opentext™

OpenText™ AppEnhancer

Installation Guide

This document provides instructions on how to install
OpenText AppEnhancer.

Table of Contents

Chapter 1

Introduction

AppEnhancer stores, organizes, and manages documents, files, and other business-critical information, and provides fast, security-controlled access to information from Microsoft Windows or web-based clients. AppEnhancer integrates document imaging, reports management such as Enterprise Reports Management, workflow, and document management services within an easy-to-use Windows system.

1.1 AppEnhancer system

This section provides information about the various components in an AppEnhancer system.

1.1.1 Backend servers

Backend servers are major components of the AppEnhancer system. They provide a foundation for the rest of the system.

You must have at least one database and at least one document storage server location set up for use by AppEnhancer. Unless you have an evaluation copy of AppEnhancer Web Access, you must also have at least one License Server installed, with registered licenses for the components and features that your AppEnhancer system will need. For more information, see "License server" on page 18.

1.1.1.1 Database server

The database server is the operating system server that hosts the AppEnhancer database. You must have at least one database set up for AppEnhancer, which uses a database to store AppEnhancer application information, index values for AppEnhancer documents, and other important operating information.

1.1.1.1.1 Database

A database is a collection of data tables in a particular database format (such as Oracle or Microsoft SQL Server). AppEnhancer uses databases to store application information. When an application is created, details such as field definitions and security information are stored in database tables. When documents are added to an application, index information is also stored in a database table, as are the pointers to document locations.

1.1.1.1.2   Data source

AppEnhancer accesses data from a database through a data source. A data source is a composite of the database where an application stores information and the data provider used by the application (or user) to access the data. When a data source is defined, an ADO.NET data provider is configured to access the database. All of these elements in combination, where the data is stored, the format of the data stored, and the data provider used to access the data, make up the data source. Data sources are created and managed in AppEnhancer Administrator.

The concept of data source management is central to the use of AppEnhancer. The AppEnhancer system uses data sources to store the tables of index and application information that form AppEnhancer applications.

> **Note:** AppEnhancer data source and application names are used to encode disk path names. This means that all configured path names must be compatible with the host operating system (for example, Microsoft Windows). If you choose to use Chinese characters in data source, application, or path names, you must run all multibyte modules on Chinese operating systems.

1.1.1.1.3   Data source group

A data source group is a pool of data sources that share the same set of system-wide settings, including License Server settings, storage settings, component settings, and so on. All AppEnhancer system components connect to one data source group that share the same pool of data sources, License Server settings, and document storage configuration settings.

AppEnhancer Administrator stores the shared settings of a data source group into one common database—the Global Configuration Database. In a typical installation, the Global Configuration Database is the same database where the first data source created in the group resides.

You can create multiple data source groups, if you need to use different system-wide settings for different pools of data sources. However, each server or workstation running AppEnhancer components or desktop clients, such as AppEnhancer Administrator, Web Access, and so on, can be configured to connect to only one data source group. Also, each data source can exist in only one data source group. You can move a data source from one group to another, if necessary.

> **Caution**
> The Global Configuration Database should be online unless you decide to retire the entire data source group.

1.1.1.1.4   Data source management

The concept of data source management is central to the use of AppEnhancer content management products. The AppEnhancer system uses data sources to store the tables of index and application information that form AppEnhancer applications. A data source is a composite of the database where an application stores information and the data provider used by the application (or consumer) to access the data.

All AppEnhancer system components connect to a common set of data sources to form a data source group.

1.1.1.1.5   Global configuration database location

When you register an AppEnhancer content management component (such as AppEnhancer Web Access Server, Rendering Server or Web Services) with the AppEnhancer Component Registration Wizard, you must locate the Global Configuration Database used by the data source group that the AppEnhancer content management module provides services for.

If your AppEnhancer database is a MySQL or Oracle database, you must install and configure the appropriate database client software on each workstation where an AppEnhancer component needs to communicate with the AppEnhancer database.

1.1.1.1.6   Database schema

To use application user credentials when connecting to a SQL Server, PostgreSQL, or Oracle database, you must specify the data source schema. AppEnhancer does not support schema for any other database software. Recognizing individual users when accessing database tables through AppEnhancer enables database-level auditing of user activities.

Usually, when a user connects to database tables through AppEnhancer, that user is not the owner of the tables, and therefore user credentials cannot be passed to the database. If a schema name is used when accessing the database tables, the database recognizes the schema being used, identifies which tables to access when the credentials of user are passed to the database, and records the user's credentials whenever a change is made to the database.

> **Note:** If you invoke the database schema for a data source, all AppEnhancer content management products that share that data source also use the database schema.

1.1.1.1.7   Multiple data source support

Multiple data sources can be viewed and accessed simultaneously from within AppEnhancer components. Your custom client components may enable multiple data sources to be viewed and accessed simultaneously through AppEnhancer Web Services or REST Services. In this way, users can access applications located on many different data sources during a single session, if they have security rights to access those data sources.

1.1.1.1.8   OLE DB

OLE DB is a programming interface used for accessing data in AppEnhancer, and is a fundamental building block for storing and retrieving data that use Microsoft Data Access Components (MDAC). OLE DB provides flexible data architecture that offers applications, such as AppEnhancer efficient access to databases. Data is accessed through OLE DB data providers. Data providers are installed with some operating systems and their service packs or with MDAC.

If MDAC has not already been installed, many AppEnhancer setup wizards install it. MDAC installs data providers for SQL Server, Oracle, and ODBC. When you install the MySQL server or client, the MySQL Connector is installed and can then be used for access to MySQL data sources.

To use OLE DB, three components are required:

- An OLE DB consumer (such as a component of AppEnhancer)
- An OLE DB data provider (installed with MDAC)
- A DBMS server (such as Microsoft SQL Server or Oracle)

1.1.1.1.9   ADO.NET

ADO.NET is a programming interface for accessing data in a .NET application such as AppEnhancer Administrator, Web Access, and Component Registration Wizard. ADO.NET drivers provide access to Oracle and Microsoft SQL Server. The MySQL Connector/Net driver provides access to MySQL.

When you select a data provider for your data source, you are configuring an ADO.NET connection string to access the database. AppEnhancer uses the ADO.NET drivers to access your data source. It will be automatically converted to an OLE DB .NET connection string for legacy AppEnhancer desktop products to use.

1.1.1.1.10   Demonstration database

The demonstration database is an optional tool for new AppEnhancer users. This database consists of the demonstration application image files and the AppEnhancerDEMO data source. By installing the AppEnhancerDEMO data source on Microsoft SQL Server, you can perform various functions without affecting the actual system.

The demonstration applications (such as _FORMS, _RSTAMP, BUSINESS-OFFICE, CHECKS, CONADMIN, COUNTY, HR, IMAGEAPP, and so on) have already been created.

> **Note:** The AppEnhancer demonstration database is in Unicode format.

1.1.1.2   Storage server

The document indexes for AppEnhancer applications are stored in a database. The actual document files (such as scanned images, text files, and other objects) are physically stored on a storage server. AppEnhancer applications can be configured to store documents to any storage device that can be mapped by network file server, local hard disk, erasable and WORM optical media, disk-based WORM, and so on. Although all document pages are compressed before storage, they can take up a significant amount of storage space. The volume and the nature of the data (image, COLD/ERM, video, sound, and so on) require special storage considerations. Optical storage systems can be used as a cost-effective alternative to large-volume magnetic disk storage. AppEnhancer can use any optical server product that appears as a logical volume to the workstation. Although these storage systems can coexist in a single AppEnhancer system, an AppEnhancer application can be configured to use only one of these storage systems.

You can designate storage for AppEnhancer documents by using the file system path to a directory on a workstation or storage system.

Document storage is configured at the application level. Although both types of storage can coexist in your environment, you must designate either one or the other of these storage types for each AppEnhancer application.

1.1.1.2.1   File system storage

In a file-based storage system, you can store AppEnhancer documents by designating any file system path to a directory on a workstation or storage system accessible to your workstation.

1.1.1.2.1.1   Secure paths

AppEnhancer requires you to designate a secure share as the write path for storing documents and annotations for all Software Retention Management applications. It is also recommended that you modify your existing applications to use secure paths. Otherwise, any Windows user who has access to the file share location can delete AppEnhancer files. After you create the necessary secure paths, set up credentials in AppEnhancer Administrator to enable AppEnhancer clients to access the paths.

1.1.1.2.1.2   Document write paths

For file-based storage systems, document storage is configured for a particular application by setting document write paths. Write paths are used to instruct AppEnhancer where to store documents, annotations, and optical character recognition (OCR) output for a particular AppEnhancer application. A simple example of a write path is a directory name on a local hard drive of a workstation (`C:\Program Files\AppEnhancer\Content Management\INVOICES\`).

> **Note:** AppEnhancer uses data source and application names to encode disk path names. This requires that all configured path names are compatible with the host operating system. If you choose to use Chinese characters in data source, application, or path names, you must run all multibyte modules on Chinese operating systems.

The storage location for documents in AppEnhancer applications is not limited to local hard drives.

1.1.1.2.1.3   Dual write paths

The dual write path feature permits redundant content file storage from two different geographic locations. You can configure two different types of dual write paths: parallel storage and remote/local.

The following tables describe each type of dual write path and indicate what happens when document files are stored and retrieved by using each type of dual write path:

Dual write paths	Description	Storage	Retrieval
Parallel storage	Enable automatic backup of AppEnhancer document files for disaster recovery.	Files are written to both the primary and secondary paths simultaneously. If either write operation fails, the entire operation fails.	Files are retrieved from the primary path. If the files are not available from the primary path, they are retrieved from the secondary path.

Dual write paths	Description	Storage	Retrieval
Remote/local	Can improve performance over a geographically diverse WAN.	If the Copy to Local option is enabled, files are written to a local file first. After the write operation is completed, the local file is copied to remote storage. Otherwise, files are written only to remote storage.	If the local file does not exist or is older than the remote file, the remote file is retrieved. If the Copy to Local option is enabled, remote files are copied to local storage and then the local file is retrieved.

Parallel storage dual write paths

Parallel storage dual write paths enable automatic backup of AppEnhancer document files for disaster recovery. When you attempt to create an AppEnhancer document, AppEnhancer writes the file to both the primary and secondary paths simultaneously. If either write operation fails, the entire operation fails.

When you try to retrieve an AppEnhancer document, AppEnhancer tries to read the file from the primary path. If the file is not available from the primary path, it is read from the secondary path instead.

Ensure the following when you configure parallel storage dual write paths:

- The primary path is where standard application files are stored (bin files, annotation files, and OCR files). It can be the same as the application write path configured in AppEnhancer Administrator, its parent folders, its root drive (such as C:), or UNC path. Files stored under the primary path are under the control of the dual write path feature.

- The secondary path is the backup path. Anything stored under the primary path will be backed up automatically under the secondary path in the same directory structure.

  📄 **Note:** Before configuring the dual write path feature in AppEnhancer Administrator, you must manually synchronize the primary and secondary paths. Ensure that any files in each path also exist in the other path. After all files are configured, the files are modified and written to both locations at the same time. Hence, files are always synchronized.

If either the primary path or the secondary path fails, AppEnhancer can still read files from the other path. However, to protect the integrity of the data, AppEnhancer does not update existing files or add new files after a primary or secondary path has failed. Administrators can manually make changes to keep the system running.

Remote/local dual write paths

Remote/local dual write paths can improve performance over a geographically diverse WAN, if you use the Copy to Local option. In this case, when you try to

---

create an AppEnhancer document, AppEnhancer writes a file to the local path first and then copies the file to remote storage. If the Copy to Local option is not enabled when you attempt to create an AppEnhancer document, AppEnhancer writes only to the remote path.

When you try to retrieve the document with the Copy to Local option enabled, AppEnhancer compares the last modification time stamps of both the remote file and the local cache file (if it exists). If the remote file is newer, the local cache file is automatically updated. Therefore, synchronization is automatically maintained.

If the Copy to Local option is not enabled, any changes to remote files will not automatically update local cache files. New or newly modified files must be manually copied to local cache, if necessary. When you attempt to retrieve a document, if the file exists in the local cache, AppEnhancer reads the file from the local cache.

When you configure the remote/local dual write path ensure that the following conditions are in place::

- The remote path is where standard application files are stored (bin files, annotation files, and OCR files). This path can be the same as the application write path configured in AppEnhancer Administrator, its parent folders, its root drive (such as C:), or UNC path. Files stored under the remote path are under the control of the dual write path feature.

- The local path is where local cache files are located. Any file accessed under the remote path will be cached under the local path in the same directory structure. Note that because the dual write path configuration is configured for each data source rather than for each workstation, it is recommended that you use a mapped drive as the local cache path. This way, each remote site or remote workstation can configure its own local cache location through a different drive mapping. This also enables workstations on the remote site where the data storage is located to skip the overhead of caching files by not mapping the cache drive. The AppEnhancer Web Access Server and Web Services Server run as background services. Therefore, to prevent inefficiency and difficult drive-mapping, the server should always be located at the primary site where the remote path is located.

The Copy to Local option should always be enabled for automatic caching and synchronizing of remote files that have been accessed. If this option is not checked, AppEnhancer does not automatically copy the files to the local cache and does not check remote file time stamps for synchronization.

Dual write path limitations

The dual write path is designed to enhance the availability or performance of an AppEnhancer system. It is not a complete storage management system. As a result, the dual write path lacks the capabilities of automatic backup, restore, synchronization, and garbage collection. Such operations require the manual intervention of system administrators.

Dual write path entries are configured through AppEnhancer Administrator and applied to all AppEnhancer content management components.

Partial path mappings are supported. For example, you can map \\server1 to \\server2 so that each time AppEnhancer reads or writes a file on server1, AppEnhancer can read or write alternative files on server2. All the paths configured as dual write paths must be Windows file system paths (local or UNC). Other types of paths that are typically supported by AppEnhancer are not supported by the dual write path feature. AppEnhancer Administrator also does not verify whether or not the path you entered in the configuration is valid.

1.1.1.2.1.4 Creating drive mappings

Paths also enables you to map paths to drive letters, so that drive letters can be used when specifying AppEnhancer Web Access Server or Web Services session cache paths, or AppEnhancer Rendering Server cache paths. When you map a drive letter to a path, all workstations that access that path through AppEnhancer Indexing Service, Web Access, or Web Services use the same drive letter. Drive mappings configured here override any conflicting drive mappings that might exist on a workstation connecting to this path. It is recommended that you only use the drive mapping capability to accommodate existing paths in your database that already include a drive letter. The drive mapping capability is intended for use only with legacy systems with existing mapped paths. In general, when specifying paths in the AppEnhancer products, it is best to use UNC paths.

> **Note:** When an administrator creates a new application or modify an application, the administrator can change the document write path to a path, as specified in **Paths**. If the path in **Paths** is changed later, the path configured in the application does not change.

1.1.1.2.1.5 Rendering cache

The AppEnhancer Rendering Server Cache Location settings dictate the path where rendered files are cached for repeated access. If the AppEnhancer Web Access Server and Rendering Server are on the same workstation, the AppEnhancer Rendering Server Cache Location can be either a local drive letter path or an UNC path. If the AppEnhancer Web Access Server and Rendering Server are not on the same workstation, the AppEnhancer Rendering Server Cache Location must be an UNC path, because the Cache Location must be available to all AppEnhancer Rendering Server workstations.

The data source for the AppEnhancer Rendering Server database points to a database containing tables that are used to manage the rendering queue for the AppEnhancer Rendering Server. You can also set up a data source that points to an existing AppEnhancer content management database and add the AppEnhancer Rendering Server queue tables to that database. For high rendering queue traffic, you may want to set up a database specifically for rendering queue management. The same databases and database versions supported for the AppEnhancer content management database can be used for the AppEnhancer Rendering Server database. You can also use a schema user with the database, if needed.

1.1.1.2.1.6   Microsoft Azure File service

Azure File service can be used as a document server for your AppEnhancer document storage. Azure File service enables you to save and retrieve files as document write paths or secure paths.

To save AppEnhancer documents to Azure File service, you must first configure an Azure Files server in AppEnhancer Administrator. On the Storage Management page in AppEnhancer Administrator, click **Azure Files Service**. Enter the Azure File server name, storage account name, and storage account key, and add the Azure File paths.

Azure File paths are supported only as Application Path and Secure Path Root.

1.1.1.3   License server

The OpenText license module manages licensing for all AppEnhancer content management products. The license module is used by these products to validate licensing options and to monitor users on the system.

Unless you have an evaluation copy of an AppEnhancer component, you must have a valid license server installed on at least one server, with registered licenses for the components and features that your AppEnhancer system requires. You can use an evaluation license with AppEnhancer Web Access, and Image Capture Classic. For AppEnhancer Image Capture Classic, the evaluation time begins when the AppEnhancer Image Capture Classic is installed. When the evaluation credentials expire (in 30 days), you must install the License Server and enter license information to continue to use your AppEnhancer system.

Connections to the License Server for each data source are configured on the **License Servers** node in AppEnhancer Administrator.

1.1.1.3.1   AppEnhancer user license

The following table describes the AppEnhancer User license information:

Attribute Name	Description
Users	Maximum number of concurrent AppEnhancer users. If users perform full-text operations, they also count against the Full-Text Clients licensing value. If users are connected as read-only users, they count against the Max Users Read-Only licensing value.
OCR	Ability to perform OCR operations. This feature is either enabled (yes) or disabled (no) for the entire AppEnhancer system.

Attribute Name	Description
Full-Text Clients	Maximum number of AppEnhancer users who can concurrently perform full-text operations, such as full-text indexing or full-text queries.

1.1.1.3.2 AppEnhancer software retention management

For the AppEnhancer Software Retention Management feature, the license specifies that the AppEnhancer Software Retention Management feature is enabled. This feature enables users with the appropriate privileges to file documents for software retention, place documents on retention hold, and remove retention holds using AppEnhancer Web Access. Authorized users can also create, edit, and delete retention policies for use with AppEnhancer.

1.1.1.3.3 AppEnhancer Web Access

The following table describes the AppEnhancer Web Access license information:

Attribute Name	Description
Read-Only Users	The maximum number of concurrent AppEnhancer Web Access read-only users. These users can only retrieve and print documents. This value applies only to Public Access Licenses (PAL) for users with the PAL User privilege in AppEnhancer Administrator.

1.1.1.3.4 License groups

License Groups enables you to control which licenses are allocated to specific users or databases. If any license groups have been created in the License Server, you can specify their use for individual users or for individual AppEnhancer databases. The following general guidelines apply:

- If you specify a license group for an individual user, each time that user logs on to an AppEnhancer component on any server for any AppEnhancer database, a license from that license group is used.

- If you specify a license group for an individual AppEnhancer database, each time any user logs on to an AppEnhancer component on any server for that AppEnhancer database, a license from that license group is used.

However, the license group for a user overrides the license group for the AppEnhancer database.

Consider assigning license groups to users rather than to databases, to minimize the number of licenses in use for each user. If license groups are assigned to AppEnhancer databases, each time a user logs in to a different data source, the user uses a separate license.

License groups can be created for the following license attributes:

- Web Access PAL Users
- Image Capture Users
- Image Processing Users
- Full-Text Users

1.1.1.3.5   Additional license server features

The License Server includes some additional features:

- Log file: A log file, `license_mmm_YYYY.log`, e.g., **license_nov_2017.log**, is automatically generated and stored in the default License Server install directory (default location is `C:\Program Files\XtenderSolutions\Content Management\License Server`). A new log file is generated at the beginning of each month.
- Administrative functionality.

1.1.2   AppEnhancer Administrator

AppEnhancer Administrator enables you, as system administrator, to:

- Configure data sources, licensing, storage, and other settings.
- Configure application and security in data source.
- Configure each component of AppEnhancer.
- Configure user settings.
- Monitor running status of AppEnhancer components and license usage.
- Perform other administrator operations.

The AppEnhancer Administrator deployment depends on the Microsoft .NET Framework. AppEnhancer Administrator is compatible with and requires Internet Information Services (IIS). IIS is a standards-based, transactional web server that is fully integrated with supported Windows Server software. AppEnhancer Administrator runs as an application on IIS.

The **Desktop Credential** node in AppEnhancer Administrator enables you to configure credentials to enable clients to access secure storage paths for an AppEnhancer application.

AppEnhancer Administrator automatically creates the database table structures used by the AppEnhancer system. AppEnhancer Administrator also enables you to manage connections to License Server and paths used by AppEnhancer system resources. AppEnhancer Administrator is also used to monitor remote AppEnhancer system components.

AppEnhancer Administrator has an AppEnhancer Web Access node that enables you to configure the following settings that apply to AppEnhancer Web Access:

- Service credentials for authentication
- AppEnhancer Web Access file types
- AppEnhancer Web Access email settings

In addition, you can use AppEnhancer Administrator to configure the settings for Web Services.

You can also configure AppEnhancer Rendering Server settings, including AppEnhancer Rendering Server service credentials and AppEnhancer Rendering Server cache location and data source.

AppEnhancer Administrator depends on the Microsoft .NET Framework, so the AppEnhancer Administrator installation wizard detects whether the framework already exists on the AppEnhancer Administrator server, then installs the framework if needed. AppEnhancer Administrator provides access to many administrative functions. For example, creating applications, users and groups, configuring user settings and so on. System security on the application, functional, and document levels is configured through web access user settings, group profiles, and the Document Level Security feature. AppEnhancer Administrator is also used to configure security mapping before you perform a data migration using the AppEnhancer Migration Wizard. Customized specifications can be created to customize the import of specific index information using the AppEnhancer Auto Index Import Wizard, Key Reference Import Wizard, or Index Image Import Wizard.

You can track actions in the AppEnhancer system using **Audit Trail** in AppEnhancer Administrator. For example, the creation and deletion of documents and applications can be tracked if you enable audit trails for those items.

1.1.3   AppEnhancer servers

📄 **Note:** If your AppEnhancer database is a MySQL or Oracle database, you must install and configure the appropriate database client software where an AppEnhancer component needs to communicate with the AppEnhancer database. This applies to each AppEnhancer server.

1.1.3.1   AppEnhancer Indexing Service and full-text server

The AppEnhancer Indexing Service provides full-text search functionality and OCR processing for AppEnhancer systems. You can submit an image to the AppEnhancer Indexing Service for OCR processing or you can add text information to the AppEnhancer Indexing Service for full-text indexing. You can process scanned images using OCR, then add the resultant text version to the full-text server through Indexing Service, enabling users to search on keywords that might be located anywhere on the document. The AppEnhancer Indexing Service runs as a service. The full-text server is where the AppEnhancer Indexing Service stores full-text index information.

If you would like users to be able to search the full-text of AppEnhancer documents, you must purchase the full-text option for AppEnhancer, and then install and

configure at least one AppEnhancer Indexing Service. Users can add the full-text of documents with formats supported for full-text indexing to a full-text server and then search that server to retrieve documents.

The AppEnhancer Indexing Service processes full-text indexing requests and outputs the indexed text to the location that you specify when configuring the AppEnhancer application.

1.1.3.2   AppEnhancer Web Access server

AppEnhancer Web Access centralizes document access on the web server. The components that make up the AppEnhancer Web Access system fall into three categories: AppEnhancer Administrator components, AppEnhancer Web Access Server components, and AppEnhancer Rendering Server components

1.1.3.2.1   AppEnhancer Web Access system

A typical AppEnhancer Web Access deployment usually consists of:

- One or more AppEnhancer Web Access Servers

- One or more AppEnhancer Rendering Servers

- AppEnhancer Administrator

- The additional servers required to support the AppEnhancer system (such as the AppEnhancer database server, the AppEnhancer Indexing Service, the full-text server, and the License Server)

- The content storage repository

- A connection through the Internet or an intranet to a number of clients using a web browser

AppEnhancer Administrator enables you to create data source groups, where all components and modules connecting to the data source group share the same pool of data sources, License Server settings, and document storage configuration settings.

All AppEnhancer Web Access components within a data source group share processing responsibilities. For example, multiple AppEnhancer Rendering Servers might be registered for a particular data source group. The AppEnhancer Rendering Server settings configured in the AppEnhancer Administrator for that data source group apply to all AppEnhancer Rendering Servers.

In each AppEnhancer data source group, you must have one AppEnhancer Web Access Server to enable browser-based access to AppEnhancer documents. At least one AppEnhancer Rendering Server is required. You can choose to install AppEnhancer Rendering Server on the same server as the AppEnhancer Web Access Server, or you can set up a separate AppEnhancer Rendering Server or Servers.

1.1.3.2.2 AppEnhancer Web Access environment components

The following components must be available in the AppEnhancer Web Access environment before you deploy AppEnhancer Web Access:

- AppEnhancer Administrator must be installed.
- The AppEnhancer system must have a data source group configured, with at least one data source.
- Applications must be created to enable storage of documents.
- User accounts must be configured and users must be assigned AppEnhancer permissions to enable them access to functionality within AppEnhancer Web Access.
- Microsoft Internet Information Server (IIS) must be installed on the server to be used as AppEnhancer Web Access Server.

1.1.3.2.3 AppEnhancer Web Access server deployment

The AppEnhancer Web Access deployment depends on the Microsoft .NET Framework. AppEnhancer Web Access is compatible with and requires IIS. IIS is a standards-based, transactional web server that is fully integrated with supported Windows Server software. AppEnhancer Web Access runs as an application on IIS.

AppEnhancer Web Access will be deployed in an AppEnhancer directory associated with the selected website. After you configure AppEnhancer Web Access, the default AppEnhancer Web Access web pages are used in conjunction with information from your AppEnhancer database to retrieve and display requested documents. Without any custom development on your part, you can enable remote users to add, delete, view, edit, annotate, and print AppEnhancer documents using a web browser.

AppEnhancer Web Access has been tightly integrated with Microsoft .NET Framework, enabling AppEnhancer Web Access to use the latest and most innovative technologies from IIS web server product.

1.1.3.3 AppEnhancer Rendering server

AppEnhancer Rendering Server renders all documents, including images and HTML files in the main frame of the Web Access viewer, images in the thumbnail panel of the Web Access viewer, and all results, including export, email, and print functions. For small systems, you can choose to install AppEnhancer Rendering Server on the same server as AppEnhancer Web Access Server. If you are deploying an enterprise system, you can also choose to install one or more AppEnhancer Rendering Servers on separate servers to provide more efficient processing of client requests.

If installed separately from AppEnhancer Web Access Server, AppEnhancer Rendering Server offloads processing tasks from AppEnhancer Web Access Server. This enhances AppEnhancer Web Access Server performance, especially when many users access the server simultaneously. This flexible deployment model is designed

to offer the greatest level of scalability and performance as the number of concurrent users grows.

AppEnhancer Rendering Server uses real-time rendering for the main frame content of the viewer. To be specific, WCF technology is used to achieve the real time rendering. If you choose to install more AppEnhancer Rendering Servers on separate servers, the rendering servers will accept real-time rendering requests in turn. In the AppEnhancer Rendering Server Installation location, there is a configuration file named `WxRenderServiceHost.exe.config` that you can open and configure. For example, you can configure the WCF port number, WCF timeouts, maxConcurrentSessions, maxConcurrentCalls, and so on. AppEnhancer Rendering Server uses both a rendering cache and rendering database to manage the thumbnail rendering request queue. After a thumbnail of a file is rendered, it is cached in the rendering cache location. Information relating to cached files and the current rendering request queue is stored in the rendering database. The AppEnhancer Rendering Server database tables can be added to the AppEnhancer database, or you can create a separate database on any of the AppEnhancer Web Access supported database platforms.

If a Web Access user requests to render a document, the rendering results are cached under the rendering cache folder, and the folder is valid throughout the Web Access log on session. After the Web Access session is logged out, the cached files are removed.

When you run AppEnhancer Rendering Server Setup Wizard, the AppEnhancer Render Service is installed. You configure AppEnhancer Render Service settings through AppEnhancer Administrator. After you install AppEnhancer Rendering Server, AppEnhancer Component Registration Wizard must be run to associate the AppEnhancer Rendering Server with the data source group

After making changes to data source or AppEnhancer Rendering Server account credential settings, running AppEnhancer Component Registration Wizard again registers those changes with the data source group.

1.1.3.4   AppEnhancer Web Services

AppEnhancer Web Services provides a standardized web services interface to AppEnhancer, using the same database and AppEnhancer Rendering Server components that are used by AppEnhancer Web Access. The manufacturer-supplied components that support an AppEnhancer Web Services system fall into four categories: AppEnhancer Web Services environment components, the AppEnhancer Web Services component, and AppEnhancer Rendering Server components. AppEnhancer Web Services deployments also depend on customer-supplied client or client/server components that use AppEnhancer Web Services to communicate with the rest of the AppEnhancer system.

AppEnhancer Web Services uses two installation wizards to deploy manufacturer-supplied components:

- AppEnhancer Web Services Setup Wizard

- AppEnhancer Rendering Server Setup Wizard

AppEnhancer Web Services configuration is performed by using the AppEnhancer Administrator.

In addition, services on the AppEnhancer Web Services Server and Rendering Server, including AppEnhancer Render Service, provide the image processing service required to respond to requests from AppEnhancer Web Services clients and to help keep AppEnhancer Web Services Server running efficiently.

1.1.3.4.1 AppEnhancer Web Services environment components

The following components are required in the AppEnhancer Web Services environment before you deploy a custom AppEnhancer Web Services solution:

- AppEnhancer Administrator must be installed.

- The AppEnhancer system must have a data source group configured, with at least one data source.

- AppEnhancer applications must be created to enable storage of documents.

- Users must be configured and assigned AppEnhancer rights to enable them access to functionality within AppEnhancer Web Services.

- IIS or AppEnhancer Web Access Host must be installed on the server to be used as the AppEnhancer Web Services Server.

1.1.3.4.2 AppEnhancer Web Services deployment components

A typical AppEnhancer Web Services deployment consists of:

- One AppEnhancer Web Services Server, which provides an HTTP interface to AppEnhancer Web Services for the customized client components

- One or more AppEnhancer Rendering Servers

- AppEnhancer Administrator and the additional servers required to support the AppEnhancer system (such as the AppEnhancer database server, the License Server, and AppEnhancer Indexing Service and full-text database server)

- Content storage repository

- An HTTP or HTTPS connection through the Internet, an intranet, or a local area network from a number of clients or a server application

In each AppEnhancer data source group, you must have one AppEnhancer Web Services Server to enable access to AppEnhancer Web Services and access to AppEnhancer system components. Web Clients can connect to a custom web server application that sends HTTP requests to AppEnhancer Web Services, or desktop clients can connect through HTTP or HTTPS to the AppEnhancer Web Services Server. If calls are made to the GetRenderingStatus(), RenderPageVersion(), or RenderPageVersionAsync() methods by custom client or server components accessing AppEnhancer Web Services, at least one AppEnhancer Rendering Server is required. You can choose to install AppEnhancer Rendering Server on the same

server as the AppEnhancer Web Services Server, or you can set up a separate
AppEnhancer Rendering Server or Servers.

1.1.3.4.3   AppEnhancer Web Services mode

AppEnhancer Web Services is installed as a separate application and has no
document display functionality. AppEnhancer Web Services cannot share sessions
between AppEnhancer Web Access and Web Services. AppEnhancer Web Services
must have the AppEnhancer license (from the User license).

1.1.3.4.4   AppEnhancer Web Services components

When you run AppEnhancer Web Services setup, you install AppEnhancer Web
Services, Web Services Test Console, and Component Registration Wizard. You can
choose to set up your AppEnhancer Web Services Server on a server already
running IIS. You can also choose to install the AppEnhancer Web Services client
code samples.

AppEnhancer Web Services, Component Registration Wizard, and Web Host require
the Microsoft .NET Framework.

The following table describes the AppEnhancer Web Services components:

Component	Description
AppEnhancer Web Services	AppEnhancer Web Services installation installs the AppEnhancer Web Services interface on your Internet Information Server or AppEnhancer Web Access Host. The AppEnhancer Web Services API interface provides API-level access to a subset of AppEnhancer functionality, including session management, data source and application management, search management, document management, page version management, document index management, full-text search management, and batch processing management.
AppEnhancer Web Services Test Console	AppEnhancer Web Services installation installs a C#-based test console that can be used to validate AppEnhancer Web Services configuration and functionality.
AppEnhancer Web Services Client Code Samples	AppEnhancer Web Services installation can install client code samples that are used as a reference during AppEnhancer Web Services development.

Component	Description
AppEnhancer Component Registration Wizard	Running AppEnhancer Component Registration Wizard on the AppEnhancer Web Services Server registers the server with the AppEnhancer data source group. After installing AppEnhancer Web Services, run AppEnhancer Component Registration Wizard to add AppEnhancer Web Services to the data source group. AppEnhancer Component Registration Wizard must be run once for each component to be registered. Running AppEnhancer Component Registration Wizard after making changes to data source or service credential settings registers those changes.

1.1.3.5 AppEnhancer REST Services

AppEnhancer REST Services are a set of RESTful web service interfaces that interact with the AppEnhancer platform. AppEnhancer REST Services provide you with high efficiency and simplicity in programming, making all services easy to use. These advantages make AppEnhancer REST Services the best choice for next-generation applications and mobile applications to interact with the AppEnhancer platform. AppEnhancer REST Services identify resources by Uniform Resource Identifiers (URIs) and define specific media types to represent resources and drive application state transfers by using link relations. These services use a limited number of HTTP standard methods (GET, POST, PUT and DELETE) to manipulate these resources over the HTTP protocol. AppEnhancer REST Services support only the JavaScript Object Notation (JSON) and XML formats for resource representation. JSON is a lightweight data interchange format based on a subset of the JavaScript Programming Language standard.

1.1.4 Clients and utilities

1.1.4.1 Index Image Import Wizard

If images, text, or foreign file format files and associated index data exist in another system, you can import them into AppEnhancer with the AppEnhancer Index Image Import Wizard. No manual document indexing is required. For more information, see the *OpenText AppEnhancer Administration Guide*.

1.1.4.2   Key Reference Import Wizard

The AppEnhancer Key Reference Import Wizard is used to import key reference information into the AppEnhancer system. When you perform a key reference import, AppEnhancer imports the data into a key reference table. Then, users adding documents can use the [TAB] key to automatically populate AppEnhancer indexes using the imported data from the key reference table.

The key reference table is a central holding area for index information and operates on a static data basis, which means that records remain in the table even after they are used. If changes occur (such as an address or name), then those changes are immediately reflected across the entire application, because any documents in the application indexed using key reference data obtain their index information from the key reference table. For more information, see the *OpenText AppEnhancer Administration Guide*.

1.1.4.3   Auto Index Import Wizard

The AppEnhancer Auto Index Import Wizard is used to import auto index information into the AppEnhancer system. When you perform an auto index import, AppEnhancer imports the data into an auto index table. Then, users adding documents can use the [F7] key to automatically populate AppEnhancer indexes by using the imported data from the auto index table.

The auto index table operates on a use-once-and-discard basis; when an item is extracted from the table, it is automatically deleted. This feature is used when initially creating documents, so that all unindexed records can be tracked.

1.1.4.4   Migration Wizard

The migration feature of the AppEnhancer Migration Wizard moves or copies AppEnhancer application definitions, application import specifications, application data, user lists, and group lists from one data source to another data source. Only users who have been given the Administrator privilege in AppEnhancer Administrator can perform migrations.

> **Caution**
> Users with the Administrator privilege must create the destination application first in the AppEnhancer Administrator before using the Migration Wizard.

The migration feature can be used to distribute information to other sites or to scale AppEnhancer solutions to larger or smaller databases. You can choose to migrate all AppEnhancer application data or just a subset of the information. Several applications can be migrated without exiting the wizard to change the source database. You can automate application migration using the save and load settings features in addition to using extensive command line options.

1.1.4.5 Retention Management

AppEnhancer Retention Management Administration enables you to manage AppEnhancer Software Retention Management.

For AppEnhancer Software Retention Management, you specify the AppEnhancer retention policy to use for an AppEnhancer application using the Retention Management Configuration Utility Wizard. The wizard also lets you create new retention policies, as well as edit and delete existing retention policies.

1.1.4.6 Web client

AppEnhancer Web Access enables users to view, edit, and print AppEnhancer documents using a web browser, without any additional software or plug-in installation.

In Web Client, users can scale documents using preset zoom functions. Users can generate printer-friendly views to facilitate document printing. In addition, if foreign file rendering has been enabled in the AppEnhancer Rendering Server settings in AppEnhancer Administrator, users can view foreign files as HTML.

AppEnhancer Rendering Server renders images and thumbnails for inclusion in the HTML pages that display in web client. In web client, document pages must be converted to a web-viewable format before they are transmitted to the client.

- Users can retrieve documents using a variety of search methods, including multi-application searches, index term searches, document property searches, full-text searches, wildcard searches, list of values searches, and expression searches. Dialog boxes aid the user in creating the necessary syntax to perform list of values and expression searches.

- When a query is executed, the list of results appears in the AppEnhancer Web Access Result Set. From here, in addition to opening documents, users can choose to sort results, toggle through results pages, and enable or disable document thumbnails.

- Thumbnails representing the pages of the currently viewed document can be displayed on the **Document View** page.

- When a document is opened from the Result Set, users can view document indexes by using the Document Index view and can (with appropriate privileges set in AppEnhancer Administrator) modify indexes as needed.

- When modifying indexes, a user can employ dual data entry, multiple indexes referencing a single document, auto indexing, and key reference indexing in applications configured with those functions.

- The AppEnhancer Web Access check-in or check-out feature provides version control. Users can check documents out after retrieving them. After a user checks out a document, other users can view the document only in read-only mode, preventing simultaneous revisions and loss of work.

- Documents can be submitted to AppEnhancer Indexing Service for full-text indexing from the **Document View** page.

The following table lists the supported AppEnhancer Web Access document types:

Document Type	Comments
Image	Preset zoom ratios are provided. Annotation viewing is supported.
Foreign File	Presented as a link, which the user can click to download the file. If a file association is configured in AppEnhancer Administrator and the application associated with the file type exists on the client workstation, the document opens with the native application in the AppEnhancer Web Access Document window. If you enable foreign file rendering in AppEnhancer Administrator, foreign files are converted to HTML format, and included graphics are rendered as GIF or JPEG.
OLE Object	AppEnhancer Web Access does not support embedded OLE objects. To add file types not supported by AppEnhancer Web Access, add the files as foreign files in AppEnhancer Web Access.
PDF File	PDF files, like foreign files, are presented as links, which the user can click to download the file (if Acrobat Reader is installed, clicking the link opens the PDF file in the browser). If a user has downloaded the optional Adobe component, PDF files display in their native format in the AppEnhancer Web Access Document window.

1.1.4.7   Image Capture

AppEnhancer Image Capture is a batch scanning module that scans images into AppEnhancer batches for indexing. AppEnhancer Image Capture enables users to create and append batches. A Batch List for each application is shared between AppEnhancer Image Capture and Web Access, so that AppEnhancer Image Capture users can add pages to batches in the list while AppEnhancer Web Access users are creating indexed documents from other batches in the list.

1.1.4.8 AppEnhancer Indexing Service components

AppEnhancer Indexing Service enables full-text search and an OCR feature. Full-text and OCR jobs are sent through WebAccess.NET and configuration for Indexing Service is performed through AppEnhancer Administrator. For more information about the Indexing Service, see *OpenText AppEnhancer Administrator Guide (EAXCORE-AGD)*.

1.1.5 Organization of content

The highest level of organization in AppEnhancer is the application, which you design. Applications contain documents. Documents consist of one or more pages. Each document has index fields, which act as a label for the document. Pages consist of one or more page versions.

The following descriptions of essential AppEnhancer terms will help you understand the AppEnhancer system and its functional components, such as applications, indexes, documents, and pages, and how they interact with each other.

1.1.5.1 Applications

An AppEnhancer application is an index-driven data storage structure where documents can be stored and retrieved. This application is based on an index that is composed of one or many fields.

You can create applications through AppEnhancer Administrator. During application creation, you can provide a name to the application and set up one or more index fields for the application. Each data source can support up to 2048 different applications.

Each time a document is stored within an application, you must enter the index information for that particular document into the index fields. After the document is saved, it is compressed and stored in the document write path or staging path as a BIN file. The record of index information for the document is stored in the AppEnhancer database with a pointer to the document location. Documents can then be retrieved through searches that query corresponding index information.

1.1.5.2 Indexes

An AppEnhancer index contains a group of fields where descriptive information pertaining to documents can be stored. This group of field definitions is used by AppEnhancer when storing index information within an application.

The index is the central component of AppEnhancer, so an understanding of indexes is an integral part of understanding the applications. The index enables users to organize and search through stored documents efficiently. Documents can be stored in the AppEnhancer system in any order, and yet are easily retrievable, because every stored document has an index record attached to it. You can search all index records within an application, and retrieve relevant documents.

The AppEnhancer system administrator designs the index for an AppEnhancer application. The administrator decides what index information will be requested from the user and the format for that information. When an index is created in AppEnhancer Administrator, the administrator designates how many descriptive entries to request for each stored document, what type of information will be used to identify the documents, how much space to provide for each entry, and how the information for a particular entry will be entered.

> **Note:** The administrator can create up to 64 index fields for each application. However, three or four indexes are sufficient for most applications. The more indexes you have in an application, the greater amount of data entry will be required.

For example, a human resources application used to store information pertaining to company employees might have the index definition described in the following table:

Field	Definition
NAME	A text field with a length of thirty-five characters
SSN	A social security number field with a format of nnn-nn-nnnn
DEPARTMENT	A user-defined list field, listing all departments
DATE OF HIRE	A date field with a format of mm-dd-yyyy

AppEnhancer components use the fields specified during application creation to build an Index view that requests exactly the information needed from the user whenever a document is added to an application. This view provides a space for the user to enter a value for each of the fields defined in the index definition.

All of a document's entries are stored as a group in the application's index, with a pointer to the document's stored location. This group of values is referred to as the document index record. A record for the human resources application for a document related to Mr. John Doe, for example, would have four fields in it, with each of the following pieces of information contained in a separate field: John Doe, 000-00-0000, Accounting, and 01-07-1994.

1.1.5.3 Documents

A document is a page or group of pages stored in an application and identified by index information. Each page of a document is comprised of a single object, such as a scanned image file or a multipage PDF document. To create a new document, users add an object to an application and attach index information to it. Subsequent objects can be added as additional pages of the same document.

A document can be as small as one page, or can contain thousands of pages. Different data types can be stored as pages within a single document because AppEnhancer supports multiple object types. For example, a patient's document can include a scanned photograph, admission forms, the text of a doctor's report, and an x-ray. Any combination of objects can be stored within a single document.

When a document is stored in an application, you must enter data to fill each of the document's index fields.

To retrieve the document later, you search for some portion of the index information that corresponds with the document. When a search request is made in AppEnhancer components, all records of index information that match the information requested are displayed in a result set.

You can retrieve any document by selecting its record from the result set. All of the pages within a document can be processed as a single unit: printed, mailed, or exported. Pages can also be processed individually. Several documents can be printed at the same time by selecting them from the result set and choosing the appropriate command. You can modify documents and document indexes at any time if you have the proper security privileges.

The various data types that can be added as documents or pages in AppEnhancer include the following:

- Images:

  - Grayscale

  - Color

  - Bitonal

- Objects:

  - Microsoft Word files

  - WordPerfect files

  - Microsoft Excel graphs/spreadsheets

  - Lotus graphs/spreadsheets

  - Microsoft Video recordings

  - Microsoft Sound recordings

  - HTML (Hypertext Markup Language) files

- RTF (Rich Text Format) files

- Adobe PDF (Portable Document Format) files

- Objects from other ActiveX embeddable applications

• Enterprise Report Management (ERM) files

• Text file formats

• Foreign files (file types not supported by AppEnhancer can be stored in AppEnhancer and opened in the native application where they originated)

1.1.5.3.1   Managing documents

AppEnhancer enables you to manage content through its entire lifecycle. AppEnhancer Web Access is used for adding, retrieving, and processing content.

1.1.5.3.1.1   Viewing documents

The AppEnhancer system provides several client options for content viewing. AppEnhancer Web Access supports a wide variety of options that let the user or the administrator control how retrieved content is displayed. AppEnhancer supports browser-based content viewing that provides access to most content management functionality through a browser.

1.1.5.3.1.2   Using annotations

AppEnhancer enables users to add annotations to any image or text document page. An annotation is a note or a shape added to a document page, typically to focus attention on a particular part of the page. Users can use annotations in AppEnhancer Web Access to comment on the contents of a page, block areas of the page from view, or highlight important information. When a user creates an annotation, it is associated with the AppEnhancer document page on which the user created it. Annotations are edited and stored separately from the image, but they are displayed along with the image in AppEnhancer Web Access.

The types of annotations available include text, highlighting, lines, arrows, shapes, and rubber stamps. Text, highlighting, line, arrow, and shape annotations can be created in AppEnhancer Web Access. While rubber stamps cannot be created or configured within AppEnhancer Web Access, as an AppEnhancer administrator, you can edit rubber stamp security properties within AppEnhancer Web Access. Rubber stamp annotations enable users to place preset and custom text annotations on a page as well as image files supported by the AppEnhancer image library and embedded foreign files.

Using redactions

When any type of annotation (line, shape, text, highlight, or rubber stamp) has the redaction property applied to it, the annotation is considered to be a redaction.

Redactions are annotation shapes that are filled and opaque. Users can use redactions to secure or hide portions of image and text document pages. Users can

apply redaction to all available shapes: lines, freehand lines, arrows, rectangles, rounded rectangles, and ovals. When applied, the area of the page behind the redaction is not visible.

> **Note:** Redaction is not applicable for PDF documents.

Using annotation groups

You can use privileges to apply annotation-related security measures, so that annotations and redactions of any type can be created, modified, hidden, moved, or deleted only by users who have appropriate privileges. However, if you want to control users' access to specific annotations, you must use annotation groups. Annotation groups enable you to create associations between users, groups, and specific annotations. You can specify which users and groups can view or modify specific annotations, and which users and groups can hide or modify specific redactions. You use AppEnhancer Administrator to create annotation groups and to populate them with existing users and existing groups.

1.1.5.3.1.3   Submitting documents for full-text indexing

AppEnhancer Web Access users can make documents full-text searchable by submitting them to the AppEnhancer Indexing Service. Full-text searching enables users to find documents even if they do not know any of the index values. Users can also use full-text searches to narrow down an index value search. You can also perform a full-text search in a multibyte language (for example, Chinese).

1.1.5.3.1.4   Submitting documents and pages for OCR

Users can process images by using optical character recognition (OCR) in AppEnhancer Web Access. This process converts an image of text (for example, a scanned document or page) into actual text. After an image is processed using OCR, the text can be submitted to the AppEnhancer Indexing Service for full-text indexing, if configured. After the image files have been submitted to the AppEnhancer Indexing Service, they are passed on to the full-text engine. These images can then be accessed by AppEnhancer clients for document display and printing.

1.1.5.3.1.5   Filing documents for retention

AppEnhancer enables you to file documents for retention by using a supported retention management solution. Documents filed for retention cannot be deleted from the repository until the specified retention period expires. Users can, however, extend the retention period by applying a new retention class or policy to a document(s) where the expiration date is later than that associated with the original retention class or policy.

Retention is enforced at a file level by secure path feature of AppEnhancer.

AppEnhancer Web Access users with retention privileges can file documents for retention by using:

- Retention administration

  Retention administration is a component of records management that involves retaining information until it is permanently filed or disposed of. For example, an organization may be required to store a record, such as a contract, for a period of two years after expiration of the agreement. The contract document can be filed for retention at the beginning of that two-year period or even before, if a retention rule is also configured.

1.1.5.4   Pages

A page implies a single entity. Because AppEnhancer supports multiple object types, a page is redefined in the AppEnhancer system to describe a single object. A very long word processing file is considered a page to AppEnhancer. A page can also be, for example, a single scanned image, a 30-minute video clip, or an audio recording. Each page of a document has the same index record attached.

When a document is created in AppEnhancer, the object added as the new document is the first page of that document. Other pages can be added to the document. All pages contained in a single document use the same index record.

Pages can be processed separately by using the **Page** menu commands. Pages can be inserted or deleted at any time. Up to 250,000 pages can be attached to a document, and a page can come from one of many different types of sources.

1.1.5.5   Page versions

You can create different versions of the same page to add another layer of organization within a document.

Different versions of the same page need not have the same object type. For example, the original page version is a Microsoft Word file, but the new version can be a Microsoft Excel file.

1.1.6   Software integrations

The following table lists software that can be integrated with AppEnhancer:

Software	Description of Integration
AppEnhancer Integration Framework	AppEnhancer Integration Framework consists of the Event Dispatch Broker (EDB) and Workflow Integration Module (WIM) components. The EDB is a Windows Communication Foundation (WCF) service hosted by an IIS website. The EDB contains interfaces that facilitate communication between applications and AppEnhancer by enabling applications to subscribe to AppEnhancer-generated events. The WIM provides a common interface for third-party workflow solutions to communicate with AppEnhancer. It uses events dispatched through the EDB to support automatically starting business processes. The WIM communicates directly with AppEnhancer clients to support manually starting business processes.
AppEnhancer Web Services	AppEnhancer Web Services provides a development interface to the AppEnhancer system, enabling custom client or server components to call AppEnhancer functions through AppEnhancer Web Services.
AppEnhancer REST Services	AppEnhancer REST Services are a set of RESTful web service interfaces that interact with the AppEnhancer platform. It provides you with high efficiency and simplicity in programming, and also makes all services easy to use.
AppEnhancer Connector **Note:** AppEnhancer Connector is a legacy component. For more information about discontinued and deprecated features, please refer to the *OpenText AppEnhancer Release Notes*.	AppEnhancer Connector provides seamless integration between AppEnhancer and third-party applications. It enables you to capture index information from external sources and use the captured index information to perform AppEnhancer content management operations.
AppEnhancer for Microsoft Office **Note:** AppEnhancer for Microsoft Office is a legacy component. For more information about discontinued and deprecated features, please refer to the *OpenText AppEnhancer Release Notes*.	AppEnhancer for Microsoft Office is a software module that enables business application architects, designers, and developers to integrate AppEnhancer content management functionality into Office Business Applications based on a Microsoft Office system.

Note: Operating systems supported by AppEnhancer might not be supported by third-party components.

1.2   Features of AppEnhancer

Understanding the key features helps you to deploy AppEnhancer content management products effectively for your organization.

1.2.1   Common features

This section describes the common features of AppEnhancer.

1.2.1.1   Content capture

The AppEnhancer system includes many options for adding content to the system. Users can add pages to existing AppEnhancer documents through AppEnhancer Web Access, or Image Capture. To add pages, AppEnhancer Web Access enables importing, but does not enable scanning.

- Creating Batches: Users can add content to the AppEnhancer system by scanning or importing files to create batches. Batch scanning is supported by AppEnhancer Image Capture for Multi-Function Peripheral Connector.

- Indexing Batches: AppEnhancer Web Access users can index batches to create documents in the AppEnhancer system. AppEnhancer Web Access enables users to index batches. When a user indexes a batch, they can choose how to split the batch into documents.

1.2.1.2   Document search

The AppEnhancer system provides several options for retrieving stored documents. The AppEnhancer components employ many options for document searching. The simplest form of query is an index value search. More advanced index-based queries include expression, list, and wildcard searches. In addition, users can retrieve documents by searching document properties and can search multiple applications using Multi-Application Search (new name for search in AppEnhancer Web Access).

- Full-Text Search: Users can perform full-text indexing and full-text searching in single-byte and multibyte languages. A full-text search retrieves only documents that have been indexed by using the AppEnhancer full-text engine. Documents can be submitted to a Full-Text Indexing Service from AppEnhancer Web Access. The Full-Text Search option searches the full-text of documents that have been added to the AppEnhancer Indexing Service database. If no documents have been indexed, the full-text search will not return any results. Full-text searches can also be run on selected documents from the Result Set. The full-text search options include All Words, Any Words, Exact Phrase, or Expression.

  > **Note:** Multibyte languages require a Unicode database.

- Multi-Application Search/Cross Application Query: Enables users to create, configure, and edit and run queries, that enable them to search multiple applications.

> **Note:** This type of query cannot be run across data sources. Applications must reside within the same data source to be included together in a Cross Application Query.

- Expression Search: Expression searches within index fields are also available to search for documents that match a range of possibilities. This feature enables more options so that user queries return concise and accurate results. Available Expression search options are Between, Greater than, Greater than or equal to, Less than, Less than or equal to, and Not equal to.

- List of Values Search: List of Values logic enables users to define as many alternatives for each search field as they wish. In the Search Criteria Search Value box, for example, one could enter List: '123-45-6789', '111-11-1111', and other social security numbers to retrieve documents that have multiple index values in a single search.

- Wildcard Search: The asterisk (*) is used as a wildcard that can match any character or number of characters. This wildcard performs the beginning with, ending with, and pattern matching searches. For example, a query run using the syntax SM* finds all names beginning with the letters SM, and a query run using the syntax SM*TH finds all names which begin with SM and end with TH. The wildcard character can be used to narrow a search, controlling the number of documents returned in the **Result Set** page. Wildcards are valid for index fields with the following data types: Text, Timestamp, SSN, Telephone, ZIP Code, Boolean Choice, and User-defined List.

1.2.1.3 Operation modes

You can create new AppEnhancer application by using AppEnhancer Administrator, setting up index fields that will be used to organize the documents to be stored in the application. After the application has been built in AppEnhancer Administrator, it can be opened in AppEnhancer components. Users can add documents to the application through the AppEnhancer interface, attaching an index record to each document that will enable the document to be retrieved at a later date. After the document is stored, it can be retrieved and then printed, exported, or emailed, again using the AppEnhancer components. The tables of index data and all pointers to document location are maintained by AppEnhancer.

The AppEnhancer end user interface has three modes of operation that you can choose from: normal mode, check-in/check-out mode, and reason audit mode. Normal mode and check-in/check-out mode or reason audit mode cannot be used in combination, but check-in/check-out mode and reason audit mode can be enabled at the same time.

Check-in/check-out mode relates to the use of revision control in AppEnhancer. Revision control, which keeps track of who is working on a document, can be used to track previous revisions of a document. When revision control is in use, a user must check a document out before they can modify it. When users finish modifying documents, they must then decide how (or whether) to save the documents to the AppEnhancer document repository again.

When a user checks out a document, a copy of the document as it exists in the AppEnhancer repository is created by AppEnhancer. This copy becomes the working copy for the user. The user can close a checked out document and keep the document checked out so the user can continue working on it.

At any time, when a user finishes modifications, the user can choose to check the document back in. When the user checks a document in, the document that was initially checked out is saved as a previous document revision (or deleted through replacement) and the working copy is saved to the AppEnhancer document repository as a new (minor or major) document revision.

The user can also choose to cancel the checkout process of the document. When a user cancels the checkout process of the document, the working copy created when the user checked the document out is deleted, and AppEnhancer considers the revision of the document that was originally checked out as the current revision. Any changes made to the working copy are discarded.

Reason audit mode, which can be configured on the AppEnhancer application level, requires users to enter comments whenever they create, display, export, print, or email documents. When users display documents in reason audit mode, they must select options to print, email, or export if they want access to those functions when the documents are displayed. Reason audit mode requires the use of Audit Trails, because it enables you to audit the reasons users are displaying, exporting, printing, or emailing documents.

If you enable the Prompt for checkout when open documents option for the AppEnhancer application, when a user checks in a document in an application, the user can mark the document as a final revision. After this is done, users can open the document only in read-only mode and cannot check out or modify the document. Users with the Delete Doc privilege can delete final revisions.

As the system administrator, you may want to ensure that all users on your AppEnhancer system are operating AppEnhancer in the same mode. Unless the mode is enabled for an entire AppEnhancer application, users can configure the use of check-in/check-out mode on the **Data** tab in the Configuration settings. You can, however, set check-in/check-out mode or reason audit mode for an entire application when configuring the application in AppEnhancer Administrator to ensure consistency.

If you restrict access to the **Data** tab by not granting users the **Configure Work Station** privilege in AppEnhancer Administrator, you can prevent users from switching between modes. You can also configure the use of check-in/check-out mode or reason audit mode in AppEnhancer Administrator for each application. Enabling check-in/check-out mode for applications supports compliance with the Health Insurance Portability and Accountability Act of 1996 (HIPAA). Check-in/check-out mode is enabled for a user if it is enabled for either the current workstation or the current application of the user.

1.2.1.4   Compliance to standards

Many AppEnhancer features enable you to comply with current standards, such as the Health Insurance Portability and Accountability Act of 1996 (HIPAA).

1.2.1.4.1   Compliance with HIPAA

If you need to configure your AppEnhancer system to comply with the Health Insurance Portability and Accountability Act of 1996 (HIPAA), consider the following recommendations:

- Enable Audit Trail

- For each AppEnhancer application that contains data protected by HIPAA, enable the options that require users to enter comments when checking documents in and out

1.2.1.4.2   Checkin/checkout comments

For each AppEnhancer application that contains data protected by HIPAA, enable the following options:

- Prompt for checkout when open documents

- Checkout comments required

- Check-in comments required

- Reason Code

These options address the needs of privacy and security outlined by HIPAA by enabling you to identify which documents are being accessed, who is accessing them, and why the documents are being accessed.

1.2.2   AppEnhancer Web Services features

AppEnhancer Web Services provides a development interface to the AppEnhancer system, enabling custom client or server components to call AppEnhancer functions through AppEnhancer Web Services. The AppEnhancer Web Services interface is a server side layer that brokers communication with the business logic components of an AppEnhancer system. AppEnhancer Web Services provides complex business logic for accessing and manipulating content through a set of APIs, each of which represents a logical and transactional operation.

AppEnhancer Web Services offers:

- An easy-to-use business logic layer that implements all content management related operations

- The ability to accommodate method invocations through either remote calls

- The ability to serve applications running on different environments and different architectures

- User session management and software licensing

1.2.3   AppEnhancer REST Services features

AppEnhancer REST Services are a set of RESTful web service interfaces that interact with the AppEnhancer platform. These services provide high efficiency and simplicity in programming, and also make all services easy to use. AppEnhancer REST Services enables next-generation applications and mobile applications to interact with the AppEnhancer platform. It identifies resources by Uniform Resource Identifiers (URIs). It also defines specific media types to represent resources and no state transfer. It uses a limited number of HTTP standard methods (GET, POST, PUT and DELETE) to manipulate these resources over the HTTP protocol. AppEnhancer REST Services supports only the JavaScript Object Notation (JSON) and XML format for resource representation. JSON is a lightweight data interchange format based on a subset of the JavaScript Programming Language standard.

1.2.4   OpenText™ Process Automation features

OpenText Process Automation is a workflow solution designed specifically for the AppEnhancer content management system. OpenText Process Automation processes the post-activity events that are generated by the AppEnhancer system when AppEnhancer Web Access users create, modify, and delete document indexes. In addition to designing workflows that start automatically based on these events, you can configure a workflow to start only when the event criteria meets your specifications. You can also enable AppEnhancer Web Access users to start workflows manually from the query result set.

You must install and configure the AppEnhancer Event Dispatch Broker (EDB) and AppEnhancer Workflow Integration Module (WIM) components for OpenText Process Automation. You need AppEnhancer Web Services to create forms and to associate workflows with AppEnhancer generated events.

Chapter 2

Planning an AppEnhancer system

This chapter provides information on how to plan an AppEnhancer system.

2.1 Overview of an AppEnhancer system implementation

1. Plan your AppEnhancer system:

   a. Familiarize yourself with AppEnhancer architecture and required components. Select optional components based on the features that you want to implement.
   b. Designate a workstation for each component.
   c. Plan the installation of each component.

2. Install and configure core components (such as AppEnhancer Administrator), and any other administrative component that you want to implement.

3. Install and configure the server components that you want to implement.

4. Install and configure any integrations that you want to implement.

5. Install and configure the user client components that you want to implement.

6. Provide instructions to end users so they can use the client components.

7. Monitor each of your AppEnhancer Servers.

2.2 Workstation allocation for each component

After you have identified the necessary components, you must then decide where each of those components will be located.

📄 **Note:** To determine the operating system version, service pack, and physical memory available for Windows, type `winver` in **Run** and click **OK**.

2.2.1   Scope of installation

Before AppEnhancer deployment, it is important to establish whether your installation will be a standalone application or an enterprise-wide system. This enables you to plan your system architecture, security settings, system components, and configuration. In addition, you cant take advantage of some of the AppEnhancer features for deployment.

2.2.1.1   Standalone deployment

If you are planning to run AppEnhancer for a small user base, with a small number of documents, you should consider deploying AppEnhancer in a standalone capacity. You can install AppEnhancer and its related features on a single workstation to be used by one to three workstations.

If you expect that the system will grow in scope as you use it, consider using a more scalable enterprise-wide method of deployment.

2.2.1.2   Enterprise deployment

If you are planning to run AppEnhancer for a large user base, with a large number of documents and document transactions, you should deploy AppEnhancer in an enterprise capacity. AppEnhancer components require a distributed network architecture (in other words, system resources such as the AppEnhancer Indexing Service and Document Storage Server should be placed on dedicated, separate, network workstations).

2.2.2   Installation of each server component on dedicated workstation

Your AppEnhancer system can contain multiple servers: AppEnhancer Rendering Server, Web Access Server, Web Services Host Server, Indexing Service, and AppEnhancer Full-Text Server. Each server should be installed on a workstation that is not a domain controller and that does not host any other servers that have heavy processor or memory usage, such as mail servers, database servers, or other AppEnhancer content management servers.

2.2.3   Workstation allocation for each backend server

Consider the following for each backend server:

- Database Location: All AppEnhancer users should have network access to the workstation where the AppEnhancer database is located. Enough space should be allotted to account for database growth.

- Document Storage Location: For an enterprise-wide deployment, you should consider giving document storage a dedicated workstation, with more than enough space to accommodate document load.

- License Server Location: A location that is always accessible to each AppEnhancer content management product.

- Security Server Location: If you want to import users and groups into AppEnhancer (rather than create them one by one), a security server is required. The minimum system requirements for supported security servers are sufficient for importing users and groups into AppEnhancer. Microsoft Active Directory Services is the supported security server for CM or Windows security provider.

2.2.4 Workstation allocation for each AppEnhancer server

2.2.4.1 Administrator location

AppEnhancer Administrator should be installed on a workstation.

2.2.4.2 Indexing Service location

It is important to decide on a proper location for the AppEnhancer Indexing Service.

Indexing Service can have multiple installations on servers acting as Index server or OCR server or both. The AppEnhancer Indexing Service requires an impersonation account that grants security privileges to it. This account is essential for the AppEnhancer Indexing Service to operate properly.

Multiple AppEnhancer Indexing Services can also be installed to process a heavier volume of requests.

Some options for AppEnhancer Indexing Service configurations include:

- One AppEnhancer Indexing Service with one queue: In this configuration, the AppEnhancer Indexing Service would poll one full-text queue for jobs to process.

- One AppEnhancer Indexing Service and multiple queues: In this configuration, the AppEnhancer Indexing Service would poll multiple queues for full-text jobs to process. This would not increase performance, in that you will still have one server processing all jobs. However, multiple queues may be preferable to accommodate distinct groups of users, multiple data sources, or different processes (full-text and OCR).

- Multiple AppEnhancer Indexing Services with multiple queues: This configuration is supported only when one AppEnhancer Indexing Service polls each queue for full-text jobs to process. This would increase performance in that multiple servers would be processing jobs from the queues.

Using multiple AppEnhancer Indexing Services with one queue is not supported.

2.2.4.3   Full-Text Server location

The full-text server location is the AppEnhancer full-text server installation location.

2.2.4.4   Web Access Server location

The AppEnhancer Web Access Server should be installed on a workstation.

2.2.4.5   Rendering Server location

The AppEnhancer Rendering Server should be installed on a workstation.

2.2.4.6   Web Services Server location

AppEnhancer Administrator data source group can have only one AppEnhancer Web Services Server. The AppEnhancer Web Services Server requires an authentication account.

2.2.4.7   REST Services Server location

AppEnhancer Administrator data source group can have only one AppEnhancer REST Services Server. The AppEnhancer REST Services Server requires an authentication account.

2.2.4.8   Auto Retention Filer Server location

The AppEnhancer Auto Retention Filer Service is required if you want to file documents for retention automatically. Only one AppEnhancer Auto Retention Filer server may be configured per AppEnhancer data source group. The AppEnhancer Auto Retention Filer also requires an authentication account.

Chapter 3

Planning security

When you plan security for the AppEnhancer include the following factors:

- Decide which security provider to use for each data source.
- Plan to create one or more authentication accounts.
- If you will be implementing AppEnhancer Web Access Server or Web Services, decide which security settings to use.
- Identify users and decide how you will implement authorization.
- Decide whether to implement security mapping.

3.1 Security providers

Security providers implement authentication, which requires all users to enter a valid user name and password to access most modules. AppEnhancer Administrator enables you to choose the security provider (only CM or Windows) for each data source. AppEnhancer Administrator enables you to create users and groups and to import users and groups. ADFS, CAS, OTDS, SAML 2.0 and customized security providers can be enabled by changing the configuration file.

AppEnhancer Administrator offers two prepackaged security providers that you can choose from for each data source:

- CM
- Windows

You can also enable customized security providers such as ADFS, CAS, OTDS, and SAML 2.0 by changing the configuration file.

You can create or import users and groups by using AppEnhancer Administrator.

3.1.1 Security provider architecture

Most AppEnhancer components (such as AppEnhancer Web Access and Administrator) use security providers for user authentication. AppEnhancer Administrator enables you to choose the security provider for each data source. Then, AppEnhancer Administrator enables you to create users and groups and to import users and groups.

3.1.2   CM security provider

The CM security provider is a prepackaged security provider, with the following capabilities:

- You can import users and groups from an existing security system such as Windows.

- The import is simply a snapshot of the current users and groups list. The imported list is not updated when a change is made to the source.

- The imported list does not contain password of each user. The password must be reentered in AppEnhancer Administrator.

3.1.3   Windows security provider

The Windows security provider is a prepackaged security provider, with the following capabilities:

- You can import users and groups only from Windows (Microsoft Active Directory Services).

- When you change the password of a user or group membership in the Windows user maintenance utility, the next time that user logs into an AppEnhancer content management module, the changes are reflected in the authentication of that user and the functions available to that user.

- You must use the Windows user maintenance utility to maintain passwords and group membership.

- Enables single logon, which means that users who are already logged into Windows do not need to log in again when starting AppEnhancer components. AppEnhancer uses the current Windows user account information for authentication.

3.1.4   ADFS, CAS, OTDS, and SAML 2.0 security providers

Active Directory Federation Services (ADFS), Central Authentication Service (CAS), OpenText Directory Services (OTDS), and Security Assertion Markup Language (SAML) 2.0 are security providers that can be configured with the following capabilities:

- Provide SSO authentication services. The administrator must configure ADFS, CAS, OTDS, or SAML 2.0 information in the `web.config` file to enable AppEnhancer Web Access to support SSO. ADFS server, CAS server, OTDS server, or SAML 2.0 must be configured to be able to return the user information to AppEnhancer Web Access for validation. To enable validation, the user information must be already imported into the AppEnhancer data source.

- With ADFS (including SAML 2.0), you can import users and groups from Windows (Microsoft Active Directory Services). With CAS (including SAML 2.0), you can create new users and groups from import tools. With OTDS, you can import users and groups from OTDS Server.

- AppEnhancer Administrator and Web Access support ADFS, CAS, OTDS, and SAML 2.0.

For more information, see the *OpenText AppEnhancer Administration Guide*.

3.1.5 Best practices for security provider

AppEnhancer Administrator enables you to import users and groups and to create users and groups manually. Users and groups in different security provider can exist in the same data source. However, the best practice for this task depends primarily on which security provider is in use.

- If you create a user or group manually, it is in CM security.

- If you import a user or group from Active Directory, you can choose to import it as Windows Security or CM Security.

- You can import users and groups from XML file which was exported from other data source.

- If the data source is using Windows security provider, only users and groups in Windows security are valid.

- If the data source is using CM security provider, only users and groups in CM security are valid.

- If you switch the data source from CM security to Windows security and vice versa, all users and groups remain.

3.2 Configuration of authentication accounts

You must create an authentication account on each of the workstations intended for the AppEnhancer Administrator, Indexing Service, Web Services, REST Services, Web Access Server, Rendering Server, or Auto Retention Filer. If any of the system resources are on remote workstations, create appropriate authentication accounts configured as Domain users for access to those resources.

1. Create one or more user accounts to be used as authentication accounts. For more information, see "Accounts and rights required for resource authentication accounts" on page 57.

2. Specify each authentication account.

   - The AppEnhancer Administrator's account is specified in the Component Registration Wizard when using it register AppEnhancer Administrator.

   - The AppEnhancer Indexing Service account is specified on the Server Management page of AppEnhancer Administrator.

   - The AppEnhancer Web Access Server, Rendering Server, and Auto Retention Filer accounts are specified in AppEnhancer Administrator.

3. Additional steps may be required for specific AppEnhancer servers.

4.   Test each authentication account.

3.2.1   Configuring SAML 2.0 authentication accounts

AppEnhancer Web Access supports single sign-on (SSO) authentication through SAML 2.0. SAML 2.0 is an XML-based protocol that uses security tokens to pass user information between an identity provider and a service provider. The SAML 2.0 identity provider (IdP) that Web Access supports can be an Active Directory Federation Services (ADFS) server or Apereo Central Authentication Service (CAS).

3.2.1.1   Configuring SAML 2.0 on a CAS server

The following instructions are based on an environment that has CAS installed with SAML 2.0 protocol support. You must have a valid certificate in place. To verify that you can access CAS, navigate to https://*<cas-server-address>*/cas/login. You should be able to sign in using the CAS login screen.

3.2.1.1.1   Configuring the CAS server

1.   Create the file `/etc/cas/services/AppEnhancer-10000003.json` as a service definition for Web Access:

```
{
   "@class" : "org.apereo.cas.support.saml.services.SamlRegisteredService",
   "serviceId" : "https://<ae-server-address>/AppEnhancer",
   "name" : "AppEnhancer",
   "id" : 10000003,
   "evaluationOrder" : 10,
   "metadataLocation" : "/etc/cas/sp-metadata.xml",
   "attributeReleasePolicy" : {
      "@class" : "org.apereo.cas.services.ReturnAllowedAttributeReleasePolicy",
      "allowedAttributes" : [ "java.util.ArrayList", [ "uid", "udcid" ] ]
   },
   "signAssertions" : true,
   "logoutType" : "BACK_CHANNEL",
   "signingSignatureAlgorithms": [
      "java.util.ArrayList",
      [
         "http://www.w3.org/2000/09/xmldsig#rsa-sha1",
         "http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha1"
      ]
   ],
   "signingSignatureReferenceDigestMethods": [
      "java.util.ArrayList",
      [
         "http://www.w3.org/2000/09/xmldsig#sha1"
      ]
   ]
}
```

Note the following:

- The *@class* of the service is "org.apereo.cas.support.saml.services.SamlRegisteredService" rather than "org.apereo.cas.services.RegexRegisteredService".

- The *serviceId* is specified as an exact-match string, not a regular expression. Specifically, this attribute must be equal to the entityID of the service.

- The attribute*metadataLocation* is used to tell the IdP where it can obtain the service provider's metadata.

- The *ReturnAllowedAttributeReleasePolicy* is used to assign SAML-specific attribute names for the service provider.

- The *signAssertions* is used to tell whether assertions should be signed. This value must be set to "true" for integration with Web Access.

2. Create the file /etc/cas/sp-metadata.xml:

```xml
<?xml version="1.0"?>
<md:EntityDescriptor xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata"
entityID="https://<ae-server-address>/AppEnhancer">
  <md:SPSSODescriptor AuthnRequestsSigned="false" WantAssertionsSigned="true"
protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
    <md:KeyDescriptor use="signing">
      <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:X509Data>
          <ds:X509Certificate>... base64-encoded certificate elided ...</
ds:X509Certificate>
        </ds:X509Data>
      </ds:KeyInfo>
    </md:KeyDescriptor>
    <md:KeyDescriptor use="encryption">
      <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:X509Data>
          <ds:X509Certificate>... base64-encoded certificate elided ...</
ds:X509Certificate>
        </ds:X509Data>
      </ds:KeyInfo>
    </md:KeyDescriptor>
    <md:SingleLogoutService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
Redirect" Location="https://<ae-server-address>/AppEnhancer/account/
Saml2SingleSignOutRequestHandler"/>
    <md:NameIDFormat>urn:oasis:names:tc:SAML:2.0:nameid-format:persistent</
md:NameIDFormat>
    <md:AssertionConsumerService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
POST" Location="https://<ae-server-address>/AppEnhancer/account/
Saml2SingleSignOnHandler" index="0"/>
  </md:SPSSODescriptor>
</md:EntityDescriptor>
```

Note the following:

- Replace the signing certificate with the one used for Web Access.

- An encryption certificate is not used; fill with signing certificate.

- Web Access supports SAML 2.0 single logout. Use location in the example <md:SingleLogoutService> endpoint elements in the metadata.

- Web Access support SSO Redirect protocol. Use location in the example <md:AssertionConsumerService> endpoint elements in the metadata.

- Web Access require that the SAML NameID format be the following:

```xml
<md:NameIDFormat>
   urn:oasis:names:tc:SAML:2.0:nameid-format:persistent
</md:NameIDFormat>
```

3.2.1.1.2   Configuring the Web Access server

1. Enable CAS SAML 2.0 in the `Web.config` file:

    a. Uncomment the <externalAuth> and the "saml2" section:

    ```
    <externalAuth>
        <providers>
          <!-- <provider name="cas" enabledToAllDataSources="true"
    aeAuthenticationChain="ProviderId">
            <datasources>
              <datasource name="AppEnhancerDEMO" />
              <datasource name="demo1" />
            </datasources>
          </provider>
          <provider name="adfs" enabledToAllDataSources="true"
    aeAuthenticationChain="ProviderId, AD">
          </provider>
          <provider name="otds" enabledToAllDataSources="true"
    aeAuthenticationChain="ProviderId, AD">
          </provider>-->
          <provider name="saml2" enabledToAllDataSources="true"
    aeAuthenticationChain="ProviderId">
          </provider>
    </providers>
      </externalAuth>
    ```

    b. Uncomment the <saml2ClientConfig> section:

    ```
    <saml2ClientConfig
          serverName="https://<ae-server-address>"
          attributeMap_Usrnam="uid"
          attributeMap_Securid="udcid"
          issuer="https://<ae-server-address>/AppEnhancer"
          saml2Server="https://<cas-server-address>:8443/cas/idp/profile/SAML2/
    Redirect/SSO"
          saml2ServerSloEndpoint="https://<cas-server-address>:8443/cas/idp/
    profile/SAML2/Redirect/SLO"
          nameIDPolicy="urn:oasis:names:tc:SAML:2.0:nameid-format:persistent"
          certificateValidator="None"
        clientCertificateThumbprint="d3b1280d739180f38cc4ffcc3ac441573a7c1028"
          signingAlgorithmUrl="http://www.w3.org/2000/09/xmldsig#rsa-sha1"
          isCasServer="true"
      />
    ```

    Note the following:

    - *serverName* – The server name that hosts AppEnhancer Web Access.

    - *attributeMap_Usrnam* – The value of this attribute will be used to extract the value from a security token and create a mapping to the column Usrnam in the table ae_login of the AppEnhancer database.

    - *attributeMap_Securid* – The value of this attribute will be used to extract the value from a security token and create a mapping to the column Securid in the table ae_login of the AppEnhancer database.

    - *nameIDPolicy* – Must be "urn:oasis:names:tc:SAML:2.0:nameid-format:persistent".

    - *clientCertificateThumbprint* – The thumbprint of the same certificate defined in sp-metadata.xml for signing. The certificate must be added to [Certificate – Local Computer\Personal\Certificates] and the private key of this certificate must be accessible by IIS (add IIS_IUSRS to security with Read permission).

- *isCasServer* – The flag used to indicate if the SAML server is a CAS server. *isCasServer* must be set to "true".

2. Create a CM user and change the following field values in the database:

   - The value of the column **securid** should be the SID of the CAS SAML 2.0 user.

   - The value of **providerid** should be changed to EF575DC6-E09D-417B-93DC-007523F177D6 (the ID is same for all CAS SAML 2.0 users).

3. Navigate to the Web Access login page and click the **SAML 2** login button. The browser is redirected to the CAS SAML 2.0 server. Input the user name and password for the user. The browser is redirected back to Web Access.

If there is a thumbprint error

- Copy thumbprint from the certificate and update in <saml2ClientConfig> section of the Web.config.

3.2.1.2 Configuring SAML 2.0 on an ADFS server

The following instructions are based on an environment that has ADFS installed in a customer's environment or separate domain from the Web Access .Net server.You must have a valid certificate in place. To verify that you can access ADFS, navigate to https://<*FQDN*>/adfs/ls/IdpInitiatedSignon.aspx. You should be able to sign on the ADFS login screen from AppEnhancer Web Access .Net server.

> **Note:** ADFS Server should set Registered Service Principal Names, but you may have to manually set this by using the `setspn` command from the command prompt.

3.2.1.2.1 Configuring the Web Access server

Complete the following steps in the Web.config file:

1. From the sub-node <modules> in the node <system.webServer>, uncomment the configuration of module WSFederationAuthenticationModule and SessionAuthenticationModule:

```
<add name="WSFederationAuthenticationModule"
type="System.IdentityModel.Services.WSFederationAuthenticationModule,
System.IdentityModel.Services, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b77a5c561934e089" preCondition="managedHandler" />
      <add name="SessionAuthenticationModule"
type="System.IdentityModel.Services.SessionAuthenticationModule,
System.IdentityModel.Services, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=b77a5c561934e089" preCondition="managedHandler" />
    </modules>
```

2. Uncomment the node <system.identityModel>.

3. Change the configuration of the node <audienceUris> (Web Access URL should be changed in this node) and <trustedIssuers> (ADFS server issuers should be changed in this node).

4. Change the thumbprint in this section to the ADFS token certificate signing thumbprint. Your token signing thumbprint can be found in the ADFS console under **ADFS** > **Service**> **Certificates**.

5. Change the configuration of node <wsFederation> (in this node, *issuer* is the URL of the ADFS server issuer, *realm* and *reply* are the Web Access URL).

```
<system.identityModel>
    <identityConfiguration saveBootstrapContext="true">
      <audienceUris>
        <add value="https://tsax8sp1.tslan2.com/appenhancer" />
      </audienceUris>
      <certificateValidation certificateValidationMode="None" />
      <issuerNameRegistry
type="System.IdentityModel.Tokens.ConfigurationBasedIssuerNameRegistry,
System.IdentityModel, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=B77A5C561934E089">
        <trustedIssuers>
          <add thumbprint="5e0be7a1dfa7aa221884ef54d46047ba9bdfed22" name="https://
adfs.test.com/adfs/services/trust" />
        </trustedIssuers>
      </issuerNameRegistry>
    </identityConfiguration>
  </system.identityModel>
  <system.identityModel.services>
    <federationConfiguration >
      <cookieHandler mode="Default" requireSsl="false" />
      <wsFederation passiveRedirectEnabled="true" issuer="https://adfs.test.com/
adfs/ls/" realm="https://tsax8sp1.tslan2.com/appenhancer" reply="https://
tsax8sp1.tslan2.com/appenhancer" requireHttps="false" />
    </federationConfiguration>
  </system.identityModel.services>
```

6. Uncomment the ADFS <externalAuth> and comment out the "CAS" section:

```
<externalAuth>
    <providers>
      <!--<provider name="cas" enabledToAllDataSources="true">
        <datasources>
          <datasource name="AppEnhancerDEMO" />
          <datasource name="demo1" />
        </datasources>
      </provider>-->
      <provider name="adfs" enabledToAllDataSources="true">
      </provider>
    </providers>
  </externalAuth>
```

7. Above the <system.diagnostics>, uncomment the <adfsClientConfig>. Change the server name to your Web Access .Net server. The name and primary SID will be used later when setting up the relying party trust claims trust.

```
<adfsClientConfig
        serverName="https://tsax8sp1.tslan2.com"
        attributeMap_Usrnam="http://schemas.xmlsoap.org/ws/2005/05/identity/claims/
name"
        attributeMap_Securid="http://schemas.microsoft.com/ws/2008/06/identity/
claims/primarysid" />
  <system.diagnostics>
```

Note the following:

- *serverName* – The server name that hosts AppEnhancer Web Access.

- *attributeMap_Usrnam* – The value of this attribute will be used to extract the value from a security token and create a mapping to the column Usrnam in the table ae_login of the AppEnhancer database.

- *attributeMap_Securid* – The value of this attribute will be used to extract the value from a security token and create a mapping to the column Securid in the table ae_login of the AppEnhancer database.

Complete the following steps in the FederationMetadata.xml file:

1. In the Web Access installation folder, navigate to `FederationMetadata\2007-06\FederationMetadata.xml`.

2. Change all instances of the Web Access server address in the file:

```
<?xml version="1.0" encoding="utf-8"?><EntityDescriptor ID="_a099d314-1cd3-4c8f-
ac07-59bc7dbf9be5" entityID="https://tsax8sp1.tslan2.com/appenhancer"
xmlns="urn:oasis:names:tc:SAML:2.0:metadata"><RoleDescriptor
xsi:type="fed:ApplicationServiceType" xmlns:fed="http://docs.oasis-open.org/wsfed/
federation/200706" protocolSupportEnumeration="http://docs.oasis-open.org/wsfed/
federation/200706" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"><fed:TargetScopes><wsa:EndpointReference xmlns:wsa="http://www.w3.org/
2005/08/addressing"><wsa:Address>https://tsax8sp1.tslan2.com/appenhancer</
wsa:Address></wsa:EndpointReference><wsa:EndpointReference xmlns:wsa="http://
www.w3.org/2005/08/addressing"><wsa:Address>https://tsax8sp1.tslan2.com/
appenhancer</wsa:Address></wsa:EndpointReference></
fed:TargetScopes><fed:PassiveRequestorEndpoint><wsa:EndpointReference
xmlns:wsa="http://www.w3.org/2005/08/addressing"><wsa:Address>https://
tsax8sp1.tslan2.com/appenhancer</wsa:Address></wsa:EndpointReference></
fed:PassiveRequestorEndpoint></R
oleDescriptor></EntityDescriptor>
```

Complete the following steps in AppEnhancer .NET

1. In **Advanced Settings**, set the AppEnhancer .NET **Load User Profile** setting to **True**.

2. Click **OK**.

Complete the following steps in the ADFS console:

1. In the ADFS console, under Trust Relationships, click **Relying Party Trust** > **Add Relying Party Trust**.

2. Select **Import data about the relaying party published online or on a network**, and then add the path to your Web Access `federationmetadata.xml` file:

   https://wx80sp1test.axqa.com/AppEnhancer/FederationMetadata/2007-06/FederationMetadata.xml

   Browse to this link in and install the certificates.

   Click **Next**.

3. On the **Specify Display Name** page, select a display name and click **Next**. This name will be displayed as your trust in the ADFS console.

4. On the **Configure Identifiers** page, make sure that the relying party identifiers are set to your Web Access .Net server.

5. On the **Choose Access Control** page, under **Policy**, permit all users to access the relying party. You do not need to configure multi-factor authentication.

6. On the **Ready to Add Trust** page, on the **Endpoints** tab, click **Add WS-Federation**. It should point to your Web Access .NET URL.

7. On the **Advanced** tab, select **Secure Hash algorithm: SHA-256**.

8. In the ADFS console, under Trust Relationships, click **Relying Party Trust**. Right-click the relying party trust that you created in the previous steps, and click **Edit Claim Rules** > **Add Rule**.

   a. Under **Claim rule name**, add a user's name and user's primary SID.

   b. In **Attribute store** drop-down list, select **Active Directory**.

   c. Under **LDAP Attribute**, select **name** and **objectSid**.

9. In the ADFS console, click **Authentication Policies**. Under **Primary Authentication**, click **Edit**. In the **Extranet** and **Intranet** sections, ensure that only the **Forms Authentication** check boxes are selected.

10. Create a CM user and change the following field values in the database:

   • The value of the column **securid** should be the SID of the ADFS user.

   • The value of **providerid** should be changed to 5AA419D9-1AC2-47B6-9397-EC298C0CFA3F (the ID is same for all ADFS users).

Complete the following steps in Web Access:

• Navigate to the Web Access login page and click the ADFS login button. The browser is redirected to the ADFS server. Input the user name and password for the user. The browser is redirected back to Web Access.

3.2.1.2.2   Configuring the ADFS server

Prerequisites

Before you configure the ADFS server, ensure that the following requirements are met:

• Your AppEnhancer installed client machine has access to local host using the computer name.

• AppEnhancer is installed with AD domain username as impersonate account.

• Check SSL settings to ensure https login is enabled.

1. In the ADFS console, under Trust Relationships, click **Relying Party Trust** > **Add Relying Party Trust**, and complete the following steps:

   a. On the Select Data Source page, select **Enter data about the relying party manually**.

   b. On the Choose Profile page, select **AD FS profile**.

    c.   On the Configure URL page, select the **Enable support for the SAML 2.0 WebSSO protocol** check box, and replace the URL with your Web Access Server URL.

    d.   On the Configure Identifiers page, under Relying party trust identifier, enter your Web Access Server URL.

2. After creating the relying party trust, open the **Properties** > **Endpoints** tab > **Edit Endpoint** to add a new SAML logout endpoint. Under Trusted URL and Response URL, enter your Web Access Server URL.

3. On the **Signature** tab, import the certificate which is used by your Web Access Server to sign the SAML 2.0 logout message.

4. On the **Advanced** tab, set the secure hash algorithm to **SHA-256**, and click **OK**.

5. In the ADFS console, under Trust Relationships, click **Relying Party Trust**. Right-click the relying party trust that you created in the previous steps, and click **Edit Claim Rules**. Create a new Name ID rule which will be used by SAML 2.0 protocol.

6. Configure the section saml2ClientConfig in the Web.config file.

3.3   Accounts and rights required for resource authentication accounts

You must configure a global Windows account before using the AppEnhancer Administrator, Indexing Service, Web Access, Rendering Server, Web Services applications, or Auto Retention Filer. This account accesses local and remote resources during its operation. AppEnhancer Web Access, and Web Services require a global account even if they are configured to use integrated database security or integrated web server credentials. It is recommended that the global account be a Windows domain account so that the application software can access remote resources. If you use a local account, then the applications may access only local resources or unsecured remote resources.

The rights required vary depending on the user account, the workstation on which the user account is located, and the operating system on that workstation.

The following service accounts require Log on as a service rights:

- AppEnhancer Administrator
- AppEnhancer Indexing Service
- The AppEnhancer Web Access (if global credentials are in use) or the supplied account (if supplied credentials are in use)
- AppEnhancer Rendering Server
- AppEnhancer Auto Retention Filer

The following service accounts require Log on locally rights when Windows security provider is in use:

- AppEnhancer Web Access, Web Services, or REST Services Server

> **Note:** To access web applications when Windows security provider is in use, the service account must have both **Log on locally** and **Log on as a service** rights enabled.

3.4 Configuration of resource authentication credentials

When a user requests a document from an AppEnhancer server, the server needs to access multiple resources to respond to that request. To do so, the server must provide appropriate credentials for each resource. Using AppEnhancer Administrator, you can configure separate credentials for the following resources:

- Data sources
- Paths, such as paths on the AppEnhancer Web Access Server, paths to the AppEnhancer Rendering Server cache, or AppEnhancer write paths

The following table describes the types of credential settings:

Credential setting	Description
Application	AppEnhancer Web Access and Web Services or REST Services uses the credentials from the user logged into AppEnhancer Web Access or Web Services or REST Services to access the resource. **Notes** - Application credentials are valid only when the credentials are valid Windows accounts. In other words, you should be using the Windows security provider with your AppEnhancer system.

Credential setting	Description
Global	AppEnhancer Web Access, and Web Services or REST Services uses the account specified in AppEnhancer Administrator as the AppEnhancer Web Access global authentication account to access the resource. 📄 **Notes** • Global credentials are selected by default when you configure data sources or paths in AppEnhancer Administrator. This model is similar to the AppEnhancer Web Access impersonation account model. • Global credentials are also selected by default when you upgrade from previous releases of AppEnhancer Web Access.
Supplied	AppEnhancer Web Access and Web Services or REST Services uses a specific set of credentials (configured in AppEnhancer Administrator) to access the resource.

Global and Supplied credentials are entered by the system administrator through AppEnhancer Administrator. Application credentials come from the user logging into AppEnhancer Web Access or Web Services. Use Global credentials (the AppEnhancer Web Access global authentication account) to ensure that this account has access to all resources and appropriate rights.

If you want to use a different credential setting, however, the options available to you depend on which security provider you select, and, in the case of data sources, the type of database and database security setting you choose. Regardless of the credential setting you select, when you assign credentials for a resource, you must ensure that the account(s) you choose have access to the resource.

3.4.1 Configuration of data source credentials

When a user requests a document from an AppEnhancer Web Access server, AppEnhancer Web Access needs to access a data source using the appropriate credentials to respond to that request. When you configure a data source in AppEnhancer Administrator, you can choose the credentials AppEnhancer Web Access uses to access the data source.

The following table describes the recommended AppEnhancer Web Access data source credential settings:

Database and security setting	Security provider	Recommended data source credentials
Username/password-based either with or without saved credentials, for example: • SQL Server using SQL Server security • Oracle using Oracle security	• CM • Windows	• Global • Supplied **Note:** You can configure Application credentials if you are using the CM security provider, but only if the AppEnhancer user accounts are also valid Windows accounts. This configuration is not recommended.

The recommended approach when configuring data source credentials is to use Global credentials (the AppEnhancer Web Access global authentication account) and to ensure that this account has access to all resources and appropriate rights.

3.4.2   Configuration of path credentials

Path credentials are required for AppEnhancer Web Access. When a user requests a document from an AppEnhancer Web Access server, AppEnhancer Web Access needs to access paths, such as paths on the AppEnhancer Web Server, paths to the AppEnhancer Rendering Server cache, or AppEnhancer write paths. To do so successfully, it needs to provide appropriate credentials for the path. When you configure a path in AppEnhancer Administrator, you must choose the credentials that AppEnhancer Web Access uses to access the path.

The following table describes the recommended AppEnhancer Web Access path credential settings:

Security provider	Recommended path credentials
• CM	• Global • Supplied **Note:** You can configure Application credentials if you are using the CM security provider, but only if the AppEnhancer user accounts are also valid Windows accounts. This configuration is not recommended.

The recommended approach when configuring path credentials is to use Global credentials (the AppEnhancer Web Access global authentication account) and to make sure that this account has access to all resources and appropriate rights.

If you do not assign credentials to a path, Global credentials (the AppEnhancer Web Access global authentication account) are used to access the resource. For example, if you configure the Document Write Path in AppEnhancer Administrator but do not configure that path on the **Paths** page in AppEnhancer Administrator and assign credentials to it, then the AppEnhancer Web Access global authentication account is used by default.

> **Note:** AppEnhancer Rendering Server uses its own account to access all paths.

3.5 Configuration of security settings for AppEnhancer Web Access, Web Services, and REST Services

The AppEnhancer Web Access Security enables you to configure settings that control the authentication methods used by clients when accessing the AppEnhancer Web Access Server or Web Services or REST Services. The settings that you should choose depend on the credentials you select for AppEnhancer Web Access or Web Services or REST Services system resources. For more information, see "Configuration of authentication accounts" on page 49.

If you are using either Global or Supplied credentials, you can enable users accessing the AppEnhancer Web Access Server or Web Services or REST Services to be authenticated anonymously using the anonymous authentication option.

The following table describes the recommended AppEnhancer Web Access security settings:

Credentials	Anonymous access	Auto-login	Request full-text license
• Global • Supplied	Enabled	Disabled	Disabled
Application	Disabled	Enabled (optional)	Enabled (optional)

3.6 Levels of authorization

Authorization is the granting of specific access privileges according to the user name. Security profiles contain information pertaining to specific privileges of users within the AppEnhancer system. Security profiles are configured from within AppEnhancer Administrator, following the standard user and group security rules.

AppEnhancer Administrator offers three levels of security (function, application, and document) to prevent unauthorized users from gaining access to sensitive information stored in the AppEnhancer system.

3.6.1   User identification

An important part of planning your AppEnhancer security implementation is identifying the users who will be creating and retrieving AppEnhancer documents, and the administrators who will be responsible for maintaining the AppEnhancer system. You might want to start by listing your users in a table format,where you can add information about hardware to be used for the installation and notes on whether or not to use workstation profile settings to configure workstation settings. If the specific people who will use AppEnhancer have already been identified in your organization, you can list those people by name and then note each person's role in the system. If you do not yet know exactly who will be using the system, you can identify the roles that will be needed to make the system work and then add specific names later.

For an enterprise system with many users, you can save time by importing Windows group and user lists using AppEnhancer Administrator. After you have imported users, you can add security profiles. If the AppEnhancer system is using the CM security provider, you must also enter a password for each user.

3.6.2   Function and application level security

All authorization is driven by security profiles. In each profile, privileges for a list of AppEnhancer functions are granted. AppEnhancer enables you to implement security user by user, or to create groups of users and establish security settings that apply to an entire group. Profiles can be created that convey default privileges for all AppEnhancer applications, or that convey privileges for a specific application. Global profiles are created to give users common functional privileges for all AppEnhancer applications. The privileges defined globally are automatically granted to the user for all applications on the data source. However, by using application profiles, the user can be given different privileges for each AppEnhancer application.

When a user has both global and application-specific profiles defined, the settings in the application-specific profile override the settings in the global profile for that application only. If, for example, a user has a global profile with Display and Scan privileges enabled, and an application profile for the application DEMO which has only Display privileges enabled, he will only be able to display documents when using the DEMO application. Privileges granted to group profiles are reflected in the Web Access User Settings for any users that belong to the group. The user has all privileges enabled in the group profile and all privileges enabled in the Web Access User Settings. However, you can choose whether to accept group settings for the user or to override those settings by setting specific privileges for the user.

After a group has been created, you can add members to the group. Any user added as a member in a group is given the same privileges as defined for the group. Privileges of a group can vary from one application to the next; you could assign full privileges to a group in one application, but limited privileges in another.

3.6.2.1 Function level security

Within each security profile, you enable privileges to perform AppEnhancer functions. You can control the activities of users within applications by granting privileges only for necessary functions. Each security profile contains privilege settings for a variety of user functions, such as creating, modifying, and deleting applications; and scanning and printing documents. There are also settings for accessing commands on certain menus, such as Image Enhancement.

3.6.2.2 Application level security

Application level security grants users/groups access to applications. Profiles can be created to grant access to all applications or to specific applications.

Global security profiles can be established to automatically assign a uniform set of access privileges to a user or group of users each time a new application is created. When a global profile exists for a user or group of users, the privileges assigned in that profile are automatically assigned to each application created. If a user or group of users does not have a global profile, an application-specific profile must be created for that user or group before they can access a new application.

Application security profiles, like global security profiles, enable you to grant a particular set of privileges to a user or group of users. You can define different privileges for a user or group of users in each application. One group may have full privileges in an application (for example, HR), but only display privileges in the another application (for example, PAYROLL). Application security profiles also enable AppEnhancer system administrators (super users) to delegate responsibility for assigning/administering user privileges for a subset of applications, users, and permissions to one or more lower-level administrators or users. This enables lower-level administrators to add and delete users and groups, and assign user and group privileges, only within the domain of the specific AppEnhancer application(s), not globally across all AppEnhancer applications. Super users can also specify whether lower-level administrators can designate additional sub-administrators with an even smaller subset of responsibility—for example, the ability to assign user privileges for a single AppEnhancer application.

When a profile is application-specific, however, the privilege settings are not carried over to a new application when it is created. If a group of users has an application-specific profile for a certain application, and no global profile, then members of the group will not be able to access a new application when it is created.

Application-specific security settings override global security settings. For example, if a group of users has privileges to create documents in their global profile, and an application-specific profile is set up (for this group) that does not have create document privileges, the users will not be able to create documents in that particular application.

3.6.3   Document level security

The Document Level Security feature lets AppEnhancer system administrators protect particular documents in an application from access by unauthorized users, or enable users access to only particular documents in an application. AppEnhancer uses the values entered as the index for a document to achieve this protection. You can mark particular fields in an index as Document Level Security fields when an application is built. You can specify particular values in those index fields as inaccessible or accessible to groups of users. If a marked value is found, AppEnhancer either grants or denies access to the document with that index value based on the settings configured in Document Level Security.

For Document Level Security to be used for a field, you must enable the Document Level Security field flag during the field definition portion of application creation. To assign secured values, you form an association between a particular Document Level Security enabled field and a particular group of users, and then assign values for that field that either allow or deny access to the particular group of users. Document Level Security can be used to prevent a user from viewing certain documents in an application, assuming that they have display privileges in that application.

For example, if the field User Name is marked as a **Document Level Security** field, you can set `John Doe` as a value that is inaccessible to the user group Scan. Any user who is a member of the group Scan will not be able to access any documents that have the index value `John Doe` stored in the User Name field.

Document Level Security can be either accessible or inaccessible. After you have specified either accessible or inaccessible security, you can create a list of secured values. If the accessible security type is chosen, AppEnhancer enables the users in the selected group to access documents with index field values matching the secured value list (and only those documents). If the inaccessible security type is chosen, AppEnhancer does not enable the users in the selected group to access documents with index field values matching the secured value list (but enables access to all other documents).

📄 **Note:** If Document Level Security and the multiple indexes referencing a single document option have both been enabled for an application, and if a user with delete privileges can access a document through at least one of its index records, the user can delete that document even if Document Level Security does not enable the user to access the document through any of its other index records.

You can use Document Level Security to deny user access to selected documents, without restricting access to all documents in an application. Document Level Security can be configured to restrict based on keywords (by user name or workstation) or by index field criteria (which can employ wildcards).

3.6.3.1 Document level security keywords

Document Level Security Keywords can be used to restrict access to documents based on user name and/or workstation. To configure this for an actual application, fields should be set up to correspond to the keyword being compared. For instance, if you wanted an application to restrict access to documents based on user name, you would create an index field (such as User Name) to contain user names, and assign it the %u keyword on the **Document Level Security** page within AppEnhancer Administrator. Within the actual document indexes, the field User Name should be populated with an actual user name. When the user runs a query on the application with Document Level Security keywords set up for user, only those documents that contain their user name in the User Name index field will be retrieved. The same process would be followed by using the %w keyword in Document Level Security with a field designed to hold workstation names to restrict documents based on individual workstations.

3.6.3.2 Document level security wildcards

Wildcard characters can be used within Document Level Security secured values to restrict or enable access to a wider range of documents. The asterisk (*), when used in a secured value, replaces several characters and the question mark (?) replaces a single character. For instance, a secured value set as 1*1 could restrict all documents where the index field data begins and ends with a 1 (101, 10021, 1541, etc.), and a secured value set as 1?1 would restrict all documents where the index field data contains three digits and begins and ends with a 1 (101, 111, 121, and so on.).

3.6.4 Precedence of privileges for users and groups

If the user is a member of a group that has either a global security profile or an application security profile configured, the privilege settings for the group carry over to the user settings by default. When the data source is using the CM security provider, the functions that are enabled for a user due to group profiles are indicated in the user settings by a grayed out checkmark next to the item. Regardless of the security provider, the check boxes for these items are grayed out. You can choose to accept the group privileges or to override the group settings. Group settings can be overridden by enabling a privilege (selecting the check box) or by disabling a privilege (clearing the check box).

A privilege is enabled if the check box for an item contains a checkmark. If the check box is clear, the privilege is disabled. If the check box is grayed out, the user inherits the privilege setting from any groups in which the user is a member:

- When the data source is using the CM security provider, the **Profile** tab of the user indicates the inherited privilege setting for each privilege. If the check box is grayed out and has a checkmark next to it, the privilege is enabled. If the check box is grayed out with no checkmark, the privilege is disabled.

- When the data source is using the Windows security provider, the **Profile** tab of the user does not indicate the inherited privilege setting for each privilege. If the check box is dimmed and you want to determine what privilege setting has been inherited, you must refer to the **Profile** tab for the group.

3.6.5   Annotation security

You can use privileges to apply annotation-related security measures, so that annotations and redactions of any type can be created, modified, hidden, moved, or deleted only by users who have appropriate privileges. However, if you want to control access of users to specific annotations, you must use annotation groups. In addition, Rubber Stamp annotation security can be configured on user, group, and application levels.

3.6.5.1   Annotation group security

Annotation groups enable you to create associations between users, groups, and specific annotations. You can specify which users and groups can view or modify specific annotations, and which users and groups can hide or modify specific redactions. You can use AppEnhancer Administrator to create annotation groups and to populate them with existing users and existing groups.

If annotation groups are configured in AppEnhancer Administrator, the annotation privileges configured in a security profile of users may or may not be used, depending on how the annotation group security is configured. If the annotation group has **Follow Legacy Rules** enabled, the annotation privileges configured in the profile of users is honored. If **Follow Legacy Rules** is disabled, the annotation security properties set on the **Annotation Group** tab in AppEnhancer Administrator is honored, but not the properties configured in the user settings.

You can use privileges to apply annotation-related security measures. However, if you want to control access of the user to specific annotations, you must use annotation groups. You can use AppEnhancer Administrator to create annotation groups and also to add existing users and groups to the annotation groups.

> **Note:** Only users with the **User Security Maintenance** privilege can create, modify, or delete annotation groups.

3.6.5.2   Rubber stamp security

Rubber Stamp annotation security can be configured on user, group, and application levels.

> **Caution**
>
> Any application that is prefaced with an underscore (_RSTAMP, for example) is accessible only to users with AppEnhancer administrative privileges. For example, if you create an application in AppEnhancer Administrator named _MEDICAL, only users who have the Administrator privilege will be able to view and/or access the application.

3.7 Security mapping

When the AppEnhancer Migration service is used to migrate documents and security information, the administrator has the option to map users and groups in the source database to users and groups in the destination database.

Consider an AppEnhancer data migration example in which the only users who will need access to the destination database are STEPHANIE, WEI, and MARK, and those users do not exist in the source database. In this example, the administrator who performs the migration should select three user accounts in the source data source, the RUTH, JOHN, and REX user accounts, which have privileges that STEPHANIE, WEI, and MARK would need. The administrator should enable alternative security (security mapping) for the RUTH, JOHN, and REX user accounts and specify that those three users should be migrated as STEPHANIE, WEI, and MARK, with the appropriate passwords.

When an AppEnhancer data migration is performed, if the Use alternative security option is enabled in the wizard, only the users and groups with alternative security information configured are migrated.

> **Note:** The availability of the security mapping feature depends on the security provider in use by the destination data source and the source data source.

The following table describes whether users or groups can be mapped from one security provider to another:

Mapping from	Mapping to	
CM	**Windows**	
CM	Enabled	Not Enabled
Windows	Enabled	Not Enabled

3.7.1 Security limitations

The following tables describes the AppEnhancer security limitations when creating or importing users or groups:

Number of	Maximum
Groups per database	250,000
Users per database	250,000

3.8   Signing in to Web Access using Windows

1. In AppEnhancer Administrator, go to **Environment > Data Sources**.

2. Select **EDIT** to modify a data source.

3. From the **Security Model** list, select **Windows Security**.

4. Click **Save**.

5. Go to the **Application Management > <your data source> > Users** node.

6. Add user by providing details in the fields described in the following table. You can also log in as a domain user.

   📄 **Note:** You can assign the required privileges to the user or copy privileges of existing users.

Field	Description
User Name	Unique user name. The user name can be up to 64 characters.
	When the data source uses the Windows security provider, you must precede the user name with its domain name and a backward slash (\). The domain name can be up to 64 characters.
Full Name	Full name of the user. The full name can be up to 132 characters.
Password **Confirm Password**	Provide a password for the user account. The password can be up to 64 characters.
	In the **Confirm Password** text box, type the same password in exactly the same format.
	📄 **Note:** By default, the password and confirm password fields are automatically populated with a randomized password. Click **Show Password** to see the password and change it.
License Group	Select a license group from the list.
	The license group name can be up to 32 characters.

7. Open IIS Manager.

8. Go to the AppEnhancer Web Access web application. By default, it is `\\Default Web Site\AppEnhancer`.

9. Double-click **Authentication**.

   If you enable both **Anonymous Authentication** and **Windows Authentication**, anonymous authentication takes precedence over Windows authentication.

To automatically log in to AppEnhancer Web Access as a Windows user, disable **Anonymous Authentication**.

Chapter 4

Designing Applications

4.1 Introducing applications

To begin using AppEnhancer to store and manage documents, you must first design and then create applications in which to store your documents. Different applications can be created to meet different content management requirements.

When preparing to create an application, you should first establish a design plan. It is important to assess current and future user requirements, and then design an application to accommodate your users' needs. Although applications can be modified, careful planning reduces the need for future redesign.

Different applications can be created to meet different content management requirements. You can create the following types of applications:

- Normal: An application that does not have any retention requirements.

- Software Retention Management: An application that is enabled for AppEnhancer Software Retention Management. You must have the license feature installed.

- Predefined: An application that is accessible only by AppEnhancer system administrators. Predefined applications begin with an underscore.

  In AppEnhancer Administrator, you can create the predefined applications _FORMS and _RSTAMP that are part of the demonstration database by using the **CREATE _FORM APPLICATION** option and the **CREATE _RSTAMP APPLICATION** option, respectively, on the **Applications List** page. Creating these two predefined applications is the same as for creating new or custom applications, except that you must accept the default values. _FORMS application is used to store and manage form overlays for Reports Management documents. Each document in the _FORMS application consists of only one page and one version, containing the contents of the form as an ASCII or image file. _RSTAMP application enables you to store and manage predefined annotation types called rubber stamps. The rubber stamp annotation provides users the ability to place preset and custom text annotations on a page as well as image files supported by the AppEnhancer image library and embedded foreign files. Each document in the application consists of only one page and one version. You must assign each user the Display privilege either in the _RSTAMP application or in a global profile.

  > **Note:** By default, only AppEnhancer system administrators can access the _FORMS and _RSTAMP applications.

4.2   Understanding design considerations

Before you can create an application in AppEnhancer Administrator, develop a design plan for the application. A design plan can help prevent situations where an application does not meet the requirements of the intended users. Additionally, the application design determines, to a significant extent, the efficiency of data entry and document retrieval.

Notes

- If you want to perform a full-text search, you must use AppEnhancer full-text server.

- AppEnhancer data source and application names are used to encode disk path names. This requires that all configured path names are compatible with the host operating system. If you choose to use Chinese characters in data source, application, or path names, you must run all multibyte modules on Chinese operating systems.

4.2.1   Plan on index fields

The most important part of designing an application is planning the index fields that will be used to hold descriptive information for documents. First, you should evaluate the storage and retrieval needs of those who will be using the application, and then you set up field definitions for each index field. A data type can be chosen for each defined field, and each data type offers specific formatting options. Several different field flags can be enabled or disabled for each field to configure different field attributes.

With your particular storage requirements in mind, you should make decisions about the set of index fields for the application, such as:

- What information is needed (fields) to adequately describe each of the documents to be stored in the application

- Whether description (more fields) or quick index data entry (fewer fields) is more important

- Whether a field, or a combination of fields, should be chosen to make up unique identifiers for each document

You make choices for each individual field in the application as well, such as:

- What type of data each field can hold (such as text, integers, dates, social security numbers, and so on) and how that data is formatted

- Whether a value is required for each field

- Whether the values entered in a field can be modified after the initial data entry

- Whether the values in each field is searchable

- Whether data validation features should be used for each field to ensure accurate data entry

- Whether values for each field can be imported using a batch import or automatic data entry mechanism or whether they are entered one document at a time

- Whether the values entered into a field should be used as a basis for granting or denying access to the documents the values describe

> **Note:** AppEnhancer enables up to 64 index fields for each application.

4.2.2 Fields order for efficient data entry

Index field design substantially influences data entry and manipulation. Fields can be arranged in the index in a way that simplifies the entry of the index information for the user. Required fields should be grouped together in the index for the most efficient data entry.

4.2.3 Fields design to simplify data entry

When thousands of documents are added to an application, new document indexing can become a time-consuming task. AppEnhancer provides data type and field flag options that can help reduce the amount of time spent indexing documents. For example, the User-defined list and Boolean Choice data types enable users to pick an item from a list rather than typing it in. Use of these data types as index fields also ensures standardized data entry by preventing typographical errors or misspellings.

Additionally, enabling the Auto Index field flag makes it possible for the user to choose a record of index values from a table, rather than typing them in. Using the Key Reference feature lets the user populate index information based on the value of a key field.

4.2.4 Data Integrity precautions

Consider the potential for errors in data entry during document indexing during the application design process. For data types where specific data formats can be chosen, AppEnhancer automatically provides data format validation. AppEnhancer either reformats the values that were entered incorrectly, or does not accept the entered value. In addition, you can use some of the data formats and field flags in AppEnhancer to help ensure standardized data entry practices. For example, the Date Stamp and Time Stamp data types cause the current date or time to be automatically filled in. These are read-only values and cannot be modified by the user.

Additionally, enabling the Validation Mask field flag for a text field enables you to configure a template for data entered into the field. All values entered must conform to the character pattern established in the validation mask. This flag also enables you to hide confidential data in text, SSN, and telephone number index fields to prevent unauthorized users from viewing the data. Enabling the Part of Unique Key field flag for a field ensures that the combination of values entered into the field(s) is unique to the application. Enabling the Dual Data Entry field flag makes users enter the value for an index field twice to ensure data entry accuracy.

4.2.5   Customized data imports

When imports are performed using one of the three import wizards, AppEnhancer follows a set of rules called a *specification*. The specification informs AppEnhancer about the fields to be imported, the field order and data format, and the character (delimiter) that indicates that the information for one field has ended and the information for another field has begun.

A default specification exists for every delimiter type. Default specifications are already configured for use with the import features. When a user performs an import and chooses a default specification, AppEnhancer automatically looks for index information for every field that is available for that type of import, in the order that the fields are listed in the application. Users can employ the default specifications to import data into AppEnhancer as long as the data in the import file is formatted to work with the specification.

However, in certain situations, the default specifications do not work with an import. For example, the import file might have fields in a certain order that does not correspond with the order of fields in the application. For this reason, AppEnhancer provides tools to customize specifications. Administrator. AppEnhancer system administrators can either modify the existing default specifications or create new specifications.

An administrator can create a new specification to import data into only some of the available fields. An administrator can also change the order in which fields are imported. If field values are formatted in a different format than is configured for the field in the application, an administrator can configure the specification to convert the imported data to the correct data format.

4.2.6   Design limitations

The following table lists the limitations to consider when designing your AppEnhancer applications:

Number of items	Maximum
Applications per database	2,048
Batches per application	2,147,483,647
Documents per application	4,294,967,295
Documents per application in Demo mode	(set by Demo License)
Documents per application in Evaluation mode	1000
Page versions (BIN files) per application	2,147,483,647
Pages per document	250,000
Pages per batch	32,767
Paths per database	32,767

Number of items	Maximum
Versions per page	255

4.2.7 Outlining application design

This outline can help you through the process of application design. Take notes on the answers to each relevant question to use when creating your application. When the application design process is complete, you can specify the application name and description, indexing options, and field definitions in the AppEnhancer Administrator module, and you can build the application. To help you apply this conceptualization process to implementing the design, portions of the AppEnhancer Administrator application creation wizard illustrate relevant options relating to each question.

4.2.7.1 General application design questions

The first step in defining an application is to name the application, describe it, and enable or disable general indexing configuration options. To do this, answer the following questions about the overall design of the application.

- What will the name of the application be?

The application must have a unique name composed of up to 64 alphanumeric characters. Only users who have been given the Administrator privilege in AppEnhancer Administrator can access applications that begin with an underscore. Two applications within a single data source cannot have the same name. The following characters should not be used in application names: double quotation mark ("), single quotation mark ('), blank space, backslash (\), forward slash (/), period (.), comma (,), asterisk (*), pipe (|), semi-colon (;), colon (:), question mark (?), percentage sign (%), less than sign (<), and greater than sign (>). The application name may not begin with a number.

- What description should be given to the application?

The description should be formulated to aid users in identifying the purpose of the application. The description can be up to 128 alphanumeric characters. The following characters should not be used in application descriptions: double quotation mark ("), single quotation mark ('), and percentage sign (%).

- Should users be enabled to attach multiple index records to a single document?

Attaching multiple indexes to a single document enables users to classify the same document in more than one way. Take, for example, an application that will be used to store documents relating to individual people. If a document relates to more than one person, this feature can reduce the amount of storage space used by referencing the same document for each person (rather than storing several copies of the document, one for each person).

> **Note:** If document level security and the Multiple indexes referencing a single document option have both been enabled for an application, and if a user with

delete privileges can access a document through at least one of its index records, keep in mind that the user can delete that document even if DLS does not enable the user to access the document through any of its other index records.

- Do you need to configure the application to comply with HIPAA?

For each AppEnhancer application that contains data protected by HIPAA, it is recommended that you enable the Prompt for checkout when open documents, Checkout comments required, Checkin comments required, and Reason Code options. These options address the needs of privacy and security outlined by HIPAA by enabling you to identify which documents are being accessed, who is accessing them, and why the documents are being accessed.

4.2.7.2   General index design questions

After overall application design questions have been answered, the next step in designing an application is to choose index field attributes. Answering the following questions can help you to provide a framework for that design.

- Is description or quick index data entry more important?

When more fields are in an index of an application, more data entry is required for each document added to an application. If you are concerned about the amount of data entry required to add documents to an application, consider using as few fields as possible. Use of a limited number of fields also helps to reduce the time required to run a search for stored documents.

If it is important that documents are well-described, a few more index fields may be a good idea. Increasing the number of fields enables users doing document retrieval to be more specific in their search requests. In general, however, it is recommended that the number of index fields are kept to a minimum.

- What information will the users who retrieve documents from the application be most likely to know?

To prevent documents from being lost in an application over time, be sure to include at least one field that will contain values that are unlikely to change and that are likely to be known by the user. If documents will be stored relating to people, for example, a name field or a social security number field would be a good idea.

- Should a combination of fields be chosen to make up unique identifiers for each document?

The Part of Unique Key field flag can be used to mark fields as part of a group of fields that must contain, as a whole, a unique combination of values for each document. For example, a name field and a social security number field in an application both have the Part of Unique Key field enabled. A user indexes one document with the values `John Doe` and `111-22-2222`. When the user tries to index another document with the same two values, AppEnhancer rejects the index entries

as invalid. Any number and combination of the fields in an application can be enabled as Part of Unique Key.

4.2.7.3   Field design questions

After a list of the fields for an application has been developed, you can define the attributes of each field. Determining field attributes involves choosing a data type, a data format (when necessary), a field length (when necessary), and any appropriate field flags. Review the following list of questions for each field:

- What type of data will the field hold (such as text, integers, dates, social security numbers, and so on) and (if applicable) how will that data be formatted?

The data type determines the kind of data that can be stored in the field. Some data types, such as text, are inclusive of other data types; other data types, such as social security number, enable entry of only a narrow range of values. From the Data Type list box, you can choose any of the following field data types: Text, Integer, Decimal/ Numeric, SSN, Telephone, ZIP Code, Currency, Boolean Choice, Time, Time Stamp, Date, and User-defined List.

- If the data type requires that you specify a field length, what should it be?

For several data types, the field length is configured when the data format is chosen. However, for the data types listed in the following table, the field length must be specified.

The following table describes the maximum length that can be specified for each data type:

Data type	Maximum field length
Currency	38
Decimal/Numeric	38
Integer	10
Text	254

A field length should be long enough to accommodate reasonably long entries. The length of the field, however, determines the amount of storage space set aside each time a value is stored for that field. For a text field where only a single word is likely to be entered, for example, a field length of 100 would waste database space.

- Should the user be required to enter a value into the field when indexing a document?

Enabling the Required field flag for a field requires users to enter data into that field when documents are added to the application. AppEnhancer will not add the document until valid values have been entered for each required field during document indexing.

- Should the user be able to search the values stored in the field?

Enabling the Search field flag makes the values stored in the field searchable by users retrieving documents.

- Should the values entered in the field be modifiable after the initial data entry, or should they be read-only?

If the Read-Only field flag is enabled for a field, the user can enter a value into the field when a document is first added to an application. However, that value cannot be modified after the initial indexing process is complete.

- Should the values entered into the field be used as a basis for granting or denying access to documents?

The AppEnhancer Document Level Security feature enables an AppEnhancer system administrator to protect documents within an application from access by users. You can enable the Doc Level Security flag for fields, then designate values within those fields as accessible or inaccessible to certain users.

When deciding whether Document Level Security should be enabled for a field, consider whether the values to be entered in the field are likely to prove effective for managing document access. The contents of the field should either relate directly to the reason that document access is being restricted or should indicate in some way whether or not a document should be restricted. Managing Document Level Security for fields with a greater variety of information entered will be more time-consuming than managing fields with a smaller range of values.

- Should the field be part of a unique key for each document?

You can ensure that a unique key, or unique combination of index values, is assigned to only one document in an application by enabling the Part of Unique Key field flag for one or more fields in the application's index. If, for example, a name field and an account number field each have the Part of Unique Key field flag enabled, a user cannot enter the combination of John Doe and 12345 for more than one document.

- Should data validation features be employed to ensure accurate data entry for the field?

For some fields, it is crucial that data be entered accurately. Fields used to store the values that will be the primary tool for document retrieval are good candidates for data validation precautions. For example, the first search field in an index is likely to contain data that will be frequently used to retrieve documents. Several field flag and field formatting options can be used to help guarantee that mistakes are not made during data entry.

The first step in building data integrity precautions into a field definition involves choosing a data type. Several of the data types available for fields in AppEnhancer have preset data formats. The Integer, Decimal/Numeric, Date, Time, SSN, Telephone, ZIP Code, and Currency data types are all preset. When you choose one of these data types, AppEnhancer will not accept values that are not the correct type of data, that cannot be formatted to match the data format, or that exceed the length

of the field. AppEnhancer also reformats valid data entered to the correct display format.

The Boolean Choice and User-defined List data types can also be used to ensure accurate data entry. Users enter values into a field of either of these data types by choosing an item from a drop-down list, eliminating the possibility of typographical errors. Whenever the same entries will repeatedly be made in a field, you should consider using a user-defined list to minimize the potential for user error.

Field flags can also be set to help validate data entry. The Time Stamp field flag, which can be set for fields with the Time data type, and the Date Stamp field flag, which can be set for fields with the Date data type, automatically enter the current time and date. This feature makes data entry for those fields unnecessary. When a field has a Text data type, you can enable the Validation Mask field flag and then create a validation mask, or template, for data entered into the field. AppEnhancer then checks any data entered into the field against that mask. Enabling the Dual Data Entry flag for a field forces users to enter data twice for the field.

The Validation Mask field flag also lets you hide all or part of a text, SSN, or telephone number field from view. With the exception of the AppEnhancer Document Index view, masked field values are not visible in AppEnhancer Web Access regardless of user permissions. An asterisk (*) appears in place of each masked character in the index field. However, AppEnhancer users with Modify Index privilege can view and modify index values for masked fields in the Document Index view.

When the logged in user does not have the Modify Index privilege, the masked portion of the index field value is hidden in the Document Index view in addition to the Result Set view.

- Can values for the field be imported using a batch import mechanism or will they be entered one document at a time?

If the values can be imported, consider using a Key Reference Import Wizard or an Auto Index Import Wizard.

With the Key Reference feature, the user enters a unique piece of data into the key reference file field, then presses the `Tab` key to populate the data reference file fields for the document. AppEnhancer fills the data fields with values from the index record with that key value in the Key Reference table. To enable the Key Reference feature for a field, you must enable the Key Reference field flag if the field will be a Key field or choose the Data Reference field flag if it will be a Data field.

The Auto Index field flag lets users or administrators import index values for Auto Index enabled fields. AppEnhancer takes the imported values and builds a table of index records. Users can then invoke the Auto Index function by pressing the `F7` key during the indexing process and picking an index record from the table. When a user picks a record from an Auto Index table, that index record is deleted from the table. To enable the Auto Index feature for a field, you must enable the Auto Index field flag.

4.2.8   Application examples

This section provides information on application examples.

4.2.8.1   Document level security for employee records

This section describes the use of the Doc Level Security field flag, the Part of Unique Key field flag, the User-defined List data type, and the Key Reference Import Wizard. A human resources department for a mid-sized company must track several different kinds of paperwork for each employee:

- Health insurance registration forms

- Tax forms

- Employee profiles (which might include a résumé and a picture of the employee)

- Performance review forms

- Registration forms for the 401(k) plan of the company

- Employee commendations or complaints

The AppEnhancer system administrator designing this application anticipates a large initial data entry process (to store all existing paper documents online), followed by occasional addition of pages to documents of employee's document and periodic addition of documents relating to new employees. To ease the data entry process, the administrator wants all paperwork relating to a single employee to be stored in a single document. However, after the security for the application has been planned, the administrator realizes that some of the paperwork to be stored, such as performance reviews and employee complaints or commendations, should be seen only by senior human resources employees. The administrator decides that each employee should have a performance records document and a personal records document, and that all paperwork relating to the employee will be stored as pages in one of the documents of the employees.

The administrator creates a field called TYPE OF EMPLOYEE RECORDS with a user-defined list data type. The administrator puts two items, PERFORMANCE RECORDS and PERSONAL RECORDS, in the list. The administrator makes the field a search field to enable a user to search for all performance documents or all personal documents. To ensure that only one personal and only one performance document will be created for each employee, the administrator enables the Part of Unique Key field flag for the field. Finally, the administrator enables Document Level Security for the field. The value PERFORMANCE RECORDS will be marked as inaccessible for a group of human resources employees responsible for keeping track of employee records such as insurance records and tax records. The members of this group will have access to all PERSONAL RECORDS documents in the application. The group including only the top-level human resources officers will have access to all documents in the application.

The remainder of the index for this application is fairly simple: SOCIAL SECURITY NUMBER, NAME OF EMPLOYEE, and DEPARTMENT. To ensure that only two

documents will be created for each employee, the administrator enables the Part of Unique Key field flag for both the TYPE OF EMPLOYEE RECORDS field and the SOCIAL SECURITY NUMBER field. This guarantees that only one PERFORMANCE RECORDS entry and only one PERSONAL RECORDS entry can be chosen in combination with each social security number.

The administrator has a spreadsheet of employee names, social security numbers, and departments. The administrator decides to export that information and import it into AppEnhancer for use in indexing documents. After considering the import options, the administrator decides that the Key Reference Import option would be best. The administrator enables the Key Reference field flag for the SOCIAL SECURITY NUMBER field and the Data Reference field flag for the EMPLOYEE NAME and DEPARTMENT fields.

By importing the information by using Key Reference, the administrator makes it easier to manage employee information in the future. The administrator can create a Key Reference table containing index records for each employee. When the user adds two documents each to the employees, AppEnhancer will use the social security number entered to locate the corresponding employee name and department information. The two data fields will be automatically populated with the information stored in the table for each document. If the name of the employee or department changes in the future, the administrator can update the information in the index of one document and, because that information is stored in the Key Reference table and also used to populate the index for the other document of employee, the information will be updated in that index as well.

The SOCIAL SECURITY NUMBER field is used as the primary search field for the application, because the number is the piece of index information most likely to be known by a user retrieving documents. Additionally, placement of the social security number first will facilitate use of the Key Reference Import Wizard, because the SOCIAL SECURITY NUMBER field is the key field. The users can enter the key field value, press `Tab`, and populate the data fields immediately. This saves the user from using additional keystrokes to move to the key field.

The following table summarizes the field names, data types, data formats and field flags used for the application:

Field name	Data type	Length	Format	Field flags
SOCIAL SECURITY NUMBER	SSN	11 (defined by data type)	nnn-nn-nnnn	Required, Search, Part of Unique Key, Key Reference
EMPLOYEE NAME	Text	40	NA	Required, Search, Data Reference

Field name	Data type	Length	Format	Field flags
TYPE OF EMPLOYEE RECORDS	User-defined List	19 (defined by user list)	Choice of PERFORMANCE RECORDS and PERSONAL RECORD	Required, Search, Part of Unique Key, Doc Level Security
DEPARTMENT	User-defined List	(defined by user list)	Choice of each of the departments in the company	Required, Search, Data Reference

4.2.8.2   Customer information import

The application in this example imports index information and is also designed to receive index information exported from another system. In this case, however, the Auto Index Utility is used. This scenario illustrates a situation in which Auto Index Import is preferable to a Key Reference Import.

A credit union decides to use AppEnhancer to store information for customers who have taken out loans. The customer account numbers, names, and social security numbers are stored in a database on the mainframe of the credit union. For some of the customers, but not all, there is also a loan type designation stored in the database. All of this information will be exported from that database and imported into AppEnhancer.

Only loan officers will access the application, so they decide that there is no need for Document Level Security. However, the credit union manager wants the application to be protected from access by any other users on the network. The AppEnhancer system administrator creates a profile for the application that filters out all privileges to the application for the group called Everyone. The administrator then creates a Loan Officers group that grants the necessary privileges to the application for those employees.

The administrator names the application LOANS, and describes it as holding loan information. The mainframe database can export information in whatever order is needed, so that the administrator decides that the field order for the application should be ACCOUNT NUMBER, NAME, SOCIAL SECURITY NUMBER, and then LOAN TYPE. The administrator makes each of the fields a search field and enables the Auto Index field flag for each. The administrator flags the ACCOUNT NUMBER, NAME, and SOCIAL SECURITY NUMBER fields as required and chooses the Integer data type for the ACCOUNT NUMBER field. The field length for that field is set to 8 digits to help prevent incorrect data entry because account numbers are only eight digits long. Although it is recommended that the LOAN TYPE field be a user-defined list, the entries from the previous database vary considerably, and therefore it is marked as a text field.

The following table lists the fields and field attributes for the LOANS application:

Field name	Data type	Length	Format	Field flags
ACCOUNT NUMBER	Integer	8	nnnn	Required, Search, Auto Index
NAME	Text	40	N/A	Required, Search, Auto Index
SOCIAL SECURITY NUMBER	SSN	11 (defined by data type)	nnn-nn-nnn	Required, Search, Auto Index
LOAN TYPE	Text	30	N/A	Search, Auto Index

4.2.8.3 Litigation database import

This scenario illustrates a situation where the Index Image Import service would be useful. This example illustrates how an AppEnhancer system administrator can plan an application to accommodate the import of indexes and images from another software program, and also demonstrates use of the Doc Level Security field flag and the User-defined List data type.

The litigation group in a law firm is working on a paper-intensive litigation case. When the case first starts, the group decides that online document storage is the best way to manage the thousands of documents that they expect to use as evidence in the case. The litigation support manager asks a consultant to design a proprietary database where they can store index information for the documents.

The consultant sets up a system using two different types of software, one for viewing images and one for storing index information. The proprietary system enables the group to search for index information and link to the related image. They index and store half the documents for the case using this system. Unfortunately, as more and more documents are added to the system, managing the system becomes more and more unwieldy, until finally the structure of the system proves to be too inflexible for the needs of the group.

The litigation support manager for the group (administrator), when learning about the combination of storage and viewing capabilities in AppEnhancer, decides to switch over to AppEnhancer. The last thing the administrator wants, however, is to have to re-index and store the documents already stored in the proprietary database. The administrator wants to export the documents and index information from the proprietary database, and import them into AppEnhancer with the correct index values attached to the correct documents.

The data stored in the proprietary database is stored in an index structure similar to the one in AppEnhancer, but it includes a field that contains a value indicating the location of the image file. The administrator decides that the fastest way of moving the data from the database into AppEnhancer is to export all of the index data from the existing database. The resulting text file can then be edited to fit the format model needed to do an Index Image Import into AppEnhancer.

The administrator names the application DOEVDOE to indicate the name of the case. For the description, the administrator chooses the full name of the case. The design of the index for this application is based on the index in the existing application, which contains several index fields: DESCRIPTION, PRODUCED BY, PROTECTED, KEYWORDS, DOCUMENT TYPE, AUTHOR, RECIPIENT, DATE OF DOCUMENT, and DOCUMENT ID. All of these fields will be defined as AppEnhancer index fields except for the PRODUCED BY and DOCUMENT ID fields. All fields will be marked as search fields, and the first two fields will have the Required field flag enabled. The KEYWORDS and DOCUMENT TYPE fields both have a User-defined List data type. The PROTECTED field has a Boolean Choice data type to help ensure accuracy of data entry for these fields.

The images to be imported for the application are named by document ID. To prepare for an Index Image Import, the litigation manager exports all of the information in the existing database as records with the fields delimited (separated) by commas. The document ID field is exported as the last field in each record. To import the data using Index Image Import, the manager edits each of the exported document ID fields to include the following:

- An @ symbol (which informs AppEnhancer that an image location follows)
- The volume label for the media where the image is located
- The directory path to the file
- A file extension

Security for this application will be somewhat complex. Users accessing the application will include paralegals and attorneys. Typically secretaries have privileges to display, print, and fax documents within the AppEnhancer system of the law firm, but they will not have privileges to perform any functions in this application. The administrator creates an application security profile with no privileges enabled and associates the Secretaries group with the profile.

Because of the highly sensitive information for this case, only certain attorneys and high-level paralegals can view documents that have a Yes value in the PROTECTED field. To ensure that access to these documents is controlled, the administrator enables Document Level Security for the PROTECTED field and then denies access to the value Yes for a group of users who are not enableed to access protected documents by using the Document Level Security field flag within the application (found in AppEnhancer Administrator).

The following table summarizes the field names, data types, data formats, and field flags that will be used for the application:

Field name	Data type	Length	Format	Field flags
DESCRIPTION	Text	50	N/A	Required, Search
PROTECTED	Boolean Choice	3 (defined by choice list)	Choice of YES or NO	Required, Search, Doc Level Security

Field name	Data type	Length	Format	Field flags
KEYWORDS	User- defined List	(defined by user list)	Choice of each of the keywords used to code the documents	Search
DOCUMENT TYPE	User- defined List	(defined by user list)	Choice of several different document types	Search
AUTHOR	Text	40	N/A	Search
RECIPIENT	Text	40	N/A	Search
DATE OF DOCUMENT	Date	8 (defined by data type)	mm-dd-yyyy	Search

4.2.8.4 Data entry validation for accounts payable

This scenario shows how field flags are used to ensure data entry accuracy. Specifically, this example illustrates the AppEnhancer field attributes that can be used to control data entry accuracy, including the Dual Data Entry field flag and the Validation Mask field flag.

An accounts payable department wants to set up an invoicing system in AppEnhancer so that they can scan all documents associated with an invoice (invoices, purchase orders, checks received, and so on). Because this invoice number controls the flow of invoices through the company, they are concerned that the number is entered accurately.

The name of the application is INVOICE, and they describe it as an invoice tracking application. The index for this application contains two fields: INVOICE NUMBER and CUSTOMER NAME.

For the INVOICE NUMBER field, because it is critical that the data is entered accurately, they enable both the Validation Mask field flag and the Dual Data Entry field flag. Every invoice number starts with three letters, which are followed by five numbers, so they put the following validation mask on the field: aaannnnn. They also enable the Dual Data Entry field flag to force the users to enter the invoice number correctly twice. In addition, they set the field length to 8 digits to prevent invoice numbers that are too long.

The following table summarizes the fields and field attributes for the application:

Field name	Data type	Length	Format	Field flags
INVOICE NUMBER	Text	8	aaannnnn	Required, Search, Validation Mask, Dual Data Entry
CUSTOMER NAME	Text	40	N/A	Required, Search

4.2.9   Understanding field attributes

When designing an application, you must enter the name, type, and length of each index field. The field name can be up to 64 alphanumeric characters. The first character must be a letter of the alphabet; it may not be a number, blank space, or symbol. The double quotation mark ("), single quotation mark ('), and backslash (\) characters should not be used in field names.

4.2.9.1   Data types

AppEnhancer supports many data types and provides standardized formatting options for each data type. For example, some formats insert special characters, such as hyphens, in a social security number. Several of the data types available for index fields in AppEnhancer have preconfigured formatting choices. When you add a field with one of these data types, the Format list becomes active. The list contains all available predefined formats. You can also create custom formats for data types.

Integer, Decimal/Numeric, Date, SSN, Telephone, ZIP Code, Currency, and Boolean Choice are the data types that activate the Format list. When you select a predefined format for all of these data types except Boolean Choice, values entered for the data type are automatically converted to the applicable format. Users choose items from a list for Boolean Choice data type fields and no reformatting is necessary. The data types that do not activate the Format list are as follows: Text (unless the Validation Mask field flag is enabled for the field), Time, Time Stamp, and User-defined List.

The Time and Time Stamp data types each have only one predefined format available, which you select by selecting the data type.

If the preconfigured field data types and formats available to you in AppEnhancer are not adequate for your organization needs, you can add new types. You can add completely new data types and accompanying formats, or you can add additional data formats for existing data types. The User Security Maintenance privilege is required to create, modify, or delete custom data types and custom data formats.

> **Note:** The Migration Wizard cannot migrate custom data types or formats. If you want to migrate an application that has custom data types or formats, you must recreate the custom data type or format in the destination application before performing the migration.

4.2.9.1.1   Text data type

The Text data type is used to store any combination of up to 254 alpha/numeric characters.

Item	Description
Maximum Length	254 characters.
Format	Any alpha/numeric characters.
Prohibited Characters	The index values entered in this field cannot contain question mark (?) or asterisk (*) characters. These characters are reserved for wildcard searches.
Available Field Flags	Required, Search, Read-Only, Doc Level Security, Part of Unique Key, Dual Data Entry, Key Reference, Data Reference, Auto Index, and Validation Mask.
Automatic Formatting	If you apply the Validation Mask flag to a text field, AppEnhancer permits the user to enter only values that match the validation mask. Otherwise, no automatic formatting is performed.

4.2.9.1.2   Integer data type

An integer is a whole number and can contain up to 10 numeric characters. Integers can be stored with or without commas. Parentheses can be used to indicate negative numbers. If the parentheses are not used, negative numbers will appear with a minus sign. To store longer whole numbers than are allowed in the Integer data type, use the decimal/numeric data type instead and choose the format with no decimal point.

Item	Description
Maximum Length	10 digits (The range of values that a user may enter into an integer field is from -2,147,483,648 to 2,147,483,647.).
Format	Whole numbers.
Available Formats	You can select a format that combines any of the following: • With or without commas • Negatives shown with or without parentheses
Available Field Flags	Required, Search, Read-Only, Doc Level Security, Part of Unique Key, Dual Data Entry, Key Reference, Data Reference, Auto Index, and Leading Zeroes.

Item	Description
Automatic Formatting	AppEnhancer does not allow the user to enter a decimal value. Also, depending on the format you select: • AppEnhancer inserts or strips commas. • AppEnhancer converts a minus sign to parentheses or vice versa. • If you apply the Leading Zeroes flag to an Integer field, and the user enters the necessary leading zeroes, AppEnhancer preserves the zeroes.

4.2.9.1.3   Decimal/numeric data type

The Decimal/Numeric data type is used to store numbers that may or may not include decimals. The number of places allowed in the decimal portion of the number is configurable. If the format without a decimal point is chosen, no decimal point will display in the stored index information for the field. Numbers can be stored with or without commas. Parentheses or a minus sign are used to indicate negative numbers. If a data format is chosen that uses parentheses, all negative numbers entered will appear in the index field with parentheses. If a data format is chosen that uses a minus sign, negative numbers will appear with a minus sign.

Item	Description
Maximum Length	The maximum length depends on the database software: • 38 digits in Microsoft SQL Server, MySQL, PostgreSQL, and Oracle databases
Format	Whole numbers or numbers with decimal places.
Available Formats	You can select a format that combines any of the following: • With or without commas • Negatives shown with a minus sign or with parentheses • Whole numbers or decimals to 5 places
Available Field Flags	Required, Search, Read-Only, Doc Level Security, Part of Unique Key, Dual Data Entry, Key Reference, Data Reference, and Auto Index.

Item	Description
Automatic Formatting	If the user enters a value with greater precision than the format allows (that is, if the value has too many digits after the decimal), the value will not be rounded. For example, if the decimal/numeric data type is formatted as nnnn.n, and 1.99 is entered, a warning appears and only 1 would be allowed to be saved.
	Also, depending on the format you select:
	• AppEnhancer inserts or strips commas.
	• AppEnhancer converts a minus sign to parentheses or vice versa.
	• AppEnhancer adds zeroes after the decimal (if necessary to store a value with the appropriate number of decimal places).

4.2.9.1.4   Date data type

Many different formats are available for the storage of dates. To have months appear with a three-character abbreviation (such as JAN), select an mmm month format.

Item	Description
Length	Not configurable (automatically set to format selected).
Format	Numeric.
Available Formats	You can select a format that combines any of the following:
	• Month, day, and year in any order
	• 2-digit or 4-digit year
	• 3-character abbreviated month (for example, JAN) or 2-digit month (for example, 01)
	• Dashes or slashes
	Two additional formats spell out the month and use spaces and a comma:
	• mmmm dd, yyyy (for example, March 26, 2001)
	• dd mmmm, yyyy (for example, 26 March, 2001)

Item	Description
Available Field Flags	Required, Search, Read-Only, Doc Level Security, Part of Unique Key, Dual Data Entry (unless Date Stamp is used), Key Reference, Data Reference (unless Date Stamp is used), Auto Index, and Date Stamp.
Automatic Formatting	If you select a format that uses 3-character months, and the user enters a 3-character month in lower case, AppEnhancer converts it to uppercase.
	If you apply the Date Stamp flag to a date field, AppEnhancer automatically stores the date of entry in the selected format.
	Otherwise, depending on the format you select:
	• AppEnhancer converts a 2-digit year to a 4-digit year or vice versa. The parameters AppEnhancer uses to do the conversion can be set on the **Data** tab of the Configuration dialog box.
	• AppEnhancer converts a 3-character month to a 2-digit month or vice versa. For example, AppEnhancer converts FEB to 02.
	• AppEnhancer converts dashes to slashes or vice versa.

4.2.9.1.5   Time data type

In fields with the Time data type, values can be entered in only one format (hh:mm:ss).

📄 **Note:** When a user enters a time into a time field, the time will be automatically converted to military time. (For example, 1: 00: 30 p.m. will be stored as 13: 00: 30.) When the user saves the index, the display of the time value will be in the military time format.

Item	Description
Length	Not configurable (automatically set to 8 digits).
Format	Numeric values for hour, minute, and second, where the hour is expressed in terms of a twenty-four hour clock.
Available Formats	hh:mm:ss format only.

Item	Description
Available Field Flags	Required, Search, Read-Only, Doc Level Security, Part of Unique Key, Dual Data Entry (unless Time Stamp is enabled), Key Reference (unless Time Stamp is enabled), Data Reference (unless Time Stamp is enabled), Auto Index, and Time Stamp.
Automatic Formatting	If you apply the Time Stamp flag to a Time field, AppEnhancer automatically stores the system time of the workstation creating the document in the hh:mm:ss format. Otherwise, if a user enters a time in the format hhmmss (without colons) AppEnhancer inserts colons.

4.2.9.1.6 Time stamp data type

When the Time Stamp data type is selected, the index field is automatically populated during index creation using the system time. Time Stamp field values cannot be changed; they are added as read-only (yyyy-mm-dd hh:mm:ss).

Item	Description
Length	Not configurable (automatically set to format selected).
Format	System date of the workstation creating the document, where the hour is expressed in terms of a twenty-four hour clock.
Available Formats	yyyy-mm-dd hh:mm:ss format only.
Available Field Flags	Search, Doc Level Security and Part of Unique Key.
Automatic Formatting	AppEnhancer automatically stores the date and time of entry in the yyyy-mm-dd hh:mm:ss format.

4.2.9.1.7 SSN data type

When the social security number (SSN) format with dashes is selected, AppEnhancer automatically enters the dashes during index creation. When the format without the dashes is selected, AppEnhancer will strip any dashes entered during index creation.

Item	Description
Length	Not configurable (automatically set to format selected).
Format	Integers.

Item	Description
Available Formats	You can select a format with or without hyphens. You can also select a format that includes field display mask characters, which enable you to hide some or all of the data from unauthorized users. You can choose: • ddd-dd-nnnn, which displays output similar to ***-**-1234 • dddddnnnn, which displays output similar to *****1234
Available Field Flags	Required, Search, Read-Only, Doc Level Security, Part of Unique Key, Dual Data Entry, Key Reference, Data Reference, and Auto Index.
Automatic Formatting	Depending on the format you select, AppEnhancer inserts or strips hyphens.

4.2.9.1.8   Telephone data type

Telephone number values can be stored with or without an area code. Dashes and parentheses in telephone numbers are automatically added or stripped by AppEnhancer during index creation, in accordance with the format selected.

Item	Description
Length	Not configurable (automatically set to format selected).
Format	Numeric.
Available Formats	Any combination of the following: • With or without area code • Area code separated by parentheses or dash • Area code and parenthesis separated by a space or not You can also select a format that includes field display mask characters, which enable you to hide some or all of the data from unauthorized users. You can choose: • nnn-ddd-dddd, which displays output similar to 123-***-**** • (nnn)ddd-dddd, which displays output similar to (123)***-**** • (nnn) ddd-dddd, which displays output similar to (123) ***-****

Item	Description
Available Field Flags	Required, Search, Read-Only, Doc Level Security, Part of Unique Key, Dual Data Entry, Key Reference, Data Reference, and Auto Index.
Automatic Formatting	If necessary, AppEnhancer inserts hyphens to match the selected format. If you select a format without an area code, AppEnhancer enables the user to enter only seven digits. If you select a format with an area code, and the user enters only seven digits, an error appears. Also, depending on the format you select: • AppEnhancer converts parentheses to a hyphen or vice versa. • AppEnhancer inserts or strips a space between area code and number.

4.2.9.1.9 ZIP code data type

You can choose whether to allow storage of the additional four digits of ZIP codes. When the long ZIP code format is selected, index validation will fail if the last four digits are not explicitly entered.

Item	Description
Length	Not configurable (automatically set to format selected).
Format	Numeric.
Available Formats	You can select a format with or without a four-digit extension. The format with the extension has a dash to separate the extension from the main part of the ZIP code.
Available Field Flags	Required, Search, Read-Only, Doc Level Security, Part of Unique Key, Dual Data Entry, Key Reference, Data Reference, and Auto Index.
Automatic Formatting	If you select the format with the four-digit extension and the user enters only the five-digit ZIP code, AppEnhancer responds with an error message to prompt the user to enter the four-digit extension. If you select the format without the four-digit extension, AppEnhancer does not permit the user to enter more than five digits.

4.2.9.1.10   Currency data type

Currency values can be stored with or without decimal places and commas. If a data format is chosen that uses parentheses, all negative amounts entered will appear in the index field with parentheses. If a data format is chosen that does not use parentheses, negative amounts will appear with a minus sign.

> **Caution**
>
> For currency data types, values will not be saved if the user enters data more than two decimal places. For example, if 9.689 is entered, a warning appears. Numbers will not be automatically rounded by AppEnhancer for currency data types.

Item	Description
Maximum Length	The maximum length depends on the database software: • 38 digits in Microsoft SQL Server, MySQL, PostgreSQL, and Oracle databases
Format	Numeric.
Available Formats	You can select a format that combines any of the following: • Negatives shown with minus sign or with parentheses • With or without 2 decimal places • With or without commas
Available Field Flags	Required, Search, Read-Only, Doc Level Security, Part of Unique Key, Dual Data Entry, Key Reference, Data Reference, and Auto Index.
Automatic Formatting	AppEnhancer always inserts a dollar sign. If the user enters data that extends to three decimal places, a warning appears and only two will be saved. Also, depending on the format you select: • AppEnhancer converts a minus sign to parentheses or vice versa. • AppEnhancer adds zeroes after the decimal (if necessary to store a value with 2 decimal places). • AppEnhancer inserts or strips out commas.

The Currency data type has several characteristics that do not apply to the other data types. These characteristics include the following:

- The Format Name, Scale, and Width are automatically specified.

- The currency symbol is entered in the Formatting 1 string field.

- The Formatting 2 string enables you to override the use of thousand separators and parenthesis. Typically, the format type or locale setting is followed. This field enables you to override the use, but not the symbols (symbols come from the locale setting).

- Entering a string that contains a comma sets the use of thousand separators on.

- Entering a string that does not contain a comma sets the use of thousand separators off.

- Entering a string that contains a parenthesis sets the use of parenthesis on.

- Entering a string that does not contain a parenthesis sets the use of parenthesis off.

- Validation Expression and Data Conversion Expression are ignored for currency custom format types.

4.2.9.1.11   Boolean choice data type

Boolean choice fields build a list box of two mutually exclusive options. During index creation, you select one of the two.

If the Boolean choice field is not flagged as a required field, AppEnhancer Administrator will insert a null value, which users can choose rather than either of the configured options. If the field is required, the null value is not added, and the list box will contain only the selected values.

If the choice that you want is not listed in the Format list for Boolean choice, you can create a User-defined List that contains the two items.

Item	Description
Length	Not configurable (automatically set to length of longest item in list).
Format	Creates a list box on data entry and search screens with a pair of choices.
Available Formats	Yes/No, True/False, On/Off, In/Out, Male/Female, Exempt/Non-exempt, Asset/Liability, Income/Expense, or Receivable/Payable.
Available Field Flags	Required, Search, Read-Only, Doc Level Security, Part of Unique Key, Data Reference, and Auto Index.
Automatic Formatting	The user must select a value from a list.
Default Value on Search Screens	A wildcard (*), which represents all values.

4.2.9.1.12 User-defined list data type

You can configure a drop-down list that contains all possible values for a field. Items can be imported from a text file and can be rearranged in the list.

If the User-defined List field is not flagged as a required field, AppEnhancer Administrator also inserts a null value, which users can choose rather than any of the configured options. If the field is required, the null value is not added, and the list box will contain only the configured values.

Item	Description
Length	Not configurable (automatically set to length of longest item in list (132 characters maximum)).
Number of User-Defined Lists that Can Be created	Unlimited, but a large number of items (more than 400) in a user-defined list adversely affects performance. Also, the effect is cumulative. For example, if an application has three user-defined list fields, each of which has 200 items, then the effect is equivalent to having one user-defined list field with 600 items.
Format	Creates a list box on data entry and search screens with all entries specified.
Available Field Flags	Required, Search, Read-Only, Doc Level Security, Part of Unique Key, Key Reference, Data Reference, and Auto Index.
Automatic Formatting	User must select a value from a list.
Default Value on Search Screens	A wildcard (*), which represents all values.

4.2.9.1.13 Configuring index fields in a non-English environment

This section describes issues that you might need to consider if you are configuring index fields in a non-English environment.

4.2.9.1.13.1 Configuring numeric data types

Numeric data types include Decimal/Numeric, and Currency. The display of numeric values is composed of several different formatting variations, such as whether a character is used to group each three digits and which character is used to indicate negative values. Some of these variations are determined by the data format selected for the index field in AppEnhancer Administrator. Others are determined by the database locale setting.

If the database locale is not specified, some variations are determined by the product (AppEnhancer Web Access), the browser locale setting, and/or the installation type (English or French). Currently, the browser locales that may be set from the browser are French, German, Spanish, and English languages (English-USA is the default language for unsupported ones).

The following table indicates which settings determine how numeric fields are displayed:

Formatting variation	Example	Display format settings
Negative symbol	(4321) vs. -4321	Determined by data format (if you are using a custom currency data format, determined by **Formatting 2** specification)
Position of negative symbol	-4321 vs. 4321-	Always on the left
Separator usage	4,321 vs. 4321	Determined by data format (if you are using a custom currency data format, determined by **Formatting 2** specification)
Separator character	4,321 vs. 4 321	• If a database locale has been specified, determined by database locale • If a database Locale has not been specified, in AppEnhancer Web Access, determined by browser locale setting
Precision	321 vs. 321.00	Determined by data format (If you are using a custom currency data format, determined by Scale specification)
Decimal character	321.00 vs. 321,00	• If a database locale has been specified, determined by database locale • If a database locale has not been specified, in AppEnhancer Web Access, determined by browser locale setting

📄 **Note:** When an aspect of display formatting is determined by a database or browser locale selection, keep in mind the following points:

• In AppEnhancer Web Access, the display formatting reflects the default for the selected locale on the AppEnhancer Web Access Server.

4.2.9.1.13.2   Configuring currency data type

Currency data types have formatting variations in addition to the formatting variations for numeric data types. For example, values in an index field that use the Currency data type may have the currency symbol to the right of the value or to the left. These variations can be determined by the database Locale setting.

However, it is highly recommended that you do not select the database locale option to avoid inaccurate currency changes. If you want to use the euro (€) or franc (F) currency symbol, use a custom currency data format that specifies the currency symbol for all AppEnhancer Web Access clients.

The following table indicates the settings the determine how currency fields are displayed:

Formatting variation	Examples	Display format settings
Currency Symbol	$ vs.€	- If a database locale has been specified and you are using a prepackaged currency data type, determined by the regional settings on the installed machine for the locale specified in the database - If a database locale has not been specified and you are using a custom currency data format, determined by **Formatting 1** specification - If a database locale has not been specified and you are using a prepackaged currency data type, always a dollar ($) sign
Position of currency symbol	$1234 vs. 1234$	- If a database locale has been specified, determined by database locale - If a database locale has not been specified, in AppEnhancer Web Access, determined by browser locale setting

4.2.9.1.13.3 Configuring boolean choice data type

Values for the Boolean data type are stored in the database in English, regardless of the language in which they are displayed. If you want Boolean data type values to be stored in a language other than English, you must create custom Boolean data formats.

The display language for these values is determined by the following settings:

- If the French, German, or Spanish database locale has been specified, these values are displayed in French, German, or Spanish, respectively.

- If any other database locale has been specified, these values are displayed in English. Currently, only the English, French, German, or Spanish locales are supported.

- If a database locale has not been specified, in AppEnhancer Web Access, these values are displayed in the language indicated by the browser locale.

4.2.9.1.13.4 Configuring date data type

For the Date data type, the data format determines most variations in the display of the date value. The database locale setting determines the long month name and the short month name. The following table indicates which settings determine how date fields are displayed:

Formatting Variation	Examples	Display Format Settings
Delimiter	31-12-99 vs. 31/12/99 vs. 31December, 1999	Determined by data format
Order	31/12/99 vs. 31/99/12 vs. 12/31/99 vs. 12/99/31 vs. 99/31/12 vs. 99/12/31	Determined by data format
Length of year	31/12/99 vs. 31/12/1999	Determined by data format (yy=99; yyyy=1999)
Length of month	31/12/99 vs. 31/Dec/99 vs. 31December, 1999	Determined by data format (mm=12; mmm=Dec; mmmm=December.)
Language used for full and abbreviated month names	31December, 1999 vs. 31Décembre, 1999	• If a database locale has been specified, determined by database locale • If a database locale has not been specified, in AppEnhancer Web Access, determined by browser locale setting

4.2.9.2   Setting field flags

You can set a variety of field flags for each index field. When you are defining the application in AppEnhancer Administrator, the flags for the current index field being created or modified are displayed. You can configure different combinations of field flags for each field in an application.

4.2.9.2.1   Required flag

When enabled, the **Required** flag designates the field as required. If a field is designated as required, the user must enter data in that field to save the index of the document. AppEnhancer will not accept an empty required field. The following table describes the properties of the flag:

Item	Description
Available for these data types	Boolean Choice, Currency, Date (unless the Date Stamp flag is enabled), Decimal/Numeric, Integer, SSN, Telephone, Text, Time (unless the Time Stamp flag is enabled), User-defined List, or ZIP Code
Always enabled (cannot be disabled) for these data types	Date (when flagged with Date Stamp), Time (when flagged with Time Stamp), or Time Stamp
Enabled by default?	Yes

4.2.9.2.2   Search flag

When enabled, the **Search** flag designates the field as a search field. If a field is designated as a search field, then users can use this field to search for documents. The following table describes the properties of the flag:

Item	Description
Available for these data types	All (Boolean Choice, Currency, Date, Decimal/Numeric, Integer, SSN, Telephone, Text, Time, Time Stamp, User-defined List, or ZIP Code)
Enabled by default?	Yes

4.2.9.2.3   Read-Only flag

When enabled, the **Read-Only** flag designates the field as read-only. If a field is designated as read-only and not required, and the document has been saved with the field value as null, AppEnhancer enables the field to be edited only once. If a field is designated as read-only and required, then it cannot be modified after the index of the document is saved. AppEnhancer does not enable index modification of a read-only field. Index fields using the Time Stamp data type, the Time Stamp flag, or the Date Stamp flag are automatically added as read-only. The following table describes the properties of the flag:

Item	Description
Available for these data types	Boolean Choice, Currency, Date (unless flagged with Date Stamp), Decimal/Numeric, Integer, SSN, Telephone, Text, Time (unless flagged with Time Stamp), User-defined List, or ZIP Code
Always enabled (cannot be disabled) for these data types	Date (when flagged with Date Stamp), Time (when flagged with Time Stamp), or Time Stamp
Enabled by default?	No, except for index fields using the Time Stamp data type, the Time Stamp flag, or the Date Stamp flag

4.2.9.2.4   Doc Level Security flag

When enabled, the **Doc Level Security** flag enables the field to be used for Document Level Security. If this flag is enabled for a field, then AppEnhancer enables or denies user access to AppEnhancer documents, based on the contents of the field.

If you apply the **Doc Level Security** flag to a field, the **Document Level Security** tab appears so that you can configure Document Level Security. The following table describes the properties of the flag:

Item	Description
Available for these data types	All (Boolean Choice, Currency, Date, Decimal/Numeric, Integer, SSN, Telephone, Text, Time, Time Stamp, User-defined List, or ZIP Code)
Corresponding application creation tab	Document Level Security
Enabled by default?	No

> **Note:** If you intend to use the **Multiple indexes referencing single document** option for an application, it is recommended that you not apply the **Doc Level Security** flag to any index field in the same application. If you use the **Multiple indexes referencing single document** option, users that have access to a document through at least one index record will have access to that document

and all index records associated with it. For example, if Document Level Security is configured in an application so that documents with Yes as a value in the Protected index field are inaccessible to the Data Entry group, and if a particular document has one index record with Yes in the Protected field and another index record with No in the Protected field, the Data Entry group can view the document and both index records. To make a document inaccessible to the Data Entry group, the value in the Protected field must be Yes in all index records of that document.

For more information, see "Document level security" on page 64.

4.2.9.2.5   Part of Unique Key flag

When enabled, the **Part of Unique Key** flag prevents the same index information from being used for more than one document. This feature is usually used on multiple index fields, but could be used to ensure a single unique index field, such as a social security number. When this option is enabled, AppEnhancer checks for redundant data entered into the enabled index field(s). If another document already contains that combination of index information in the unique field(s), AppEnhancer does not accept the entry. The following table describes the properties of the flag:

Item	Description
Available for these data types	All (Boolean Choice, Currency, Date, Decimal/Numeric, Integer, SSN, Telephone, Text, Time, Time Stamp, User-defined List, or ZIP Code)
Enabled by default?	No

4.2.9.2.6   Dual Data Entry flag

When enabled, the **Dual Data Entry** flag functions as a validation measure to ensure that documents are indexed correctly. If it is enabled, the user must enter the data of index field twice. AppEnhancer accepts the information upon verification of the second entry.

Item	Description
Available for these data types	Currency, Date (unless the Date Stamp field flag is enabled), Decimal/Numeric, Integer, SSN, Telephone, Text, Time (unless the Time Stamp field flag is enabled), or ZIP Code
Not available for these data types	Boolean Choice, Time Stamp, or User-defined List
Cannot be used for a field in combination with these flags	Date Stamp or Time Stamp
Enabled by default?	No

4.2.9.2.7 Key Reference flag

When enabled, the **Key Reference** flag enables the field to be used for key reference file indexing. Index information can be imported into a Key Reference table, from which data fields are automatically populated during indexing, based on the value entered in the key index field. To use key reference file indexing, an application must have one Key Reference field and at least one Data Reference field defined.

If you apply the **Key Reference** and **Data Reference** flags to fields, the **Key Reference File** tab appears so that you can configure Key Reference Import. The following table describes the properties of the flag:

Item	Description
Available for these data types	Currency, Date (unless the Date Stamp flag is enabled), Decimal/Numeric, Integer, SSN, Telephone, Text, Time (unless the Time Stamp field flag is enabled), User-defined List, or ZIP Code
Not available for these data types	Boolean Choice or Time Stamp
Cannot be used for a field in combination with these flags	Date Stamp or Time Stamp
Corresponding application creation tab	Key Reference File Setup
Enabled by default?	No

4.2.9.2.8 Data Reference flag

When enabled, the **Data Reference** flag enables the field to be used with the Key Reference for reference file indexing. To use key reference file indexing, you must define one Key Reference field and at least one Data Reference field in an application.

If you apply the **Key Reference** and **Data Reference** flags to fields, the **Key Reference File** tab appears so that you can configure Key Reference Import. The following table describes the properties:

Item	Description
Available for these data types	Boolean Choice, Currency, Date (unless the Date Stamp flag is enabled), Decimal/Numeric, Integer, SSN, Telephone, Text, Time (unless the Time Stamp flag is enabled), User-defined List, or ZIP Code
Not available for this data type	Time Stamp
Cannot be used for a field in combination with these flags	Date Stamp or Time Stamp
Corresponding application creation tab	Key Reference File Setup
Enabled by default?	No

Chapter 5

Installation overview

Plan your system components, security settings, and other configuration before installation. Planning enables you to take advantage of some of the AppEnhancer features for deployment. For more information about components, see the *OpenText AppEnhancer Administration Guide*.

When you install AppEnhancer, all components are not mandatory. Install the required components based on your requirements. The following table outlines each of the AppEnhancer components and what is included in the setup:

Component	What gets installed (Note: some of these components are optional)
AppEnhancer Administrator	• AppEnhancer Demo Database • Component Registration Wizard • Administrator (on IIS)
AppEnhancer Web Access	• Web Access Server (on IIS) • Component Registration Wizard
AppEnhancer Rendering Server	• Rendering Server • Component Registration Wizard
AppEnhancer License Server	• AppEnhancer License Service
AppEnhancer Web Services	• AppEnhancer Web Services • Web Services client code samples • Web Services test console • Utility Web Services • Component Registration Wizard
Storage and Retention	• Auto Retention Filer Service • File Access Manager Service • Component Registration Wizard
AppEnhancer Integration Framework	• AppEnhancer Event Dispatch Broker • AppEnhancer Workflow Integration Module
AppEnhancer Image Capture	• AppEnhancer Image Capture
AppEnhancer REST Services	• AppEnhancer REST Services • Component Registration Wizard
OpenText Process Automation	• OpenText Process Automation

Chapter 6

Before you install

Before beginning installation, ensure that your system meets the requirements described in the *AppEnhancer Release Notes* for this release.

6.1 Prerequisites for AppEnhancer

Before you install AppEnhancer Administrator, ensure that the following requirements are met:

- For AppEnhancer installation, we recommend to use a Local Administrator (service) account.

- For AppEnhancer configuration, you must use the configured account (not necessarily a Local Administrator account) that has Log on as a service user rights.

  If the Windows security provider is in use, the following accounts also require Log on locally user rights: AppEnhancer Web Access, Web Services, and REST Services Server.

- Optionally, create an AEDEMO database. Follow the instructions in "Installing and configuring data sources" on page 107

- You might need to install and configure the appropriate database client software on each AppEnhancer server component computer. It is recommended that you ensure the IIS site for AppEnhancer Administrator has a secure binding (HTTPS) and can accept SSL connections.

  📄 **Note:** Ensure that you log in with administrator privileges to add or remove AppEnhancer components.

6.1.1 Installing and configuring data sources

AppEnhancer uses a database to store AppEnhancer application information, index values for AppEnhancer documents, and other important operating information. AppEnhancer supports Oracle, MySQL, PostgreSQL, and SQL databases. If your AppEnhancer database is an Oracle or MySQL database, you must install and configure the appropriate client software on each AppEnhancer component computer where you need to locate the AppEnhancer database.

Read-only databases are supported as AppEnhancer databases only for retrieval of documents.

📄 **Note:** If your AppEnhancer database is a Microsoft SQL Server or PostgreSQL database, no client configuration is required.

For more information about supported databases, see *AppEnhancer Release Notes*.

6.1.1.1   Installing and configuring Microsoft SQL Server for AppEnhancer

1.   Install Microsoft SQL Server.

     📄 **Note:** If you use a case-sensitive database as an AppEnhancer database, performance might improve, but you must use only AppEnhancer components to enter data into this database. If you use any other method to enter data into a case-sensitive database, data mismatch might occur.

2.   Configure Microsoft SQL Server for AppEnhancer. The Microsoft documentation set contains instructions on creating and configuring SQL Server databases.

     Ensure that you have enabled the SQL Server and Windows authentication mixed mode for the server authentication to enable users to connect to the AppEnhancer database.

3.   Add the Microsoft SQL Server data source in AppEnhancer Administrator. For more information, see the *AppEnhancer Administration Guide*.

6.1.1.2   Installing and configuring Oracle for AppEnhancer

AppEnhancer works with ODP.NET that is installed with Oracle ODAC. You must install ODAC before you configure an AppEnhancer data source that targets an Oracle database.

1.   Install Oracle Server. For more information, see the Oracle documentation.

2.   Install Oracle ODAC and configure the Oracle clients. For more information about how to install and configure Oracle clients, see the Oracle documentation set.

3.   Add the Oracle data source in AppEnhancer Administrator.

6.1.1.2.1   Connecting to Oracle Databases 18c and 19c

For upgraded environments where Oracle Data Access Components (ODAC) 12c clients are installed, connecting to Oracle Database 18c or Oracle Database 19c works without any additional steps. For environments where ODAC 12c clients are not installed, follow these steps to manually register Oracle .NET assemblies into the Global Assembly Cache (GAC):

1.   Open the Windows command prompt as an Administrator.

2.   At the command prompt, run the following commands:

     - ```
       <Oracle_Installation_Folder>\product\19.0.0 (or 18.0.0)\client_1
       \ODP.NET\bin\4\OraProvCfg.exe /action:gac /providerpath:<Oracle
       _Installation_Folder>\product\19.0.0 (or 18.0.0)\client_1\ODP.NET\
       bin \4\Oracle.DataAccess.dll
       ```

     - ```
       <Oracle_Installation_Folder>\product\19.0.0 (or 18.0.0)\client_1
       \ODP.NET\bin\4\OraProvCfg.exe /action:gac /providerpath:<Oracle
       ```

```
_Installation_Folder>\product\19.0.0 (or 18.0.0)\client_1\ODP.NET
    \PublisherPolicy\4\Policy.4.112.Oracle.DataAccess.dll
```

- `<Oracle_Installation_Folder>\product\19.0.0 (or 18.0.0)\client_1`
  `\ODP.NET\bin\4\OraProvCfg.exe /action:gac /providerpath:<Oracle`
  `_Installation_Folder>\product\19.0.0 (or 18.0.0)\client_1\ODP.NET`
  `\PublisherPolicy\4\Policy.4.121.Oracle.DataAccess.dll`

- `<Oracle_Installation_Folder>\product\19.0.0 (or 18.0.0)\client_1`
  `\ODP.NET\bin\4\OraProvCfg.exe /action:gac /providerpath:<Oracle`
  `_Installation_Folder>\product\19.0.0 (or 18.0.0)\client_1\ODP.NET`
  `\PublisherPolicy\4\Policy.4.122.Oracle.DataAccess.dll`

6.1.1.3 Installing and configuring MySQL for AppEnhancer

1. Install MySQL Server software and a MySQL administrator.

2. Configure MySQL Server.

   a. Start the MySQL administrator.
   b. Create a database for use with AppEnhancer.

3. On each computer where you need to access an AppEnhancer MySQL database, including the computer on which AppEnhancer Administrator has been installed, install the MySQL Connector/ODBC client driver.

4. On each computer where you need to access an AppEnhancer MySQL database, including the computer on which AppEnhancer Administrator has been installed, create an ODBC data source.

5. Add the MySQL data source in AppEnhancer Administrator.

📄 **Notes**

- The AppEnhancer data source name must be exactly same as the ODBC data source name that you have configured.

- If you configure MySQL as your Render Server database, you must configure an ODBC data source that points to the Render Server database that uses the name `RenderServer`.

6.1.1.4 Installing and configuring PostgreSQL for AppEnhancer

1. Install PostgreSQL Server software and a PostgreSQL administrator.

   a. The PostgreSQL Server software can be downloaded from the official PostgreSQL website.
   b. For the PostgreSQL administrator, you can use pgAdmin, which is an open source administration and development platform for PostgreSQL.

2. Configure PostgreSQL server.

   a. Start the PostgreSQL administrator.

      b.   Create a database for use with AppEnhancer. For more information about creating and configuring PostgreSQL databases, see the PostgreSQL documentation.

3.   Add the PostgreSQL data source in AppEnhancer Administrator. For more information, see *AppEnhancer Administration Guide*.

Chapter 7

Installation

This chapter provides instructions on how to install and maintain AppEnhancer components. Download the ZIP file from OpenText MySupport at http://www.opentext.com/support to a temporary directory on your computer. If you download multiple products, download all of them to the same temporary directory.

7.1 Installing AppEnhancer Administrator

You must install AppEnhancer Administrator to configure environment settings for AppEnhancer components, among other tasks. Installing the DEMODB is optional.

Ensure that IIS is installed by using the ASP.NET role service.

> 📄 **Notes**
>
> • You must apply the hotfix if you are using Windows Server 2008 x64 Edition. The KB980368 article on the Microsoft website contains the hotfix and the information.

1. Log in as a user with administrative rights on the computer where you want to install AppEnhancer Administrator.

2. Close all open applications.

3. Run the installer.

4. Click **Next**.

5. On the **License Agreement** page, select **I accept the terms in the license agreement**, and click **Next**.

6. On the **Customer Information** page, specify who is allowed to use the application on the computer:

   • To allow open access to the application, select **Anyone who uses this computer (all users)**.

   • To restrict access to the application, select **Only for me**. Only the user listed in the **User Name** field on the **Customer Information** page is allowed to use the application.

7. Click **Next**. The **Setup Type** page appears.

8. Select the setup type. Click **Next**.

   a. If you click **Custom** proceed to step 9.
   b. If you click **Complete** proceed to step 10.

---

9. Select the components that you want to install. Proceed to step 10 if all features (Administrator, Demo Database) are selected (by default). If Demo Database is not selected proceed to step 11.

10. If you choose to install the Demo Database, you are prompted to provide the database connection information.

    a. Enter a valid database server instance name in the Database Server field.

    b. Leave the default name AEDEMO of the database or change it to another name for creating the Demo database.

        The Demo Database is an optional tool for new AppEnhancer users and consists of the demonstration application index data, the demonstration application image files, and the AppEnhancerDEMO data source. The AppEnhancerDEMO data source refers to a configured SQL server database that enables you to perform various functions without affecting the actual system. The demonstration applications (CHECKS, COLDAPP, CONADMIN, COUNTY, DOCS, HR, IMAGEAPP, MEDICAL, _FORMS, and _RSTAMP) will be created in the demo database.

        • The demo database can be installed only on SQL Server or SQL Server Express.

        • If you need to install the demo database on SQL Server Express, the Server Name should be Server name\SQLExpress.

        • If you cannot find the database instance name by clicking the Browse button, enter the name manually in the Database Server field.

    c. Click **Next**.

11. On the **Select Web Application** page, the Internet Information Services (IIS) service displays a list of your available websites—for example, Default Website.

12. Select the target website from the Site List. This website will host the AppEnhancer Web Access application.

    > ⚠️ **Caution**
    > Do not install AppEnhancer Administrator on the Administration (root) website.

13. To change the default application name, type a name in the **Application name** field.

    > 📄 **Note:** If you do not change the application name, the setup wizard installs the Web Content files to the website that is currently selected in the Site List. Make a note of this location because it will be a part of the AppEnhancer Web Access website address.

14. **Require SSL** is selected by default. If SSL is not required, deselect this option. Ensure that SSL is configured if the option is enabled.

15. Choose the Web Server destination folder and click **Next**.

16. Click **Install**.

17. Click **Finish**.

18. After installation run Component Registration Wizard, select **Administrator** and type system credentials. Click **Test** and **Next** to register Administrator. For information about Component Registration, see"Registering AppEnhancer Administrator" on page 132.

19. Restart IIS.

7.2 Installing AppEnhancer Web Access Server

AppEnhancer Web Access can be installed on a single computer, or it can be deployed on several computers to take advantage of distributed processing by multiple computers.

The AppEnhancer Web Access Server setup installs the AppEnhancer Web Access website. If you have the AppEnhancer software retention license, the AppEnhancer Web Access Server also enables you to perform retention tasks for AppEnhancer software retention-enabled applications. The setup installs the IIS Rewrite Module if it is not present on the client.

Ensure that the following prerequisites are met before you register the AppEnhancer Web Access Server:

- At least one supported database that is network-accessible to the AppEnhancer Web Access Server, configured as a data source through AppEnhancer Administrator.

- A fully configured AppEnhancer Administrator installation.

- A computer with Rendering Server installed.

- A License Server that is network-accessible to the AppEnhancer Web Access Server and contains appropriate licenses for anticipated usage on your system.

- AppEnhancer applications and security settings must be configured through AppEnhancer Administrator in the database (or databases) for the data source group for which you want to provide AppEnhancer Web functionality.

- A storage location for AppEnhancer documents.

- If you are using dual write paths with your AppEnhancer deployment, verify that your write path configuration complies with the recommendations provided there for systems that include AppEnhancer Web Access.

- AppEnhancer Web Access is also supported with Secure Socket Layer (SSL). Before you use SSL with an AppEnhancer Web website, you must connect to the web server by using SSL to confirm that SSL is correctly configured.

- Ensure that IIS is installed by using the ASP.NET role service.

1. Close all open applications.

2. Log in as a user with administrative rights on the computer where you want to install AppEnhancer Web Access Server.

3. Run the installer.

4. Click **Next**.

5. On the **License Agreement** page, select **I accept the terms in the license agreement**, and click **Next**.

6. On the **Customer Information** page, specify who may use the application on the computer:

   • 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 field, select **Only for me**.

7. Click **Next**. The **Select Web Application** page appears. The Internet Information Services (IIS) service displays a list of your available websites—for example, Default Website.

8. Select the target website from the Site List. This website that will host the AppEnhancer Web Access application.

   > ⚠ **Caution**
   > Do not install AppEnhancer Web Access on the Administration (root) website.

9. To change the default application name, type a name in the **Application name** field.

   > 📄 **Note:** If you do not change the application name, the setup wizard installs the Web Content files to the website currently selected in the Site List. Make a note of this location because it will be a part of the AppEnhancer Web Access website address.

10. **Require SSL** is selected by default. If SSL is not required, deselect this option. Ensure that SSL is configured if the option is enabled.

11. Click **Next**. The **Destination Folder** page appears.

    The default Web Server Destination Folder is `C:\inetpub\wwwroot`

12. Click **Next**.

13. On the **Ready to Install** page, click **Install** and then click **Finish**.

14. Configure Web Access server settings in AppEnhancer Administrator.

15. Type the address for AppEnhancer Web Access in the browser. The format for the website URL for a AppEnhancer Web Access server is as follows:

- `https://<Server>/AppEnhancer`. If SSL is not enabled then the URL should be `http://<Server>/AppEnhancer`

- The placeholder <Server> represents the IP address or domain of the IIS server.

16. On the **Login** page, type your credentials and click **Login**.

17. Test the user ability to create or display documents.

7.3 Installing AppEnhancer Rendering Server

AppEnhancer Rendering Server renders all documents including images displayed in the viewer. You can choose to install AppEnhancer Rendering Server on the same workstation as AppEnhancer Web Access Server. When deploying an enterprise system, you can install multiple AppEnhancer Rendering Servers for a single data source group to improve rendering efficiency. After installing the AppEnhancer Rendering Server, you must register the server with the data source group by using the AppEnhancer Component Registration Wizard.

1. Close all open applications.

2. Log in as a user with administrative rights on the computer where you want to install AppEnhancer Rendering Server.

3. Run the installer.

4. Click **Next**.

5. On the **License Agreement** page, select **I accept the terms in the license agreement**, and click **Next**.

6. On the **Customer Information** page, specify who may use the application on the computer:

   - 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 field, select **Only for me**.

7. Click **Next**. The **Destination Folder** page appears.

   The default installation directory is `C:\Program Files\XtenderSolutions\ Content Management`

8. Click **Next**.

9. On the **Garbage Collection** page, select **Enable Garbage Collection on this Rendering Server** to ensure the Render Server removes cached files when necessary. If you deploy multiple Render Servers, ensure that only one Render Server has this option enabled.

10. Click **Next**.

11. Click **Install**.

12. Click **Finish**.

13. Configure Rendering Server settings in AppEnhancer Administrator.

14. Register Rendering Server by using Component Registration Wizard.

15. Verify that the AppEnhancer Rendering Server service is running.

16. Log in to AppEnhancer Web Access.

17. Test the rendition by creating or displaying a Word document.

18. Verify that JPG files can be created in the AppEnhancer Rendering Server cache directory.

7.4   Installing AppEnhancer License Server

The OpenText license module manages licensing for all AppEnhancer content management products. The License Server is used by products to validate licensing options and to monitor users on the system. You must have a valid license server installed on at least one workstation, with registered licenses for the components and features that your AppEnhancer system requires.

Consider the following for the location of the License Server:

• A location that is always accessible to each AppEnhancer content management product user.

• The database server is the recommended location for License Server installation, because users already must have access to the database server to use AppEnhancer content management products.

📄 **Note:** You must have a fingerprint ready via Fingerprint Generator prior to requesting a license. Any changes made to the License Server machine may invalidate the license and will require a new fingerprint.

Fingerprint Generator

1. Copy FingerprintGenerator.exe to a folder on the License Server machine.

2. Launch a CMD window.

3. Go to the folder in Step 1.

4. Run FingerprintGenerator.exe.

You will see the fingerprint displayed in the command window. This is what you need to request a new license. You can copy it to clip board or redirect it to a file using a command like `fingerprintgenerator.exe > .\fingerprint.txt`.

License Installation

1. Close all open applications.

2. Log in as a user with administrative rights on the computer where you want to install AppEnhancer License Server.

3. Run the installer.

4. Click **Next**.

5. On the **License Agreement** page, select **I accept the terms in the license agreement**, and click **Next**.

6. On the **Customer Information** page, specify who may use the application on the computer:

    *   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 field, select **Only for me**.

7. Click **Next**. The **Destination Folder** page appears.

    By default, it installs to:`C:\Program Files\XtenderSolutions\Content Management`

8. To change the default installation directory, click **Change**, navigate to the new directory, and then click **OK**.

9. Click **Next**.

10. On the **Ready to Install the Program** page, click **Install**.

11. Click **Finish**.

For more information about assigning and managing License Groups, see the *AppEnhancer Administration Guide*.

7.5 Installing AppEnhancer Storage and Retention

The AppEnhancer Storage and Retention setup program installs components which are necessary to support Centera storage and optional retention management licensed features.

Before running the AppEnhancer Storage and Retention Setup Wizard, ensure that you have configured the computer to be used as the AppEnhancer Auto Retention Filer Server. Configuration includes setting up credentials and other settings for the AppEnhancer Auto Retention Filer service and/or AppEnhancer File Access Manager service in AppEnhancer Administrator.

1. Close all open applications.

2. Log in as a user with administrative rights on the computer where you want to install AppEnhancer Storage and Retention.

3. Run the installer.

4. Click **Next**.

5. On the **License Agreement** page, select **I accept the terms in the license agreement**, and click **Next**.

6. On the **Customer Information** page, specify who may use the application on the computer:

   • 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 field, select **Only for me**.

7. Click **Next** to continue.

   On the **Select Features** page, specify which component you want to install on the computer.

   > **Note:** File Access Manager Service is not installed by default.

Component	Description
Auto Retention Filer Service	This service provides automatic retention capabilities for the optional retention management licensed features. You must have a valid license to use retention. **Note:** If the Automatic Retention Filer Service will be used heavily, it is recommended that you set up a dedicated user account for this service and installing it on a separate server from the File Access Manager Service to improve performance.
File Access Manager Service	This service, which provides Centera storage access and manual retention capabilities, is required for Centera storage device and Centera retention for AppEnhancer licensed feature. You can install the File Access Manager service on one dedicated computer where Auto Retention Filer service is installed. You must have a valid license to use retention.

8. To change the default installation directory, click **Browse**, navigate to the new directory, and then click **OK**.

9. Click **Next**.

10. Click **Install**.

11. Click **Finish**.

    > **Note:** It is recommended that you reboot the server when the installation is finished.

12. Create service credentials for the installed component in AppEnhancer Administrator.

13. Run Component Registration Wizard, and register the components. "Registering other AppEnhancer components" on page 132 provides information on Component Registration.

7.6 Installing AppEnhancer Web Services

Web Services enable the integration of applications. AppEnhancer Web Services setup installs AppEnhancer Web Services.

Before running the AppEnhancer Web Services setup, ensure that the following prerequisites are met:

- A License Server that is network-accessible to the AppEnhancer Web Services Server and contains appropriate licenses for anticipated usage on your system.

- A fully configured AppEnhancer Administrator installation.

- The operating system must be Windows Server. If you perform the operating system installation, ensure that the following requirements are implemented during installation:

    - Windows Indexing Service (optional) must be configured: to enable client-side full-text searches of your website.

    - Windows Script Debugger (optional) must be configured: to assist in troubleshooting and debugging.

- The account used to install the AppEnhancer Web Services must be a member of the local Administrators group on the computer where the install is performed. The local Administrators group should have the following Advanced Rights: Log on as a service, and Act as part of the operating system

- If the required operating system has already been installed without IIS and ASP.NET, you must add them. If you install IIS before you install the Microsoft .NET Framework, ASP.NET is automatically installed. However, if you install IIS after you install the Microsoft .NET Framework, you must install ASP.NET.

    - Microsoft Internet Information Server (IIS) must be installed and configured on the AppEnhancer Web Services computer. Select IIS as your web server during setup and install it before you install the Microsoft .NET Framework.

- Microsoft .NET Framework must be installed.

- If any of the system resources are on computers remote to the AppEnhancer Web Services computer, an appropriate impersonation account configured as Domain user for access to those resources.

- Ensure that IIS is installed using the ASP.NET role service.

- AppEnhancer Web Services is also supported with Secure Socket Layer (SSL). Before you use SSL with an AppEnhancer Web website, you must connect to the web server using SSL to confirm that SSL is correctly configured.

1. Close all open applications.

2. Log in as a user with administrative rights on the computer where you want to install AppEnhancer Web Services.

3. Run the installer.

4. Click **Next**.

5. On the **License Agreement** page, select **I accept the terms in the license agreement**, and click **Next**.

6. On the **Customer Information** page, specify who may use the application on the computer:

   - 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 field, select **Only for me**.

7. Click **Next**. The **Setup Type** page appears.

   The **Setup Type** page displays all of the available components in a tree and enables you to install only specific features.

8. To install only **AppEnhancer Web Services** on this computer, click on each of the other components that you want to exclude and select**This feature will not be installed**.

   a. If you have IIS installed and running select **Internet Information Services (IIS) Deployment**:

      i. Click **Next**.
      ii. A list of the available sites for the installation of the application appears. Select the site where you want to install it.
      iii. Accept the default or type a different name for the application in the **Application name** field.
      iv. Click **Next**.
      v. On the **Ready to Install the Program** page, click **Install**.
      vi. Click **Finish**.

    b. If you do not have IIS installed, select **AppEnhancer Web Services Host Deployment**:

        i. Click **Next**.

        ii. Accept the default or type a different name for the application in the **Application name** field.

        iii. Click **Next**. The **Destination Folder** page appears.

            The default directory is installed `C:\Program Files\XtenderSolutions \Content Management`. Click **Browse** to specify a different directory.

        iv. Click **Install**.

        v. Click **Finish**.

9. To install only *Utility Web Services* on this computer, click on the icon for each of the other components that you want to exclude and select **this feature will not be installed** from the pop-up menu.

    a. If you have IIS installed and running you can select **Internet Information Services (IIS) Deployment**:

        i. Click **Next**.

        ii. On the **Ready to Install the Program** page, click **Install**.

        iii. Click **Finish**.

    b. If you do not have IIS installed, select **AppEnhancer Web Services Host Deployment**:

        i. Click **Next**.

        ii. Accept the default or type a different name for the application in the **Application name** field. The default name is XsWebServices

        iii. Click **Next**. The **Destination Folder** page appears.

            The default directory is installed `C:\Program Files\XtenderSolutions \Content Management`. Click **Browse** to specify a different directory.

        iv. Click **Next**.

        v. Click **Install**.

        vi. Click **Finish**.

10. Configure the session cache path in AppEnhancer Administrator.

11. Create service credentials for the service in AppEnhancer Administrator.

12. Run Component Registration Wizard, and register the components. "Registering other AppEnhancer components" on page 132 provides information on Component Registration.

7.6.1   Specifying Web Service settings

The **Web Service Configuration** page enables you to select settings specific to AppEnhancer Web Services.

- Take note of information as described in the following table:

Property	Description
Host Type	Take note of the host type that the AppEnhancer component resides on. The Host Type setting on this page reflects the deployment choice made during AppEnhancer component installation.
Port	The Port setting reflects the selections made during AppEnhancer component installation as follows: • If IIS deployment was selected, this setting automatically reflects the port number associated with the selected website. • If AppEnhancer Web Host deployment was selected, this setting automatically reflects the port number associated with the AppEnhancer component. If you want to change the website later, you can run AppEnhancer Component Registration Wizard again to change the port number.
Site URL	Take note of the site URL (Universal Resource Locator) for the website where the AppEnhancer component is installed. The site URL is automatically provided, depending on the IP address entered on the **Component Information** page of the AppEnhancer Component Registration Wizard.

Property	Description
Virtual Root	Take note of the virtual root that will be used for the AppEnhancer component website. You can use this setting to include a company name or create a hierarchy. The method to modify this setting depends on the deployment selection as follows: • If IIS deployment was selected, the virtual root can be specified only in the Windows utility, Internet Information Services. • If AppEnhancer Web Host deployment was selected, you can modify the virtual root in the AppEnhancer Component Registration Wizard.
Physical Path	Take note of the physical path to the AppEnhancer component files. (The Physical Path text box is available only for AppEnhancer Web Host.)

7.7 Installing AppEnhancer REST Services

AppEnhancer REST Services are RESTful services that interact with the AppEnhancer platform.

Ensure that IIS is installed using the ASP.NET role service.

1. Close all open applications.

2. Log in as a user with administrative rights on the computer where you want to install REST Services.

3. Run the installer.

4. Click **Next**.

5. On the **License Agreement** page, select **I accept the terms in the license agreement**, and click **Next**.

6. On the **Customer Information** page, specify who may use the application on the computer:

   • 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 field, select **Only for me**.

7. **Require SSL** is selected by default. If SSL is not required, deselect this option. Ensure that SSL is configured if the option is enabled.

8. On the **Select Web Application** page, select the website that you would like to install.

9. Accept the default or type a different name for the application in the **Application name** field. The default name is AppEnhancerReST.

10. Click **Next**. The **Destination Folder** page appears. The default folder is `C:\inetpub\wwwroot`

11. Click **Install**.

12. Click **Finish**.

> **Note:** Restart the system if prompted.

AppEnhancer REST Services is also supported with Secure Socket Layer (SSL). Before you use SSL with an AppEnhancer Web website, you must connect to the web server using SSL to confirm that SSL is correctly configured.

7.8 Installing AppEnhancer Administrative Services

AppEnhancer Administrative Services include the Archive Service and Migration Service. You may install all three services or choose to install only the services that you need. Administrative Services can also be installed in a multi-server environment.

1. Close all open applications.

2. Log in as a user with administrative rights on the computer where you want to install Administrative Services.

3. Run the `ApplicationEnhancer Administrative Services.msi` installer file.

4. On the **License Agreement** page, select **I accept the terms in the license agreement**, and click **Next**.

5. On the **Customer Information** page, enter the user name and organization, and click **Next**.

6. On the **Setup Type** page, select the installation type. Select **Complete** for a full installation that contains all of the subcomponents of Administrative Services or **Custom** for a customized installation.

    Click **Next**.

7. On the **Ready to Install the Program** page, click **Install**.

    The Installation Completed page appears when the setup wizard has successfully installed Administrative Services.

8. Click **Finish**.

> **Note:** Restart the system if prompted.

9. In AppEnhancer Administrator, configure the service credentials for the Administrative Services before proceeding to step 10. For more information, see the *AppEnhancer Administration Guide*.

10. Run the Component Registration Wizard, and register the components. "Registering other AppEnhancer components" on page 132 provides information about component registration.

7.9 Installing AppEnhancer Import Utility

AppEnhancer Import Utility allows you to add documents more efficiently by importing multiple documents at once based on your import specifications.

Prerequisites

Before you can set up AppEnhancer Import Utility, you must make sure you have the following components ready:

- A minimum of Microsoft .NET Runtime version 7.0 because it is required to run the application. If you do not have this software framework installed, you will be prompted to install it when running the installer.

- Administrative rights for the user performing the installation.

Installation procedure

To install AppEnhancer Import Utility:

1. Download and save the AppEnhancer Import Utility 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 .NET Runtime installed, you will be prompted to install the required software before you can continue.

   When you are finished, click **Next**.

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**.

When you are finished, click **Next**.

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

    When you are finished, click **Next**.

7. On the **Ready to Install the Program** page, click **Install**.

8. Click **Finish**.

9. Before you can use AppEnhancer Import Utility, you must update the `AEImportUtility.dll.config` file with your AppEnhancer Administrator URL and Datasource name. In the file, update the **AdminURL** value with your AppEnhancer Administrator URL and the **DefaultDS** value with your Datasource name.

    The configuration file is stored in the folder where AppEnhancer Import Utility is installed. By default, it is the `C:\Program Files\XtenderSolutions\AppEnhancerImportUtility` folder.

    Save your configuration file.

7.10   Installing AppEnhancer Integration Framework

7.10.1   AppEnhancer Integration Framework components

- Microsoft .NET Framework 4.8 (WCF Service is required to install EDB)
- Ensure that IIS is installed by using the ASP.NET role service

7.10.2   Installing AppEnhancer Integration Framework components

The AppEnhancer Integration Framework Components connect other products with AppEnhancer products. The setup package contains the installation files for EDB and WIM.

EDB and WIM can be installed on the same computer or installed separately. You can adopt installation strategy based on your requirements.

7.10.2.1 Installing the EDB and WIM on the same machine

Install the components on the same machine if the machine has enough processing power to handle the anticipated volume of transactions.

1. Navigate to the setup package and run the `setup.exe` file.

2. Install the prerequisite software if prompted. If the wizard displays a message indicating that a reboot is necessary, click **Yes** to reboot your machine. After you reboot, click **Install** again to resume the setup. The installation wizard displays the Welcome screen after all required components have been located or installed.

3. Click **Next**.

4. On the **License Agreement** page, read and accept the license agreement and click **Next**.

5. On the **Customer Information** page, provide customer information and specify who may use the application.

6. Click **Next**.

7. On the **Setup Type** page, select **Complete** and click **Next**.

8. On the **Select Web Application** page, select **Default Web Site** from the **Site List** and click **Next**.

9. On the **Destination Folder** page, click **Next**.

10. On the **WIM Configuration** page, specify user account credentials for the WIM service. The WIM uses the account to communicate with the EDB.

    > **Note:** The user account must be associated with the **Log on as a service** policy in the WIM machine security settings. Refer to the operating system help for information on how to assign user rights.

11. Click **Next**.

12. On the **Ready to Install the Program** page, click **Install**.

13. Click **Finish**.

7.10.2.2   Installing the EDB and WIM on Separate Machines

Install the components on separate computers if one computer does not have enough processing power to handle the anticipated volume of transactions.

7.10.2.2.1   Installing EDB

1. Navigate to the setup package and run the `setup.exe` file.

2. Install the prerequisite software if prompted. If the wizard displays a message indicating that a reboot is necessary, click **Yes** to reboot your computer. After you reboot, click **Install** again to resume the setup. The installation wizard displays the Welcome screen after all required components have been located or installed.

3. Click **Next**.

4. On the **License Agreement** page, read and accept the license agreement and click **Next**.

5. On the **Customer Information** page, provide customer information and specify who may use the application.

6. Click **Next**.

7. On the **Setup Type** page, select **Complete** and click **Next**.

8. On the **Custom Setup** page, specify the following installation options:

   - Select **AppEnhancer Event Dispatch Broker**.

   - Clear **Workflow Integration Module**.

9. Click **Next**.

10. On the **Select Web Application** page, select **Default Web Site** from the **Site List** and click **Next**.

11. On the **Destination Folder** page, click **Next**.

12. On the **Ready to Install the Program** page, click **Install**.

13. Click **Finish**.

7.10.2.2.2   Installing WIM

1. Navigate to the setup package and run the `setup.exe` file.

2. Install the prerequisite software if prompted. If the wizard displays a message indicating that a reboot is necessary, click **Yes** to reboot your computer. After you reboot, click **Install** again to resume the setup. The installation wizard displays the Welcome screen after all required components have been located or installed.

3. Click **Next**.

4. On the **License Agreement** page, read and accept the license agreement and click **Next**.

5. On the **Customer Information** page, provide customer information and specify who may use the application.

6. Click **Next**.

7. On the **Setup Type** page, select **Complete** and click **Next**.

8. On the **Custom Setup** page, specify the following installation options:

   - Clear **AppEnhancer Event Dispatch Broker**.

   - Select **Workflow Integration Module**.

9. Click **Next**.

10. On the **WIM Configuration** page, specify the EDB server address in **Event Dispatch Broker URL**.

11. Specify the user account credentials for the WIM service. The WIM uses the account to communicate with the EDB.

    > **Note:** The user account must be associated with the **Log on as a service** policy in the WIM computer security settings. Refer to the operating system help for information about how to assign user rights.

12. Click **Next**.

13. On the **Ready to Install the Program** page, click **Install**.

14. Click **Finish**.

7.10.2.3   Verifying the installation

1. Verify that **AppEnhancer Event Dispatch Broker** is available in Windows Programs and Features.

2. Verify that **Event Dispatch Broker Database Configuration** appears under **All Programs > OpenText**.

3. Verify that EDB site appears in the IIS Manager under **Web Sites** > **Default Web Site**.

4. If you installed the WIM component, verify that the **AppEnhancer WIM** service appears in the Windows Services console. The AppEnhancer WIM service starts automatically, by default.

7.10.2.4   Configuring the event profile database

When you install EDB for the first time, you must configure the event profile database for the EDB and Event Profile Manager.

Note: Only an administrator can locally configure the event profile database.

1. From Windows Start menu, go to **OpenText AppEnhancer > Event Dispatch Broker Database Configuration** to open the EDB database configuration page.

2. Set the configurations for the following values:

   - **EDB database**: You can create a new database or select an existing EDB database.

   - **Administrator**: Configure an Administrator password which will be used to login to the Event Profile Manager page.

   - **Email**: Provide an email address to which error message notifications are sent.

3. Click **Save**.

Chapter 8

Post-installation configurations

Run the AppEnhancer Component Registration Wizard to register components after installation.

8.1 Setting up a new AppEnhancer system

1. On Administration server, start AppEnhancer Component Registration Wizard.

2. Click **Next**.

3. On the **AppEnhancer System** page, select **Create a new AppEnhancer system**.

4. On the **Administrator Credentials** page, specify the administrator credentials of the AppEnhancer system.

5. On the **Service Credentials** page, specify the service credentials of the AppEnhancer administrator and click **Next**.

   The account (impersonate user) used must be a member of the local Administrators group on the computer. The local Administrators group should have the following Advanced Rights: Log on as a service.

   AppEnhancer Component Registration Wizard runs a diagnostic utility to detect whether the account used for resource authentication credentials has the advanced rights applied. If the diagnostic utility detects any discrepancies in permissions for AppEnhancer accounts, the Review Permissions dialog box appears.

   1. In the Review Permissions dialog box, select **Save** to save the list of issues to an XML file, or select **OK** to close the dialog box.

   2. If you need to modify rights of the local administrators group, exit the AppEnhancer Component Registration Wizard before doing so. Apply any missing privileges to the appropriate accounts. (You may need to restart the computer after applying permissions.) Then restart the AppEnhancer Component Registration Wizard and repeat previous steps.

6. Click **Finish**.

7. Login to AppEnhancer Administrator and choose Global Administration.

8. Create the data source group in AppEnhancer Administrator.

   The *AppEnhancer Administration Guide* provides instructions on configuring database client software on the AppEnhancer component servers.

8.2   Registering AppEnhancer Administrator

1. Start AppEnhancer Component Registration Wizard.

2. Click **Next**. If the Global Configuration database has been located on the computer, proceed to step 4.

3. Select **Connect to an existing AppEnhancer system** and locate the Global Configuration Database. For more information, see "Locating the Global Configuration Database for AppEnhancer components" on page 134.

4. In the Component type list, select **Administrator**.

5. On the **Service Credentials** page, specify the service credentials of the AppEnhancer administrator.

    The account (impersonate user) used to install must be a member of the local Administrators group on the computer where the install is performed. The local Administrators group should have the following Advanced Rights: Log on as a service and Act as part of the operating system.

    AppEnhancer Component Registration Wizard runs a diagnostic utility to detect whether the account used for resource authentication credentials has the advanced rights applied. If the diagnostic utility detects any discrepancies in permissions for AppEnhancer accounts, the Review Permissions dialog box appears.

    a. In the Review Permissions dialog box select **Save** to save the list of issues to an XML file, or select **OK** to close the dialog box.

    b. If you need to modify rights of the local administrators group, exit the AppEnhancer Component Registration Wizard before doing so. Apply any missing privileges to the appropriate accounts. (You may need to restart the computer after applying permissions.) Then restart the AppEnhancer Component Registration Wizard and repeat previous steps.

6. Click **Finish** and restart AppEnhancer Administrator site in IIS.

8.3   Registering other AppEnhancer components

Register each component using the Component Registration Wizard after installation:

- Auto Retention Filer

- Rendering Server

- REST Services

- Web Access Server

- Web Services

- Administrative Services (You must register each subcomponent separately):

- Archive Service

- Migration Service

- Index Image Import Service

- Indexing Service

- AutoIndex KeyRef Service

> **Note:** Run the AppEnhancer Component Registration Wizard and register the components one at a time.

1. Ensure that Administrator is registered before you register other components.

2. Start AppEnhancer Component Registration Wizard.

3. On the **Welcome** page, click **Next**.

   If the Global Configuration database has been located on the computer, proceed to step 5.

4. If prompted, select **Connect to an existing AppEnhancer system** and locate the Global Configuration Database. For more information, see "Locating the Global Configuration Database for AppEnhancer components" on page 134.

5. On the **Select the operation you would like to start** page, select **Register**, and click **Next**.

   > **Note:** You can also use the AppEnhancer Component Registration Wizard to unregister components. To get started, select **Unregister**.

6. Select the component that you want to register, and click **Next**.

7. Type a description in the Description text box. This description appears in the registered components listing in AppEnhancer Administrator.

8. Confirm the correct IP address appears in the IP Address text box or enter the correct one.

   - If you are registering AppEnhancer Web Services, the **Web Services Configuration** page appears. "Specifying Web Service settings" on page 122 provides the details to configure web service settings.

   - If you want to register another component enable **Run Component Registration wizard again?**

9. Click **Finish**.

You must run the AppEnhancer Component Registration Wizard again in the following circumstances:

- Deletion of the default data source.

- Selection of a different data source as default.

- Modification of the user name or password in the data link properties for a data source.

- Modification of resource authentication credentials in AppEnhancer Administrator.

- Application of a software update.

8.3.1   Locating the Global Configuration Database for AppEnhancer components

When registering an AppEnhancer component, the AppEnhancer Component Registration Wizard prompts you to locate the Global Configuration Database if it has not been located on this computer.

> **Caution**
>
> The Global Configuration Database should be kept online unless you decide to retire the whole data source group.

1. For a SQL Server, PostgreSQL, or Oracle database, if a schema has been set up in the database, enter the name in the **Schema** box. You must specify the data source schema to use application user credentials when connecting to a SQL Server, PostgreSQL or Oracle database. You must avoid using any spaces, keywords, or beginning the name with a numeral (an underscore can be used in place of a space.)

2. Click **Data Link**. The Data Link Properties dialog box appears. In this dialog box, enter location information for a database within the data source group where you want to register the AppEnhancer component. The instructions for locating a data source within this dialog box vary depending on the database software.

    For details, see the following sections:

    - "Locating a Microsoft SQL Server data source" on page 135
    - "Locating an Oracle data source" on page 135
    - "Locating a MySQL data source" on page 136

    The *AppEnhancer Administration Guide* provides instructions on installing and configuring database client software on the AppEnhancer component servers.

3. After you have entered data source location information in Data Link Properties dialog box, click **OK**.

    a. If a user account has not been saved with the database connection information (in the Data Link Properties dialog box), the Login dialog box appears.

    b. In the User Name text box, type an administrative user name. In the Password text box, type the password for the user name you entered.

c. Click **Login**.

4. Click **Next**.

8.3.1.1 Locating a Microsoft SQL Server data source

The **Provider** tab of the Data Link Properties dialog box enables you to locate a SQL Server data source.

1. Select Microsoft OLE DB Provider for SQL Server and click **Next**. The **Connection** tab appears.

2. From the Server Name list, select the server on which you placed your SQL Server database. In the User Name and Password text boxes, type the user name and password. (These are the database login values that you created in SQL Server for the AppEnhancer database.) Select the AppEnhancer database name from the Select Database list.

> **Note:** If you do not check **Allow saving password**, you must use a database schema.

3. You can test the connection between the AppEnhancer component and the database by clicking **Test Connection**.

4. Click **OK** and proceed with the selecting the Component Type for registration.

8.3.1.2 Locating an Oracle data source

The **Provider** tab of the Data Link Properties dialog box enables you to locate an Oracle data source. AppEnhancer components have been tested with ODP.NET provider that is installed with Oracle ODAC component. You must install ODAC before you configure an Oracle data source.

1. Select Oracle Provider for OLE DB.

2. Click **Next**. The **Connection** tab appears.

> **Note:** If you do not check **Allow saving password**, you must use a database schema.

3. In the Data Source text box, enter the data source name you configured in the tnsnames.ora file. In the User Name and Password text boxes, type the user name and password. (These are the database login values that you created in Oracle for the AppEnhancer database.)

4. You can test the connection between the AppEnhancer component and the database by clicking **Test Connection**. Click **OK**.

5. Click **OK** and proceed with the selecting the Component Type for registration.

8.3.1.3   Locating a MySQL data source

The **Provider** tab of the Data Link Properties dialog box allows you to locate a MySQL data source.

1.   Select Microsoft OLE DB Provider for ODBC Drivers, and click Next. The **Connection** tab appears.

2.   From the Use data source name list, select the MySQL data source that you created in ODBC Administrator.

3.   In the User Name and Password text boxes, type the user name and password. (These are the database login values that you created in MySQL for the AppEnhancer database.)

4.   Select **Allow saving password.**

5.   You can test the connection between the AppEnhancer component and the database by clicking **Test Connection**. Click **OK**.

6.   Click **OK** and proceed with the selecting the Component Type for registration.

8.4   Updating the database schema name and credentials

To update the database schema name and credentials:

1.   Enter the command line option `/updateDB` or `-updateDBstart` to launch the Component Registration Wizard in Update Database Connection mode.

2.   Select the database connections, update the schema name and credentials, and save the changes to the configuration database.

3.   In the Component Registration Wizard, update the local `XSCM.config` file for each AppEnhancer server:

   •   If the original connection is no longer valid, the Component Registration Wizard will prompt you to enter new database connection information and update the `XSCM.config` file.

   •   If the original connection is still valid but not up to date, click **Update Database Connection** to update the configuration database connection.

   📄 **Note:** You do not need to rerun the Component Registration Wizard after making data source changes and render server changes. However, it is recommended to restart all services and web applications after making changes, especially if there are changes to the `XSCM.config` file.

   Run the AppEnhancer Data Source Selector on each workstation (with legacy AppEnhancer desktop applications and services) again. Click **Locate** or **Refresh** to update the DataSourceConfigMgr section in the local `XSCM.config` file.

For more information about advanced Component Registration Wizard options, refer to Appendix A, Advanced Component Registration Wizard options on page 147.

Chapter 9

Installing and configuring add-ins

9.1 AppEnhancer Office 365 add-in

9.1.1 Prerequisites

Before you install the AppEnhancer Office 365 add-in, ensure that the following requirements are met.

9.1.1.1 Server requirements

- If you plan to run your add-in in Office Online, it must be SSL-secured. Self-signed certificate is not recommended, as it will block features of the Outlook Online add-in.

- To run an Outlook add-in, the user's Outlook email account must reside on a Microsoft Exchange Server 2013 or later, which is available through Office 365, Exchange Online, or through an on-premises installation.

- To run an Outlook Online add-in, AppEnhancer Web Access needs to have a public IP address or public DNS and must be accessed over HTTPS.

- To run an Outlook Online add-in, AppEnhancer Web Access needs to have internet access. For those servers using a web proxy to access the internet, you need to specify the proxy configuration in the Web.config file for Web Access. By default, the Web.config file can be found in the following directory: `C:\inetpub\wwwroot\AppEnhancer\Web.config`.

  The following is an example of the proxy configuration:

  ```
  <configuration>
    <configSections>
    ...
    </configSections>
    <appSettings>
    ...
    </appSettings>
    <system.net>
      <defaultProxy>
        <proxy
          usesystemdefault="true"
          proxyaddress="http://192.168.1.10.3128"
          bypassonlocal="true"
        />
        <bypasslist>
          <add address="192\.168\.\d{1,3}\.\d{1,3} />
        </bypasslist>
      </defaultProxy>
    </system.net>
  </configuration>
  ```

  For more information about specify the proxy configuration, see docs.microsoft.com/en-us/dotnet/framework/configure-apps/file-schema/network/defaultproxy-element-network-settings (https://docs.microsoft.com/en-

us/dotnet/framework/configure-apps/file-schema/network/defaultproxy-element-network-settings).

9.1.1.2   Client requirements

- Microsoft Windows 10 or later.

- Office Online or Office 2016 for Windows or later (32- or 64-bit).

- Microsoft Edge must be installed, but does not have to be the default browser.

> **Note:** Multiple Office desktop applications cannot run multiple AppEnhancer Web Access add-ins at same time. If multiple add-ons are running, you will receive an error message, "An unexpected error occurred. Details: Invalid Request!"
>
> It is not recommended to run multiple Web Access add-ins at the same time. However, if you must do so, you can suppress this error message by turning off XSRF attack prevention in the Web Access Web.config file as follows:

```
<!--If prevent XSRF attack for Web Access server-->
<add key="ValidateXSRFToken" value="false"/"
```

9.1.2   Configuring the AppEnhancer Office 365 add-in

1. Navigate to `C:\inetpub\wwwroot\AppEnhancer\Office365Manifest\`.

2. Open each of the following files in a text editor:

   - AEWordAddIn.xml

   - AEExcelAddIn.xml

   - AEPowerPointAddIn.xml

   - AEOutlookAddIn.xml

3. In each file, find all instances of the string "*https://localhost/AppEnhancer*" and replace it with the URL of your AppEnhancer server address.

4. Save the files.

> **Note:** Users who wish to install the AppEnhancer Office 365 add-in can download the manifest files from:
>
> - https://*<server-address>*/AppEnhancer/Office365Manifest/**AEExcelAddIn.xml**
>
> - https://*<server-address>*/AppEnhancer/Office365Manifest/**AEPowerPointAddIn.xml**
>
> - https://*<server-address>*/AppEnhancer/Office365Manifest/**AEWordAddIn.xml**
>
> - https://*<server-address>*/AppEnhancer/Office365Manifest/**AEOutlookAddIn.xml**

9.1.3 Configuring AppEnhancer Web Access settings

9.1.3.1 Disabling X-Frame-Options

AppEnhancer Web Access uses X-Frame-Options to prevent clickjacking attacks by ensuring that Web Access content is not embedded into other sites. To use Web Access as an add-in with Microsoft Office, you must disable this setting.

1. Navigate to `C:\inetpub\wwwroot\AppEnhancer\Web.config`.

2. In the Web.config file, find and comment out the following line:

```
<add name="X-Frame-Options" value="sameorigin" />
```

See example below:

```
<httpProtocol>
  <customHeaders>
    <!-- <add name="X-Frame-Options" value="sameorigin" />
    <remove name="X-Powered-By"/>
  </customHeaders>
</httpProtocol>
```

3. Restart Internet Information Services (IIS).

9.1.3.2 Configuring AppEnhancer to use SameSite cookies

To ensure that AppEnhancer works as expected with SameSite cookies, complete the following steps:

1. Navigate to docs.microsoft.com/en-us/aspnet/samesite/kbs-samesite (https://docs.microsoft.com/en-us/aspnet/samesite/kbs-samesite). Install .Net Framework 4.8 and the KB for SameSite according to the operating system on the server that is running Web Access.

2. Navigate to `C:\inetpub\wwwroot\AppEnhancer\Web.config` and edit the Web.config file as follows:

```
    <!-- To allow cross-site cookie use, for instance, CDK or Office add-in which
make use of iframe, cookieSameSite="None" must be used in system.web/sessionState-->
    <sessionState cookieName="ASP.NET_SessionId_Wx" cookieSameSite="None" />
    <!-- To allow cross-site cookie use, for instance, CDK or Office add-in which
make use of iframe, sameSite="None" requireSSL="true" must be used in system.web/
httpCookies-->
    <httpCookies httpOnlyCookies="true" sameSite="None" requireSSL="true" />
```

3. To enable support for older browsers, edit the Web.config file as follows:

```
    <!--If true, we will detect the browser to strip the sameSite=None attribute
from cookies if a browser is known to not support it-->
    <add key="SameSiteSupportForOldBrowsers" value="true"/>
```

4. Configure the IIS to ensure that the AppEnhancer site is using HTTPS.

> 📄 **Note:** Users cannot log in to Web Access over HTTP protocol with the settings above. You must use HTTPS to log in to Web Access.

Chapter 10

Upgrading AppEnhancer

10.1 Planning an AppEnhancer upgrade

To upgrade, you must first uninstall all previously installed AppEnhancer components. After upgrading the AppEnhancer database, do not connect to the database with an older version of AppEnhancer.

10.1.1 Connectivity between releases

When you upgrade your AppEnhancer database, it is recommended that you upgrade all workstations at the same time, and before upgrading the database. If you must delay upgrading some workstations, do not use any new features until all workstations have been upgraded. Also, do not use any administrative components from previous releases. Server components from any previous release of AppEnhancer Content Management, such as AppEnhancer Web Access, and Index Server, must be uninstalled before you upgrade the AppEnhancer system.

> **Caution**
> Uninstall any previous release of AppEnhancer Web Access Server before you upgrade.

10.1.2 Supported platform upgrade considerations

If you are upgrading from a previous release of AppEnhancer Web Access, ensure the operating systems on your servers are still supported. With each release of AppEnhancer Web Access, the list of operating systems supported for the AppEnhancer computer may change, for a variety of reasons. If operating system support does change, and if you plan to use the same hardware for your AppEnhancer Web Access server, you may need to upgrade the operating system before you install the current version of AppEnhancer.

For more information about supported operating systems, see the *AppEnhancer Release Notes*.

> **Caution**
> If you upgrade from one type of deployment to another, you should completely uninstall all AppEnhancer Web components and redeploy the system.

10.1.3   Upgrading the current version of AppEnhancer

Plan your upgrade carefully and back up all databases before you upgrade AppEnhancer.

1.   Process all batches and auto-index data.

2.   Back up and restrict access to databases. Use the utilities included within the database programs to perform the backup. Refer to the documentation for your database program. If you are upgrading an AppEnhancer system that has more than one user, restrict access to the database. This is necessary to prevent any other users on the AppEnhancer system from interfering with the upgrade process.

3.   Upgrade to a supported database, if any of your databases are no longer supported by AppEnhancer. For a list of supported databases, see *AppEnhancer Release Notes.* For instructions about upgrading your database to a version that is supported by AppEnhancer, refer to the documentation for your database.

> **Note:** For more information about third-party components that you need to upgrade, see the *AppEnhancer Release Notes*.

4.   Uninstall the previous release of AppEnhancer components.

If you attempt to install the current release of AppEnhancer when a previous release is still installed, the Product Conflict dialog box appears and lists all the previous products. Click **Remove All** to remove the products and continue to install.

> **Note:** If some files are still available on the client machine after the uninstallation process, delete them manually.

5.   Install the latest version of AppEnhancer.

6.   Upgrade data sources from the previous version using AppEnhancer Administrator. For steps to upgrade data sources in AppEnhancer Administrator, see "Upgrading data sources" on page 145.

7.   Switch to a different security if you want to do so. It is recommended that you do so at this point in the upgrade process. For more information, see the *AppEnhancer Administration Guide*.

8.   Upgrade each remaining AppEnhancer system workstation.

9.   Reconfigure registered email accounts from the previous version in **Web Access User Settings**after the upgrade.

10.   The xPlore full-text engine is no longer supported, and documents will need to be re-indexed.

10.1.3.1 Upgrading data sources

Upgrade data sources from a previous version by using AppEnhancer Administrator.

> 📄 **Note:** Before you upgrade the data source, ensure the PID table of that data source is cleaned up.

1. In AppEnhancer Administrator, go to Environment > Data Sources.

    Data sources created in a previous version of AppEnhancer do not have the Valid check mark associated against them.

2. Select the data source that you want to upgrade and click **Upgrade**.

3. Click **Yes** to confirm and continue with the upgrade.

    A message appears that indicates that the data source is upgraded.

10.1.3.2 Security providers and upgrading

You can switch to a different security provider after you upgrade all of your AppEnhancer databases to the current version of AppEnhancer.

10.1.3.3 Selecting a different security provider

After you have upgraded your AppEnhancer databases, you can switch to a different security provider. For more information about security providers, see the *AppEnhancer Administration Guide*.

> ⚠️ **Caution**
>
> All workstations on the AppEnhancer system must be upgraded before you switch security providers. You must then recreate all groups and their privileges, users and their privileges, Document Level Security settings, and annotation group settings. Use caution when switching security providers, especially in a production environment.

1. In AppEnhancer Administrator, go to **Environment > Data Sources**.

2. Select the data source that you want to modify.

3. Select the **Security Model**.

4. From the drop-down list, select the security provider that you want to use.

5. Click **Apply**.

6. Open AppEnhancer Administrator with this data source set as default. A message appears indicating that the security provider for the current data source has been changed.

    > 📄 **Note:** Use caution when switching security providers, especially in a production environment.

7.   If you are certain that you want to change the security provider in use, click **Yes**.

8.   Re-create all groups and their privileges, users and their privileges, Document Level Security settings, and annotation group settings.

10.1.3.4   One AppEnhancer Administrator login account for all data sources

When you log in to AppEnhancer Administrator, you can log in to all of the data sources or a selected data source in AppEnhancer Administrator. To log in to all of the data sources in AppEnhancer Administrator by using either the CM security provider or the Windows security provider, ensure that the following criteria is met:

- The administrator user account must exist on all data sources.

- The administrator user account must have the same password on all data sources.

- The administrator user account must have the Administrator user privilege on all data sources. (This criterion does not apply to SYSOP.)

To log in to a selected data source, ensure that the user account that you use to log in to AppEnhancer Administrator has sufficient privileges.

For this reason, ensure there is at least one user account that satisfies these criteria before you begin the upgrade process. If such a user account is not in place before you upgrade, do not set up data sources in AppEnhancer Administrator after the upgrade.

Appendix A. Advanced Component Registration Wizard options

The following options can be used in the Component Registration Wizard. For more information, refer to .

- `-NewConfig` or `/NewConfig`: Displays a file selection dialog to enter a new config file path

- `-updateDB` or `/updateDBstart`: Starts the Component Registration Wizard in Update Database Connection mode

  > **Note:** The above two options can be used together, but cannot be combined with the headless command line mode.

- `ComponentSetup.exe NewSystem` *<Password for admin user SYSOP> <User> <UserPassword>*: Starts the headless command line mode to create a new system.

- `ComponentSetup.exe ConnectDB` *<mode> <DB connection string> <DB schema>*(optional): Starts the headless command line mode. If *<mode>* is "Admin", a user name and password for Admin are also required.

- `ComponentSetup.exe Unregister` *<mode>*: Starts the headless command line mode to unregister components.

- `ComponentSetup.exe` *<mode> <AEAdminUserName> <AEAdminPassword>*: Starts the headless command line mode.

  > **Note:** *<mode>* can be one of the following strings:
  >
  > - `Admin`
  > - `WebServer`
  > - `Rendering`
  > - `AutoRetention`
  > - `RestService`
  > - `CenteraServer`
  > - `XSServices`
  > - `AEServices`
  > - `Migration`
  > - `Archive`
  > - `III`