. The Indexing Service uses the same security provider as AppEnhancer to sign in and access images and documents. The impersonation account must have the Log on as a service advanced user right.

If any of the components used by Indexing Service will be located remotely, the Indexing Service impersonation account must be added as a domain account on a domain that is trusted by the Indexing Service and License Server. For example, you may install Indexing Service on one computer and install SQL Server and AppEnhancer Web Access Server on another computer to allow for faster

processing. The domain account and the local accounts must be the same Windows account for account rights to translate across the domain to the local machines.

Also, in a native Active Directory environment, when domain-level permissions are active, they override local permissions, so it may be necessary to grant the two advanced rights at the domain level.

It is recommended that the Indexing Service impersonation account be created as a domain account and then added locally. By adding the domain account to the local Indexing Service, you ensure that only one account is used.

After creating the impersonation account, user will be able to verify this account in indexing service configuration of Administrator by saving the information. System validation will check. You must create and correctly configure an impersonation account to run the Indexing Service.

### 1.3.1 Creating the Indexing Service Impersonation Account

1. Sign in to the computer on which Indexing Service has been installed. Make sure that you are signed in as a member of the local Administrators group.
2. From the Windows Start menu, click **Programs > Administrative Tools > Computer Management**.  
The Computer Management console appears.
3. Click **System Tools > Local Users and Groups**. Right-click **Users**, and select **New User**.  
The **New User** dialog box appears.
4. Enter a **User name**, **Full name**, **Description**, **Password**, and **Password confirmation** in the available text boxes.
5. Disable the **User must change password** at next logon check box, and select the **Password never expires** check box.
6. Click **Create**. The new account is created. Click **Close** to exit the **New User** dialog box. When you return to the Computer Management console, the list of local users should appear in the console's right-side pane (if it does not appear, click the **Users** folder, located beneath **Local Users and Groups**).
7. Right-click the account you just created and select **Properties**.  
The User Properties pages appear.
8. Click the **Member Of** tab.
9. Click **Add**.  
The **Select Groups** dialog box appears.
10. Select **Administrators** from the top Groups pane, and click **Add**. `DOMAINNAME\Administrators` should appear in the bottom pane. Click **OK** to return to the

**User Properties** page. Click **OK** again to exit the User Properties page and return to Computer Management.

11. Close Computer Management. You now need to configure the new account to have the following advanced right: Log on as a service.

### 1.3.2 Adding Advanced Rights to the Indexing Service Impersonation Account

The Indexing Service impersonation account requires the advanced right to Log on as a service.

1. Open the **Local Security Settings** utility. From the Start menu, select **Programs**, then **Administrative Tools**. From the **Administrative Tools** menu, choose **Local Security Policy**.

The Local Security Settings Console appears.



**Note:** This method may differ depending on your operating system.

2. Expand the **Local Policies** folder and then select **User Rights Assignment**. All user rights should appear in the right-side pane of the Local Security Settings Console.
3. Right-click **Log on as a Service** and choose **Security**.  
The **Local Security Policy Setting** dialog box appears.
4. Click **Add**.  
The **Select Users or Groups** dialog box appears.
5. Select the account you created to use as the Index Agent impersonation account from the top pane, and click **Add**.  
It should appear in the bottom pane as `DOMAINNAME\Account Name`.
6. Click **OK** and then click **OK** again to return to the Local Security Settings Console.
7. Close the Local Security Settings Console.

Your impersonation account for Indexing Service can now be entered onto the Setup tab of the Indexing Service Configuration dialog box.

## Chapter 2

# Installing, Configuring, and Using AppEnhancer Full-text Server

OpenText AppEnhancer Full-text Server for Windows is a full-text engine for AppEnhancer. This chapter describes the following:

- “Installing AppEnhancer Full-Text Server” on page 13
- “Configure idol.common.cfg for AppEnhancer Full-Text Server” on page 14
- “Creating Full-Text Database” on page 15
- “Configuring Applications to use AppEnhancer Full-text Engine in AppEnhancer Administrator” on page 15

## 2.1 Installing AppEnhancer Full-Text Server

1. Download and save the AppEnhancer full-text server installation package on your local machine.
2. Go to the folder where you saved the installation package and run `setup.exe` to start the installation process.
3. If you do not have Microsoft Visual C++ Redistributable (x64) installed, you will be prompted to install the required software before you can continue.
4. On the **License Agreement** page, select **I accept the terms in the license agreement** and then click **Next**.
5. On the **Customer Information** page, you must specify who may use the application on the computer:
  - In the **User Name** box, enter the user name for the user you want to let access the application.
  - In the **Organization** box, enter an application name.
  - To grant all users access to the application, select **Anyone who uses this computer**.
  - To restrict access to the user listed in the **User Name** box, select **Only for me**.

Click **Next** to continue.

6. On the **Destination Folder** page, you must indicate the installation folder. By default, the utility is installed in the `C:\Program Files\XtenderSolutions\Content Management\FullText Server` folder. To specify a different folder, click **Change** and select a new folder.

Click **Next** to continue.

7. On the **Ready to Install the Program** page, click **Install**.
8. Click **Finish**.
9. The status of AppEnhancer full-text server will be running from **Windows Services** after installation.

## 2.2 Configure idol.common.cfg for AppEnhancer Full-Text Server

1. Verify that the AppEnhancer full-text server is stopped before modifying the configuration. To stop the server, go to Windows Service and right-click **Stop**.
2. Navigate to AppEnhancer full-text server folder.  
The default path is: C:\Program Files\XtenderSolutions\Content Management\FullText Server.
3. Open **idol.common.cfg** with Administrator privilege and add trust IP address to **Clients** field in the following four sections:
  - AdminRole
  - IndexProxyRole
  - IndexRole
  - QueryRole

By default, only localhost can access the AppEnhancer full-text server. If AppEnhancer administrator, Web Access, REST Service or Indexing Service is installed in another server or network, a specific IP address or subnet is required. For example: below configuration allows localhost, 10.3.72.55 and all hosts from 10.2.\*.\* to access AppEnhancer full-text server.

```
//----- Authorization Roles -----//
[AdminRole]
StandardRoles=admin,servicecontrol,query,servicestatus
Clients=localhost,10.3.72.55,10.2.*.*
//SSLIdentities=
//GSSPrincipals=

[IndexProxyRole]
StandardRoles=index
Clients=localhost,10.3.72.55,10.2.*.*
//SSLIdentities=

[IndexRole]
StandardRoles=index
Actions=getpid,getstatus,indexergetstatus
ServiceActions=getstatusinfo
Clients=localhost,10.3.72.55,10.2.*.*
//SSLIdentities=
//GSSPrincipals=

[QueryRole]
StandardRoles=query,servicestatus
```

```

Clients=localhost,10.3.72.55,10.2.*.*
//SSLIdentities=
//GSSPrincipals=

```

4. **Save** and **Close** idol.common.cfg.
5. **Start** the AppEnhancer full-text server from Windows Service.

## 2.3 Creating Full-Text Database

A full-text database is a server data pool that stores indexed information. All documents submitted for indexing are assigned to a full-text database. A full-text database generally contains one category of documents.

When you configure AppEnhancer full-text engine in AppEnhancer Administrator, you must specify Full-Text Database.

1. In AppEnhancer Administrator, go to the **Server Management > Indexing Service page**, and then switch to the **Full-Text Database Management** tab.
2. Enter the **Full-Text Server URL**.
3. Enter the **Full-Text Database Name**.
4. Click **Add**.

## 2.4 Configuring Applications to use AppEnhancer Full-text Engine in AppEnhancer Administrator

1. Sign in to AppEnhancer Administrator.
2. Open the AppEnhancer application that you want to setup the AppEnhancer Full-Text configuration.



**Note:** Make sure you have created and configured the Indexing Service impersonation account before proceeding with the AppEnhancer Administrator configuration. For more information about creating the impersonation account, see [“Creating the Impersonation Account” on page 10](#).

3. Enter the **Full-Text Server URL** and the **Full-Text Database Name** for this application.



**Note:** Include the port number (default is 9100) at the end of the URL. You are recommended to use a static IP for your Full-Text Server URL.

4. Click **Apply** to close out of AppEnhancer Administrator.



## Chapter 3

# Registering Indexing Service via Component Registration Wizard (CRW) and Configuration Indexing Service

### 3.1 Registration Overview



**Note:** Before registering Indexing Service, make sure all **Service credentials** are properly introduced in Administrator.

- Open CRW from Start menu under the Open Text folder
- Select register indexing Service component in CRW and follow the prompts
- After **Finish** registration, service starts automatically

### 3.2 Configuring Indexing Service Overview

The following configurations are required before you use the Indexing Service.



**Note:** After configuration has been completed in Indexing Service located in Server Management of Administrator, you must click **Save** and restart the Indexing service in Windows Services. The service continues running while you configure the Indexing Service.

To configure the Indexing Service, you can choose from:

- Configuration through Administrator
- Configuration through Configuration file (`XtenderSolutions.IndexingService.exe.config`)



**Note:** All configurations can be overwritten from the Configuration file, with the exception of Service Credentials.

### 3.2.1 Configuration through Administrator

To configure the Indexing Service, you must complete tasks on four separate pages:

- **Storage Management** in the Environment node
- **Service Credentials** and **Settings** tabs under the Indexing Service options in the Server Management node
- **Queues** menu for each data source in the Application Management node

#### 3.2.1.1 Service Credentials tab

Navigate to **Server Management > Indexing Service** and on the **Service Credentials** tab, specify the user name and password for the **AppEnhancer** and **Impersonation** accounts.

#### 3.2.1.2 Settings tab

On the **Server Management > Indexing Service** page, switch to the **Settings** tab and configure the following fields for your indexing service:

Configuration Option	Note
Indexing Service Role	You can choose the appropriate role from the list: <ul style="list-style-type: none"> <li>• <b>Full-text &amp; OCR Both</b></li> <li>• <b>Full-text Only</b></li> <li>• <b>OCR Only</b></li> </ul>
OCR Processor number	User can add a value between 0-8. The default value is 0, which indicates that the indexing service can set this setting automatically.
Remove job from queue when finished	Users can indicate whether to remove the job from the queue when it is finished running using a True or False setting. By default, this is set to False.
<b>Full-Text</b>	
Connection Timeout	User can add a value between 0-10 minutes to specify the amount of time before the connection times out. The default value is 2.
Max Retry time	User can add a value between 0-15 to indicate the maximum time between retries. The default value is 0.
Retry Interval	User can add a value between 5-300 seconds to indicate the retry interval. The default value is 10.

### 3.2.1.3 Queues

As part of configuring the Indexing Service, you must set up queues. To access the **Queues** menu, you must select a data source under **Application Management**.



**Note:** A queue setting is unique for each data source and a queue can have the same name in different data sources.

For more information about queues, see [“Managing Queues” on page 21](#).

## 3.2.2 Configuration through Configuration file (XtenderSolutions.IndexingService.exe.config)

To configure the Indexing Service from the configuration file, go to your Indexing Service application folder and open `XtenderSolutions.IndexingService.exe.config` for editing.

Making changes to the values set in the `<indexingserviceappconfig>` tag will overwrite settings for your current Indexing Service. The only settings needed to be modified reside in the `<settingSection>` and `<dataSourceSection>` tags, either of which can be hidden.

The following is a sample `XtenderSolutions.IndexingService.exe.config` file snippet:

```
<!-- <indexingserviceappconfig>
  <settingSection>
    <settings>
      <setting name="ServiceRole" value="both" />
      <setting name="OCRProcessor" value="3" />
      <setting name="RemoveJobs" value="false" />
      <setting name="ConnectionTimeout" value="2" />
      <setting name="MaxRetries" value="0" />
      <setting name="RetryInterval" value="10" />
    </settings>
  </settingSection>
  <dataSourceSection>
    <dataSources>
      <datasource name="AppEnhancerDEMO" processingFTQueue="Full text"
processingOCRQueue="OCR" />
      <datasource name="KJDEMO" processingFTQueue="Full text" processingOCRQueue="OCR" />
      <datasource name="KOracle" processingFTQueue="Full text" processingOCRQueue="OCR" />
    </dataSources>
  </dataSourceSection>
</indexingserviceappconfig>
```

### <settingSection> properties

Field	Description
-------	-------------

ServiceRole	<p>Indicates the Indexing Service role. You can select from:</p> <ul style="list-style-type: none"> <li>• 0 or both (Full-text and OCR)</li> <li>• 1 or fulltext</li> <li>• 2 or ocr</li> </ul> <p>When setting this option, you can use either the numerical or text value.</p>
OCRProcessor	<p>Indicates the OCR processor number. Valid values can be between 0 and 16, with 0 indicating that the Indexing Service can set this automatically.</p>
RemoveJobs	<p>Designates whether to remove jobs from queues when finished with full-text server settings. You must indicate either true or false.</p>
ConnectionTimeout (minute(s))	<p>Specifies the time before the full-text server connection times out. Valid values are between 0 and 10 minutes. By default, this is set to 2.</p>
MaxRetries	<p>Indicates the interval between AppEnhancer full-text server retries. Valid values are between 0 and 15. By default, this is set to 0.</p>
RetryInterval (second(s))	<p>Indicates the maximum number of AppEnhancer full-text server retries following a failure. Valid values are between 5 and 300. By default, this is set to 10.</p>

**<dataSourceSection> properties**

If you want to add multiple queues, you can use a comma (,) to separate the queues. For example, `<datasource name="ae" processingFTQueue="AEFT,AEFT2" processingOCRQueue="AEOCR,AEOCR2" />` is acceptable.

**<appSettings> properties**

Field	Description
BulkJobCount	<p>Specifies the number of jobs that can be taken from a queue simultaneously. By default, the value is 50. This number should be determined based on your system resources as a larger value will require more resources.</p>
ServiceInterval (second(s))	<p>Specifies the interval between service runs. By default, the value is 15 seconds. The smaller this value is set to, the more system resources required.</p>

MaxThread	Specifies the maximum number of threads used for job processing. By default, the value is 10. The larger this value is set to, the more system resources required.
-----------	--

## 3.3 Managing Queues

Queues are used by Indexing Service to collect full-text and OCR jobs. A full-text job is created in a full-text processing queue when a document is submitted to the Indexing Service for full-text indexing. An OCR job is created in an OCR processing queue when a document is submitted for OCR processing.

Initially, there are no queues for the Indexing Service. Therefore, to add documents to the full-text database, you must create full-text queues. To submit OCR jobs to the Indexing Service, you must create OCR queues. The Queues tab allows you to add to the Available Queues list by creating new queues.

To initiate processing of an available queue, you must move it to the Processing Queues list. When you add a document to the full-text database, you select an Available Queue for the document. The document is not processed, however, until that queue is added to the Processing Queues list. Once a queue has been added to the Processing Queues list, the documents in the queue are either processed and added to the full-text database or processed with OCR.

You can add new, modify, and delete queues in Available Queues and move the queues from the Available Queues list to the Processing Queues list or move queues from the Processing Queues list to the Available Queues.



**Note:** Following any changes with moving queues between the lists, you must restart Indexing Service to apply the changes.

### 3.3.1 Adding a New Queue

Adding a new queue adds a new entry for the queue in the **AE\_QUEUE** table in the database for storing full-text database information for documents. You can choose whether a queue will be a full-text queue (for documents that are already in text format or for images that have been processed using OCR) or an OCR queue (for documents that must be processed to extract text from an image). These options are only available for Available Queues.

To add a new queue:

1. In AppEnhancer Administrator, click **Application Management** > *<your data source>* > **Queues**.
2. Click **New**.
3. In the **Create New Queue** dialog box, type a name in the **Name** box.
4. In the **Type** list, select a queue type. You can choose from **OCR** or **FULLTEXT**. OCR queues can be used by users submitting documents for OCR and

FULLTEXT queues can be selected when a user submits a document for full-text indexing.

5. In the **Description** box, you can enter an optional description that will appear in the Available Queues list with the queue name.
6. Click **OK**.  
The new queue is listed in the **Available Queues** list.
7. For each additional queue you want to create, repeat Step 3 to Step 6.
8. When you are finished, click **CANCEL** to close the **Create New Queue** dialog box.

### 3.3.2 Modifying a Queue

Users can only modify a queue's description in Available queues.

To modify a queue description:

1. Select a queue from the **Available Queues** list.
2. Click **Modify**.
3. In the **Modify Queue Description** dialog box that appears, update the description.
4. When you are finished, click **OK**.

The updated description appears in the Available Queues list.

5. When you are finished, click **CANCEL** to close the **Modify Queue Description** dialog box.

### 3.3.3 Deleting a Queue

If a queue is no longer required, you can delete the queue from the Available queues list. When deleting a queue, you must ensure that it is not assigned to any applications.

To delete a queue:

1. Select a queue from the **Available Queues** list.
2. Click **Delete**.
3. In the **Confirmation** dialog box that appears, confirm the deletion by clicking **OK**.

The selected queue is deleted from the Available queues list.

4. When you are finished, click **CANCEL** to close the Confirmation dialog box.

### 3.3.4 Moving Queues between Available Queues and Processing Queues lists

The Available Queues list displays the existing queues in your database and the Processing Queues list displays queues you can currently use in a job. Users can move queues between the **Available Queues** to **Processing Queues** lists.



**Note:** Following any changes with moving queues between the lists, you must restart Indexing Service to apply the changes.

To move a queue from the Available Queues list to the Processing Queues list:

1. In the **Available Queues** list, select a queue.
2. Click the right arrow button.

The selected queue is moved from the Available Queue list to the Processing Queues list.

3. Click **Save** to save the move.

To move a queue from the Processing Queues list to the Available Queues list:

1. In the **Processing Queues** list, select a queue.
2. Click the left arrow button.

The selected queue is moved from the Processing Queues list to the Available Queue list.

3. Click **Save** to save the move.



## Chapter 4

# Managing the Indexing Service

This chapter explains how to manage to Indexing Service with resubmit documents and controlling the Indexing Service.

## 4.1 Resubmitting Documents to the Indexing Service

If you have changed the full-text engine for an application, keep in mind that full-text searching does not return any documents in this application until you submit them to the Indexing Service, even if they have already been full-text indexed by the previous engine. However, if you submit documents to the Indexing Service from AppEnhancer Web Access Server result set, the number of documents that can be submitted at a time is limited.



**Note:** All documents can be resubmitted from AppEnhancer Web Access Server. Fulltext indexing should be done before modifying a document, otherwise Indexing Service won't process. When you create or modify a document, it updates modified date, and when you submit a full-text job, it updates timestamp for indexing.

## 4.2 Controlling the Indexing Service

Indexing Service can be registered as a Windows Service. Once you have completed the configuration through the Administrator or Configuration file, you must restart the service from Windows Service. Windows Service provides starting, pausing, stopping, and restarting abilities, which can also be used for Indexing Service.

## 4.3 Monitoring the Indexing Service

### 4.3.1 Viewing Indexing Service Activities

To view detailed information related to the operation of an Indexing Service, navigate to the **Monitoring > Indexing Service** node and select the component.

The following table describes the options available on the page:

Field	Description
Indexing Service	Name of the AppEnhancer Indexing Service.
DocsIndexed	Number of documents successfully indexed.
DocsIndexFailed	Number of document indexing attempts that failed.

Field	Description
DocsOCRRed	Number of documents successfully OCR processed.
DocsOCRFailed	Number of document OCR attempts that failed.
DocsResubmitted	Number of resubmitted documents successfully indexed.
LocalConfig	Whether Indexing Service uses a local configuration

### 4.3.2 Viewing the Indexing Service Footprint in Registered or Running Components

You can view the Indexing Service footprint in registered or running components within AppEnhancer Administrator.

- To view detailed information about registered components, navigate to the **Monitoring > Registered Components** section.
- To view detailed information about running components, navigate to the **Monitoring > Running Components** section.

## 4.4 Logging

The Indexing Service provides four levels of logging in Event Viewer.

- **Verbose:** Useful for debugging purposes.
- **Information:** Issues informational messages only. This is the default value.
- **Warning:** Issues warning and errors that originated from Indexing Service.
- **Error:** Issues only error messages that originated from Indexing Service.



**Note:** It is recommended to select the Warning log level to avoid consuming large amounts of storage space.

To change logging levels, open the `XtenderSolutions.IndexingService.exe.config` file for editing and modify the value under the `<switches>` option.

```
<sources>
  <!--This trace source logs Indexing Service events
    Write trace events to Windows Event Log at level of log switch only use
    this when switch set to Error or Warning else too much data in Windows
Event log
    if needs to verify root cause of any issues, please switch value to
Information
    instead or warning/Error
-->
  <source name="AEIndexingServiceTrace" switchName="EngineSwitch"
```

```
switchType="System.Diagnostics.SourceSwitch">
  <listeners>
    <add name="aeEventLogListener" />
    <!--<add name="aeFileListener1" />-->
  </listeners>
</source>
</sources>
<switches>
  <add name="EngineSwitch" value="Information" />
</switches>
```



## Chapter 5

# Indexing documents with Full-Text server

The AppEnhancer Full-Text engine is used for the initial Massive Full-Text indexing.

A substantial set of documents are re-indexed using the following methods:

- Massive Indexing mode
- Multiple Indexing Service instances and Full-Text servers

## 5.1 Massive Indexing mode

The Massive Indexing mode reduces the time required for initial document indexing process. To enhance performance, the `delayedSync` mode is selected from the Full-Text server to skip multiple status checks from the Indexing Service. After the initial indexing is complete, turn off Massive Indexing mode to ensure accurate status tracking for daily operations.

To turn on Massive Indexing mode:

1. Go to the `C:\Program Files\XtenderSolutions\Content Management\IndexingService` folder path and open `XtenderSolutions.IndexingService.exe.config` file in your preferred text editor.
2. Change the `MassiveIndexingMode` value to `true` and save the file.  

```
<add key="MassiveIndexingMode" value="true" />
```
3. Stop the Full-Text server. Take a back up of the configuration file before making any changes.
4. Go to the `C:\Program Files\XtenderSolutions\Content Management\FullText Server` folder path and open `content.cfg` file in your preferred text editor.

Modify the following settings and save your changes:

Settings	Details	Default value
Threads	If your server has N processors, set this value to N + 1. For example, set Threads = 11 for a 10-core CPU.	10
XSLTemplates	Change the value to FALSE.	TRUE
LogRequestTiming	Change the value to FALSE.	TRUE

DelayedSync	To turn on the Massive Indexing mode, change the value to TRUE. To enhance performance, the Full-Text server stores indexing requests in cache and saves the index to disk.	FALSE
MaxSyncDelay	This setting is applicable when DelayedSync value is set to TRUE. To re-index a large set of documents and maximize indexing speed, set the MaxSyncDelay value to 3600. The Full-Text server clears and sends the index to the disk every hour. The search results are available an hour after the indexing job is completed.	120
IndexCacheMaxSize	To maximize indexing speed: <ul style="list-style-type: none"> <li>• If the server has 64 GB of memory, set the IndexCacheMaxSize value to 4096000 (4 GB).</li> <li>• If the server has 16 GB of memory, set the IndexCacheMaxSize value to 1024000 (1 GB).</li> </ul>	102400

5. Restart the Full-Text server.
6. After completing the initial Massive Full-Text Indexing, stop the Indexing Service and Full-Text server. Take a back up of the configuration file before making any changes.
7. Modify the following settings for daily usage and save your changes.:
  - Change MassiveIndexingMode value to FALSE from the XtenderSolutions.IndexingService.exe.config Indexing Service configuration file.
  - Change DelayedSync value to FALSE from the content.cfg Full-Text server configuration file.
8. Restart the Indexing Service and the Full-Text server.

## **5.2 Multiple Indexing Service instances and Full-Text servers**

Multiple instances of Indexing Services and Full-Text servers are used for better performance and workload distribution.

### **5.2.1 Multiple Indexing Services**

Multiple Indexing Service instances process Full-Text jobs simultaneously. For example, if you have two pending Full-Text jobs and two active indexing service instances, each instance picks up one job to process independently. If no specific data source or queue settings are configured for different instances, the services pick up Full-Text jobs sequentially from the application queue. If specific data sources or queues are configured for individual instances, each instance processes only the jobs assigned to its configured data source and queue as defined in the configuration file.

### **5.2.2 Full-Text servers**

For environments with many documents, deploying multiple Full-Text servers enhances performance. For example, certain applications use Full-Text Server A, while other applications use Full-Text Server B. The indexing and search processes are independent across the servers, helping in distributing the workload and improving overall system performance.

