
LinkOne WebView Technical Reference

Version 3.22

© Copyright 2018 ABB
All Rights Reserved
Confidential and Proprietary
Published: Wednesday, 12 September 2018

Legal Disclaimer

The product described in this documentation may be connected to, and/or communicate information and data via, a network interface, which should be connected to a secure network. It is your sole responsibility to ensure a secure connection to the network and to establish and maintain appropriate measures (such as but not limited to the installation of firewalls, application of authentication measures, encryption of data, installation of antivirus programs, etc.) to protect the product, the network, your systems, and the interface against any kind of security breach, unauthorised access, interference, intrusion, leakage, damage, or corruption or theft of data. We are not liable for damages or losses related to any such security breach, unauthorised access, interference, intrusion, leakage, damage, or corruption or theft of data.

Contents

System Requirements	7
Where to Get Help	9
Documentation	10
Overview	11
Installation	12
Installing Using Installer Application.....	12
Setup Reference.....	12
Installing Using Copy	13
Internet Information Services (IIS) 6.....	13
Internet Information Services (IIS) 7, 8.....	15
Installing on a Single Server	15
Configuring Side-By-Side	16
Installing on a Web Farm	16
Configuring Files.....	17
Configuring Databases.....	18
Configuring State	18
Operation	19
Web Site	19
Service Layers.....	19
Folder Structure.....	20
Page Themes	20
Page Format	20
Security	21
Authentication	21
Authorization.....	21
Profiles	21
Roles	22
Users.....	22
Policies	22
Site Licensing.....	23
Content Caching.....	23
Selection Lists.....	24
Supplied Selection Lists	25
OCI Selection List	25
Selection List Changes Since WebView 2.x.....	26
Selection List Reference	26
Book Libraries	33
Content Encoding.....	33
Error Logging.....	33
Event Logging.....	33
Backing Up.....	34
Tools	36
Clear Cache Tool	36
Library Re-Indexer	36

LinkOne Book Indexing Service	37
Migrating User Notes From WebView 2.2.....	38
Library Synchronizer	40
Samples	43
Silent Login.....	43
Pricing Selection List.....	44
Walkthroughs	45
How To Authenticate Using Active Directory.....	45
How To Determine Chosen Microsoft .Net Version	45
How To Authenticate Using Windows Authentication	46
Creating a Web Site Theme.....	46
How To Add New Translations	47
How To Change the Book or Selection List Menus	47
How To Change the Mapped Selection List Fields.....	48
How To Create a Custom Selection List	48
How To Create a Custom Security Provider.....	48
How To Create a Custom Policy	50
How To Share Notes with LinkOne WinView	51
How To Use SQL Server Instead of SQL Express.....	51
How to Install Mass Library Replication.....	52
Cygwin, Rsync and OpenSSH Installation	52
OpenSSH Server Configuration.....	53
SSH Public Key Authentication	53
Rsync Client Configuration	54
How To Install on x64 Windows	54
How To Configure the Web Site for HTTPS.....	54
How To Configure IIS 7	55
How To Configure for Multiple Web Sites	55
How To Configure DBMan Configuration Manager	55
How To Use DBMan Configuration Manager to Open a Book Via Equipment Id	56
How To Configure Book Libraries on a Network Share	56
How to Configure SharePoint Integration.....	57
Requirements	58
Set Up Process	58
Sharepoint Alerts (Notifications).....	61
How to Configure Advanced Search Facility	61
Administration	61
Operation.....	66
Overriding Publisher defined Parts List Formatting	67

System Requirements

Requirements for LinkOne WebView Servers:

- One of the following operating systems:
 - Microsoft Windows 2008 R2 Server
 - Microsoft Windows 2012 Server
 - Microsoft Windows 2016 Server
 - The following are supported for non-production use only.
 - Microsoft Windows 10
 - Microsoft Windows 7
 - Microsoft Windows 8
- Microsoft .NET Framework 3.5 SP1 (for the installer)
- Microsoft .NET Framework 4.0 (for WebView)
- ASP .NET 4
- Microsoft IIS 6, 7 or 8
- Microsoft SQL Express 2008 or later
- Microsoft SQL Server 2008, 2012, 2014, 2016.
- 2GB of RAM
- 100 MB of available hard disk space (plus space for any LinkOne books and notes)

Minimum Requirements for LinkOne WebView Clients:

The following versions of the supported browser have been actively tested during the development of LinkOne WebView 3.22. Other versions of the supported browsers may operate but any issues may be rectified at ABB's discretion.

Operating System	Supported Browser	Version
Microsoft Windows 7, 8, 8.1, 10	Internet Explorer	11, Edge
Microsoft Windows 7, 8, 8.1, 10	Google Chrome	Current version at time of release testing of LinkOne WebView 3.22
Microsoft Windows 7, 8, 8.1 10	Firefox	Current version at time of release testing of LinkOne WebView 3.22
Apple OSx	Safari	5.0.5
Apple iOS	Safari	Current version at time of release testing of LinkOne WebView 3.22
Google Android	Chrome	Current version at time of release testing of LinkOne WebView 3.22

NOTE: 3D Model viewing is currently only supported on Internet Explorer 11 and Firefox.

Minimum Requirements for Advanced Search:

- Advanced Search Facility is supported by using a SOLR Server
- SOLR index Hard Disk space varies but as a guide, space equivalent to the size of the LinkOne libraries is required.

Where to Get Help

The *Enterprise Software Online Customer Community*

(<https://enterprisesoftware.force.com/customerportal/login>) (OCC) provides help for all users of LinkOne products. You are encouraged to actively participate in helping other users, sharing knowledge and ideas on how you use LinkOne and how you think it can be improved.

If you have an issue with a LinkOne product, you should first search the OCC to see if any other users have encountered this problem to check if there is a solution available. If there is no solution already available, post to the appropriate OCC group and supply the following in your post:

- A short description of the problem, including any steps required to reproduce the problem.
- Details of the book you were viewing. If necessary upload a small test book illustrating the problem.
- Any configuration settings you think may have contributed to the problem.
- Any customizations you may be using with LinkOne.
- Your current operating system and service packs installed.
- Any other factors you think are relevant to the issue you are experiencing.

The LinkOne team will also be monitoring and responding to posts and we will try our best to solve any issues before the next release.

Documentation

LinkOne WebView provides the following documents to help you get started.

LinkOne WebView Release Notes

The Release Notes contain an overview of what changes are included in the 3.22 release.

LinkOne WebView Getting Started Guide

The Getting Started Guide is the starting point for all users of LinkOne WebView. The guide provides a quick overview of installation and configuration and new features available in Version 3.

LinkOne WebView User Online Help

This is the online help system for end users, which can be accessed at any time from Help > Help on the browser menu. This provides information on features in the LinkOne WebView application.

LinkOne WebView Technical Reference

The Technical Reference is designed for developers and system administrators who will be deploying and integrating LinkOne WebView into their corporate environments.

Overview

In this document, you will find information on the technical operations of LinkOne WebView, targeted at those people tasked with administering, extending, or integrating the web site.

Readers should have a solid knowledge of the workings of Asp.net before continuing.

Further technical material can be found in the API documentation, available by browsing to [~/Documentation/LinkOne WebView API.chm](#).

If you need further guidance, contact your LinkOne reseller or **Enterprise Software Global Customer Care** (if you purchased LinkOne directly from Ventyx) via the **Customer Portal** (<https://enterprisesoftware.force.com/customerportal/login>) or via **email** (<mailto:CustomerCare.ES@abb.com>). You can get help in answering your questions as well as provide feedback and suggestions on the **Enterprise Software Online Customer Community** (<https://enterprisesoftware.force.com/customerportal/login>) (OCC).

Installation

LinkOne WebView can be installed in a number of ways, and then configured for a number of environments. The simplest installation is for a single web server, and running the installer to completion with all default settings. The other ways and configurations are detailed in this section.

Installing Using Installer Application

LinkOne WebView is published as an installer application which contains a complete copy in several languages of the application web site, documentation, and sample books.

You would choose to install this way if you are running a single web server environment, or you are looking to get a single server working correctly before deploying to other web servers in a web farm.

For the steps to install and the server requirements, please read the LinkOne WebView Getting Started Guide.

Setup Reference

A full list of install options is available from the command line. Use:

```
>WebView30.exe /?
```

If a log of the install is needed, then specify the log file on the command line. For example:

```
>WebView30.exe /l*v c:\install.log
```

The LinkOne WebView installer application uses properties that can be set on the command line prior to initiating the installation. Properties are passed to the installer using name-value pair sets on the command line. For example:

```
>WebView30.exe TARGETDIR="c:\installdir"
```

The published Properties are:

Property	Function
TARGETDIR	The physical directory where the web site files will be copied to.
WEBSITENAME	The name of the web site under which the virtual directory will be created.
VIRTUALDIRECTORYNAME	The name to use for creating the virtual directory.
APPLICATIONPOOLNAME	The name to use for creating the application pool.
VERSIONIIS	The installed Microsoft IIS version. Set this property to a value of 6 or higher to bypass checking if Microsoft IIS is installed.
VERSIONSQLSERVER	Set this property to a value of 9 or higher to bypass checking if Microsoft SQL Server is installed.
VERSIONASPNET	The installed version of Microsoft Asp.Net. Set this property to a value of 3.5 or higher to bypass checking if Microsoft Asp.Net is installed.

CREATESTARTMENUSHORTCUTS	Set to "yes" to create start menu shortcuts.
PRINTERNAME	The name of the print device to install for server side printing (only required if no print device is installed on the computer).
REMOVEUSERDATA	This is an uninstall property which can be used during silent uninstall to specify if user data should be removed during the uninstall. By default this property is set to 1 which will remove user data. Setting it to an empty string "" will ensure that user data is not removed during the uninstall.

Installing Using Copy

The entire LinkOne WebView web site is contained within a single folder, and is designed to be copied to other folders and recreated as a new web site.

You would choose to install this way if you wanted to quickly start a physically separate new instance of an existing web site. This allows you to try out new configurations, new libraries, etc, in complete isolation to the original web site.

You would also choose to install this way if you were rolling out a copy of the same web site to other web servers in a web farm. You would configure the first server to use network shared configuration and data, then copy that server's web site folder to the other servers.

The steps required to install this way are as follow:

Internet Information Services (IIS) 6 Configuring a new WebView web application

1. On the server you are copying from, open the services management console by clicking "**Start / Administrative Tools / Services**".
2. Right click on the "**SQL Server**" service and click "**Stop**".
3. Right click on the "**World Wide Web Publishing Service**" service and click "**Stop**".
4. Copy the WebView installation folder from its' current location (usually under "**C:\inetpub\wwwroot\yourdirectoryname**") to the new location/server under "**C:\inetpub\wwwroot**".
5. On the server you are **copying from**, open the services management console by clicking "**Start / Administrative Tools / Services**".
6. Right click on the "**SQL Server**" service and click "**Start**".
7. Right click on the "**World Wide Web Publishing Service**" service and click "**Start**".
8. On the server you are **copying to**, open Internet Information Services (IIS) Manager by clicking "**Start / Administrative Tools / Internet Information Services (IIS) Manager**".
9. Expand the tree until you locate "**Application Pools**".
10. Right click on "**Application Pools**" and click "**New... / Application Pool...**".
11. Enter a name for the application pool and press "**OK**".
12. Expand the tree until you locate the new installation folder.
13. Right click on the new installation folder and select "**Properties**".
14. Under the "**Directory**" tab, in the Application settings section, click "**Create**" to create a new application.

15. Set the "Execute permissions" to be "Scripts and Executables".
16. Set the "Application pool" to the application pool created earlier.
17. Click "Configuration".
18. Click "Insert..." in the Wildcard application maps section to create a new wildcard mapping.
19. Un-check the box "Verify that file exists".
20. Click "Browse..."
21. Locate the file "C:\windows\microsoft.net\Framework\v2.0.50727\aspnet_isapi.dll" and click "Open".
22. Click "OK".
23. Click "OK".
24. Under the "Documents" tab, click "Add".
25. Type "Default.aspx" and press "Enter".
26. Under the "HTTP Headers" tab, tick "Enable content expiration".
27. Select "Expire after" and enter 7 "Days".
28. Click "Add..."
29. Enter the "Custom header name" to be "X-UA-Compatible" and the "Custom header value" to be "IE=EmulateIE8".
30. Click "OK".
31. Under the "ASP.NET" tab, set the "ASP.NET version" to be "2.0.50727".
32. Click "OK".

Configuring compression on the new application

1. Open Internet Information Services (IIS) Manager by clicking "Start / Administrative Tools / Internet Information Services (IIS) Manager".
2. Right click on the second element from the top in the left hand pane "yourcomputername (local computer)" and click "Properties".
3. Ensure "Enable Direct Metabase Edit" is ticked.
4. Click "OK".
5. Right click on the second element from the top in the left hand pane "yourcomputername (local computer)" and click "All Tasks / Save Configuration to Disk".
6. Open the folder "C:\Windows\System32\inetsrv".
7. Make a backup copy of "Metabase.xml".
8. Open "Metabase.xml" in notepad.
9. Search for "<IISCompressionScheme" you will find 3 occurrences. The following 2 steps apply to the sections immediately below the two with a "Location" setting of "/LM/W3SVC/Filters/Compression/Deflate" and "/LM/W3SVC/Filters/Compression/gzip" (usually the first two).
 - a. Set the value of "HcDynamicCompressionLevel" to 9.
 - b. Add the following file extensions to the "HcScriptFileExtensions" setting, ensuring the spacing between each existing file extension matches the spacing between each new file extension:
 - "aspx axd js css svc"
10. Search for "<IIsWebDirectory" where the end of the "Location" setting matches the name of the new application you created earlier in IIS.
11. Add the setting "DoDynamicCompression="TRUE"" below the "HttpExpires" setting.
12. Save and close the "Metabase.xml" file.
13. Open Internet Information Services (IIS) Manager by clicking "Start / Administrative Tools / Internet Information Services (IIS) Manager".

14. Right click on the second element from the top in the left hand pane "**yourcomputername (local computer)**" and click "**All Tasks / Restart IIS...**"
15. Click "**OK**".

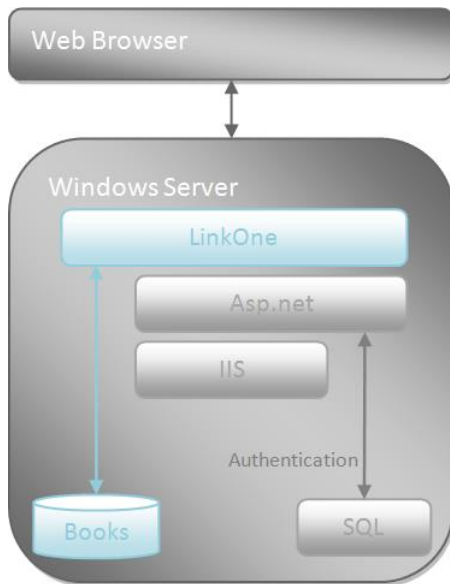
Internet Information Services (IIS) 7, 8 Configuring a new WebView web application

1. On the server you are copying from, open the services management console by clicking "**Start / Administrative Tools / Services**".
2. Right click on the "**SQL Server**" service and click "**Stop**".
3. Right click on the "**World Wide Web Publishing Service**" service and click "**Stop**".
4. Copy the WebView installation folder from its' current location (usually under "**C:\inetpub\wwwroot\yourdirectoryname**") to the new location/server under "**C:\inetpub\wwwroot**".
5. On the server you are **copying from**, open the services management console by clicking "**Start / Administrative Tools / Services**".
6. Right click on the "**SQL Server**" service and click "**Start**".
7. Right click on the "**World Wide Web Publishing Service**" service and click "**Start**".
8. On the server you are **copying to**, open Internet Information Services (IIS) Manager by clicking "**Start / Administrative Tools / Internet Information Services (IIS) Manager**".
9. Expand the tree until you locate "**Application Pools**".
10. Right click on "**Application Pools**" and click "**Add Application Pool...**".
11. Enter a name for the application pool.
12. Select "**.NET Framework v2.0.50727**".
13. Click "**OK**".
14. Under "**Application Pools**", right click on the newly created application pool and click "**Advanced Settings...**".
15. If the windows operating system is **64 bit** you must change "**Enable 32-Bit Applications**" to "**True**".
16. Change the "**Identity**" to "**Network Service**".
17. Click "**OK**".
18. Expand the tree until you locate the new installation folder.
19. Right click on the new installation folder and click "**Convert to Application**".
20. Click "**Select...**" and select the application pool created earlier then click "**OK**".
21. Click "**OK**".
22. You may need to add full permissions to the files under the **App_Data** folder of the installation path to the "**Network Service**" user.

Installing on a Single Server

Installation on a single web server can be done using either the installer application or copy methods. In the single server scenario, the web site configuration, book libraries, session state, and membership and reporting databases are generally kept within the web site folder. This is the default if the installer application is used.

Here is the structure of LinkOne WebView once it is installed:



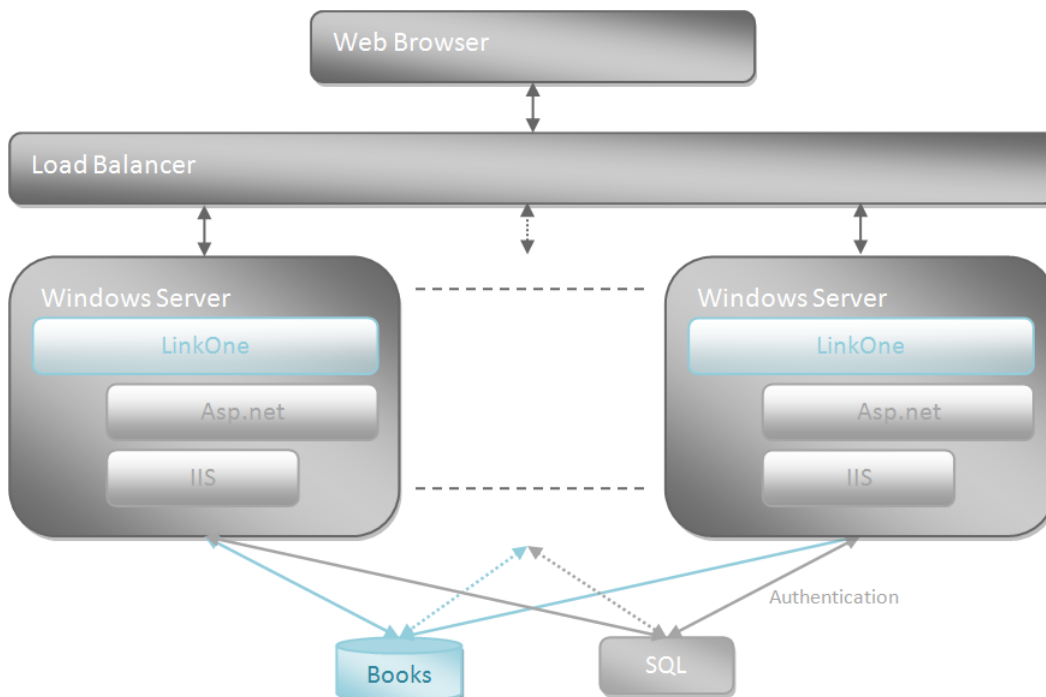
You would possibly reconfigure the web site to use book libraries that are shared with LinkOne WinView, in which case those book libraries would possibly be located on a network share. Browse to `~/Configuration/ConfigureLibraries.aspx` to choose a new library folder.

Configuring Side-By-Side

Installation for side-by-side running of the same or different versions of LinkOne WebView is simply a matter of running the associated Setups and choosing different names for each web site.

Installing on a Web Farm

A simple server configuration for a LinkOne WebView web farm will look like this (excluding firewall, switches, and failover equipment).



In this configuration:

- each WebView server requires an exact duplicate of the web site folder on each web server
- the file server has the function of hosting:
 - the LinkOne WebView configuration file through a file share
 - the LinkOne book library through a file share
 - the LinkOne book notes through a file share
 - the web site membership and profile SQL databases
 - the web site session state server

Before duplicating the web site to each web server, configure all of the files, folders, and databases as detailed in the following sections.

Configuring Files

LinkOne WebView relies on a few files and folders for configuration and data, and these are required to be shared amongst the web servers on the web farm. These files and folders must be placed on a network share for all of the web servers to read or write. These are detailed below.

Configuration File

All configuration settings for LinkOne WebView are kept in a file named **~/Mincom.LinkOne.WebView.config**. By default this file is in the web site root folder, but must be placed on a file share for all web servers to read and write configuration.

This file location is set with an appsetting in the **~/web.config** file.

For example:

```
<appSettings>
  <add
    key="Mincom.LinkOne.WebView.WebServer.Services.ServiceBase.ApplicationDataFile
    ePath" value="\\Server\LinkOne\Mincom.LinkOne.WebView.config" />
</appSettings>
```

The value for this appsetting is the absolute file name of the application configuration file.

Note: The configuration for Asp.net still resides in the **~/web.config** files found in each web site on each web server.

Data Log Folder

When an unexpected application error is caught by LinkOne WebView, a uniquely named error log file is created and written to the **~/Data/Log** folder. This folder must be moved to a file share for all web servers to read and write log files.

For example:

```
\\Server\LinkOne\Log
```

This folder is set by an administrator by browsing to **~/Configuration/ConfigureReporting.aspx**.

Library Folders

The books that LinkOne WebView shows in the Browse Books web page come from the configured library folders which by default consists of one library folder which is **~/Data/Libraries/Mincom**. This library must be moved to a file share for all web servers to read.

For example:

```
\\Server\LinkOne\Libraries\Mincom
```

These folders are set by an administrator by browsing to **~/Configuration/ConfigureLibraries.aspx**.

Notes Folders

By default the web site contains a `~/Data/PublicNotes` folder and a `~/Data/PrivateNotes` folder where public and private notes respectively, are stored. These folders must be moved to a file share for all web servers to read and write notes.

While public notes for all users go in the same folder, private notes must be specific to the logged on user, and not shared with other users. To facilitate this, the private notes folder path can contain a `{0}` place holder which will be replaced with the name of the logged on user to become the final user's private notes folder.

For example, a private notes folder path of:

```
\\Server\LinkOne\Users\{0}\LinkOne\PrivateNotes
```

Means for user John, the actual folder will resolve to:

```
\\Server\LinkOne\Users\John\LinkOne\PrivateNotes
```

These folders are set by an administrator by browsing to `~/Configuration/ConfigureNotes.aspx`.

Note: When deleting a user through the WebView configuration user interface, all the users' private notes will also be deleted. If the private notes need to be kept, then a backup of the private notes should be taken before deleting the user.

Configuring Databases

A single SQL database is used for the functions of membership, roles, and profiles. This database is found at `~/App_Data/ASPNETDB.MDF` and must be moved to the file share server and attached to a SQL Server instance on that server. The connection string to the database is found in `~/web.config` on each web server and must be changed to point to the shared database.

A second database is used for reporting and is found at `~/App_Data/LINKONEDB.MDF`. This database must also be moved to the file share server and attached to a SQL Server instance on that server. The connection string to the database is also found in `~/web.config` on each web server and must be changed to point to the shared database.

Configuring State

The session state is configured to use the in-memory state server by default. There are two options to share the state amongst web servers: either using a common Asp.net state server, or a SQL server. Please see <http://msdn.microsoft.com/en-us/library/ms178586.aspx> for details on configuring each.

Running a shared Asp.net state server on the file share server will be sufficient in most cases. Once running, update the `~/web.config` `sessionState` entry on each web server to point to the new Asp.net state server.

The view state being received and generated for each browser request needs to be shareable amongst each web server as well. This requires a common machine key on each web server which is found in `~/web.config` `machineKey` entry. Please see <http://msdn.microsoft.com/en-us/library/ms998288.aspx> for more details. A machine key generator can be found at <http://www.aspnetresources.com/tools/keycreator.aspx>.

Operation

The LinkOne WebView web site is built on Asp.net technology and leverages many of its features. These include:

- Theme folders to theme the web site pages
- Localization folders to localize the web site pages
- web.config files in each folder to secure that folder
- Site map for menu generation
- Security using the membership, role and profile database

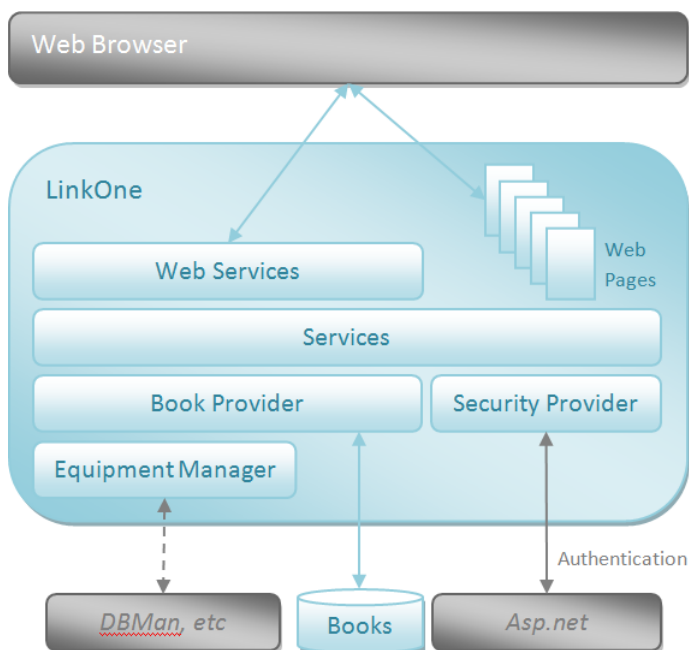
This section details operations of these features.

Web Site Service Layers

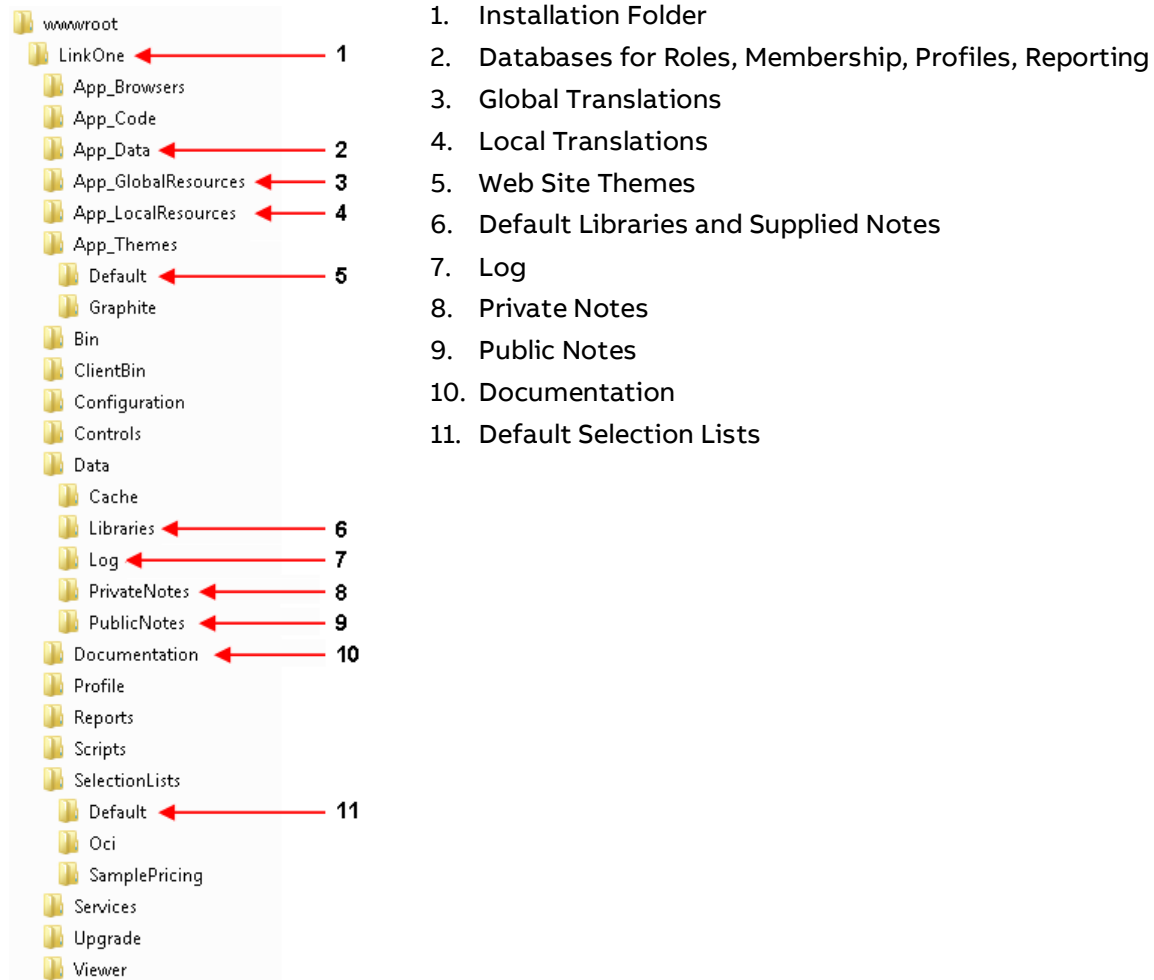
LinkOne WebView consists of several runtime areas that collectively generate content for a connected web browser. A high level diagram of these areas is shown below.

- Web Pages - these are the pages directly requested and shown by the web browser
- Web Services - these are the web services that are called by code running on the page in the web browser
- Services - these are the services that provide content for the web pages when they are requested, and for the web services when they are called
- Book Provider - this provider reads the LinkOne book content and supplies the LinkOne Document Object Model objects to the services
- Security Provider - this provider wraps the Asp.net security and is replaceable with a custom security provider
- Equipment Manager - this is an optional component allowing LinkOne WebView to interact with third party systems to resolve equipment identifiers to LinkOne parts

Further documentation on the services and providers can be found in the API documentation, available by browsing to [~/Documentation/LinkOne WebView API.chm](#).



Folder Structure



The LinkOne WebView web site is installed the physical structure shown above, which is the structure of a standard Asp.net 2.0 web site.

The name of the install folder defaults to the same name as the chosen virtual directory.

The location of the install folder defaults to a sub-folder of the physical folder of the chosen web site.

The LinkOne WebView application pool process identity is given read/write permissions to the entire web site folder.

Page Themes

LinkOne WebView uses Asp.net themes to style each web page. The default theme is identified in the `~/web.config` file and is overridden in the session by the theme chosen at login time. New themes can be added as a sub-folder to the web site themes folder, `~/App_Themes`, and they will automatically appear in the list of available themes on the login web page.

Page Format

LinkOne WebView uses master pages to provide a common look and feel across all web pages. They also provide basic page layout and error notification controls.

The web site contains two master pages, broadly separating book viewing functions, and site management functions.

Master Page	Derived Pages
~/web.master	~/* ~/Configuration/* ~/Reports/*
~/Viewer/web.master	~/Viewer/* (popup pages only)

Security Authentication

The web site by default is secured using Forms Authentication, and will redirect the user to ~/Login.aspx when a new request or a session-timed-out request is received.

Alternative methods of authentication are detailed by Microsoft at <http://msdn.microsoft.com/en-us/library/532aee0e.aspx>.

Authorization

The main web site folder is unsecured, and allows access to the documentation folder and login pages. Other sub-folders in the web site are secured and require the user to be authorized.

The following is a table of commonly accessed sub-folders and the authorization required for access.

Web Site Folder	Authorization Required
~/	None
~/Configuration	Administrators
~/Documentation	None
~/Profile	Administrators
~/Reports	Administrators
~/Viewer	Authenticated Users

Profiles

The web site by default uses the Asp.net SQL Profile Provider and stores profile details in the web site database at ~/App_Data/ASPNETDB.MDF.

A profile is maintained for each user and role. A user profile is stored against the user name, and a role profile is stored against the role name prefixed by "Role\$".

A profile consists of configuration and data specific to that user or role. For example policies, licenses, allowed books, allowed notes, and saved selection lists.

A user profile inherits some role profile details. For example policies, licenses, allowed books, and allowed notes.

Roles

The web site by default uses the Asp.net SQL Role Provider and stores role details in the web site database at `~/App_Data/ASPNETDB.MDF`.

The default installation includes two registered roles:

Role Name
Administrators
Users

Roles can be managed by browsing to `~/Configuration/ConfigureRoles.aspx`.

Users

The web site by default uses the Asp.net SQL Membership Provider and stores user details in the web site database at `~/App_Data/ASPNETDB.mdf`.

The default installation includes two registered users:

User Name	Password	Roles
Admin	admin	Administrators
User	user	Users

Users can be managed by browsing to `~/Configuration/ConfigureUsers.aspx`.

User password strength is controlled by a setting in `~/web.config`.

Policies

Policies are flags that are used to control behavior in the web site and are set on a role or user. For example, there are individual policies allowing the creation of supplied, public, and private notes. If a policy is set on a role, then all users in that role also inherit that policy. A policy can't be unset for a user if the policy is set on a role the user is in.

Policy flags are stored on the profile of the associated role or user, and are indicated by having a name prefixed with Policy.

The default installation includes the following policies:

Policy	Affects	Assigned To
Allow Category Writing	The ability to create or modify note categories.	Administrators
Allow Public Note Writing	The ability to create or modify notes for public viewing.	Administrators
Allow Supplied Note Writing	The ability to create or modify notes supplied with book content.	Administrators
Allow Private Note Writing	The ability to create or modify notes for private viewing.	Users
Allow Multilingual Note Writing	The ability to create or modify notes in languages other than the login language.	Administrators

Allow Selection List Sharing	The ability to see Selection Lists from other users that are in at least one of the roles the Assigned User is in.	Administrators
Disable changing Password	When set, stops the assigned user or users in the assigned role from changing the password on the 'My Profile' dialog. (Default: not set)	Administrators

Site Licensing

LinkOne WebView requires all users to possess a token to access the web site. If a user accesses the web site without a token then they will be granted one automatically. If all tokens licensed to the web site are taken, then the user will be denied access and redirected to the unlicensed web page which is **~/Unlicensed.aspx**.

Once a token has been granted to a user, it remains unavailable to other users for a period of 15 minutes from last use. So as long as the user stays active on the web site, they will continue to hold that token. If the user has had no activity for more than 15 minutes, then their token is released for use by another user.

It is the responsibility of the web site owner to authenticate each user prior to them gaining access to the web site and consuming a token. LinkOne WebView includes a sample login page to accomplish this authentication, which uses Asp.net Windows Forms Authentication.

By default, each LinkOne WebView installation comes with one free login token, allowing one login. When a valid LinkOne WebView site license is added to the configuration (**~/Conguration/ConfigureSiteLicenses.aspx**) then the number of tokens identified by the license will be available. The total number of tokens available is the sum of all tokens from valid site licenses.

A site license is valid if it has not expired, and it is serving the web site domain that it is licensed for.

Content Caching

LinkOne WebView takes advantage of the fact that most book content is static, and thus keeps a cached copy of pictures, embedded documents, and additional book graphics, outside the library in the **~/Data/Cache** folder. This cache folder can be shared amongst other web servers in a web farm, and can also be made to point to a different folder, by reconfiguring with the **~/Configuration/ConfigureLibraries.aspx** page.

Pictures are cached on demand, and in a lazy manner. The initial opening of a page will cache a picture large enough to be shown in the browser. As the user zooms in on the picture, the picture is magnified into larger resolution layers, and also cached. To hasten the serving to the browser of these magnified and thus larger pictures, each picture is diced into tiles and served to the browser based on what picture tile the user can see.

Embedded documents are also cached on demand. This makes for faster serving of large PDF documents for example.

Additional book graphics are also cached on demand for faster serving, which helps where these graphics are used on every page in things like parts list headers.

The cache will continue to grow as the library is walked by the users. Any out of date cached entity will be replaced if it is requested and the associated published book entity has been

republished. The cache can be deleted at any time, and the server will recommence caching of any requested entity.

With the release of WebView 3.4, a number of performance improvements have been added which affect caching. These improvements are all configurable in `Mincom.LinkOne.WebView.config` in the form:

```
<add key="<valueName>" value="<value>" enabled="true" />
```

New configuration value

Sliding window for caching books

```
Mincom.LinkOne.WebView.WebServer.Services.BookService.BookCacheWindow=<seconds>
```

New features

Start priming picture cache when a page is requested

```
Mincom.LinkOne.WebView.WebServer.Services.BookService.EnableBackgroundPictureCache=true/false
```

Start caching loading books while browsing the browser

```
Mincom.LinkOne.WebView.WebServer.Services.BookService.EnableBackgroundBookCache=true/false
```

and

```
Mincom.LinkOne.WebView.WebServer.Services.BookService.PrimeBookCacheWindow=<seconds>
```

Selection Lists

The LinkOne WebView selection list holds a list of parts, selected from LinkOne books. This list of parts can then be "checked out" to another application such as a parts requisitioning system.

A custom selection list may be developed in order to comply with corporate branding requirements and to integrate LinkOne WebView with other software applications, by developing a customized check-out procedure.

The types of customizations that are possible with the selection list include:

- The contents of the selection list, column headings.
- Mapping of parts list fields down to selection list columns.
- Multilingual support for column headings, toolbar tooltips, and parts list field mappings.
- Retrieval of information (such as part prices) from external systems as a part is added to the selection list.
- Creation of a check out procedure to pass the selection list contents on to another software application.

LinkOne WebView comes with a SAP Open Catalog Interface (OCI) compliant selection list, which can be seamlessly integrated with applications that support OCI punch-out.

The LinkOne WebView selection list is NOT compatible with a LinkOne WinView selection list, although it supports similar concepts such as:

- Configurable columns, headings, etc.
- Configurable parts list mappings.
- Language dependent behavior.

A LinkOne WebView selection list consists of a sub-folder in `~/SelectionLists` which contains:

- An XML definition file (`selectionlist.xml`) that defines the list field mappings

- Asp.net web pages forming the user interface

Supplied Selection Lists

LinkOne WebView is installed with the following selection lists:

Selection List Name	Description
Default	This selection list is identical to the default selection list provided with LinkOne WinView. This selection list is located in ~/SelectionLists/Default .
Sample Pricing	This selection list demonstrates how part prices can be retrieved from an external system via an ASP.NET script, as parts are added to the list. This selection list is located in ~/SelectionLists/SamplePricing .
OCI	The OCI selection list is a SAP OCI 3.0 compliant selection list. OCI defines an interface between SAP Business-to-Business Procurement and third-party catalog systems. OCI specifies a naming standard for fields in a compliant selection list. This can be seen in the OCI selection list definition file. The specification for OCI on the SAP web site at http://www.sap.com . This selection list is located in ~/SelectionLists/Oci .

OCI Selection List

The OCI selection list is a SAP OCI (3.0) compliant selection list. OCI defines an interface between SAP Business-to-Business Procurement and third-party catalog systems. The full OCI specification can be found on the SAP web site at www.sap.com. The LinkOne implementation of OCI for a selection list is located in **~/SelectionLists/Oci**.

OCI specifies a naming standard for fields in a compliant selection list which can be seen in the OCI selection list definition file at **~/SelectionLists/oci/selectionlist.xml**.

OCI Login

Login to an OCI enabled LinkOne WebView session is through the **~/OciLogin.aspx** address.

The login address and the parameters below form the setup in the Standard Call Structure for an SAP LinkOne Catalog Web Service.

The supported URL login parameters defined by OCI are:

Parameter name	Details	Default Value
HOOK_URL	The location that LinkOne WebView will post selection list data back to. This parameter is a mandatory requirement.	
~target	The target of the form data within the HOOK_URL. Additionally this value will be included in the form data as a field.	_top
~okcode	The code to include in the form data to indicate success.	ADDI

~caller	The source of the request. This value will be included in the form data.	CTLG
---------	--	------

Additional URL login parameters to support the LinkOne session that is created are:

Parameter Name	Details	Default Value
uid	The user name to login with.	
pwd	The user password to login with.	
theme	The web site theme to use for web page rendering.	default
lang	The two letter preferred language of the viewed book content.	en
selectionlist	The configuration name of the selection list to use.	oci
book	A specific book to open LinkOne at. If not specified, then LinkOne will open at the browse book page.	

Test OCI Login

LinkOne WebView provides a test harness page at `~/OciTest.aspx`. This page allows the user to test OCI by supplying various parameters. Using the test page with the default values, you can initiate a WebView session, select parts, and then post the parts back to the test page where the OCI output may be viewed.

Selection List Changes Since WebView 2.x

When customizing the selection list in LinkOne WebView 3.0, implementers are not able to simply copy a selection list folder from LinkOne WebView 2.x to LinkOne WebView 3.0. This is because the underlying architecture of Selection List functionality has changed in LinkOne WebView 3.0. The major differences between the two are:

- LinkOne WebView 3.0 no longer allows selection lists to inherit common data from another selection list. A selection list must be fully self contained in the selection list folder.
- LinkOne WebView 3.0 default selection lists now make use of Asp.net data grids to store / display and allow the editing of information.

Selection List Reference

Every selection list requires an XML definition file named `selectionlist.xml` that specifies:

- A mapping of fields from the parts list to columns in the selection list.
- The format of columns in the selection list.
- Translations for column headings and different parts list mappings for each supported language.

The definition file contains a selection list definition (`<selectionlistdefinition>`) for each supported language. Each selection list definition contains a list of the columns in the selection list (`<fields>`) and a list of field mappings that describe which parts list entries are moved to each column in the selection list (`<fieldmappings>`).

See `~/SelectionLists/Default/SelectionList.xml` for an example of the selection list format.

Tag	Tag Description	Attribute	Attribute Description
<selectionlistdefinition>	Encloses the entire selection list definition for a specific language.	lang=	Specifies the two-character ISO 639 language code for which this selection list definition applies. Use lang="" to specify that this selection list definition is the default definition to be used when a matching language code is not found.
<fields>	Encloses the list of <field> tags. These describe the number and type of columns in the selection list.		
<field>	Defines a column in the selection list.	title=	The heading for this column
		name=	The unique name for this column. This name is used to refer to the column in the field mappings.
		editable=	true false If true, the column is an editable text entry field. The value of the column is saved when the user clicks on Save . The default is false.
		hidden=	true false Specifies that a column not displayed in the browser. The default is false.

		requested_qty=	This identifies the field as holding the requested quantity - that is the quantity to be ordered. When you mark a field as requested_qty, it is not assigned any values from the selected parts list entry. Instead, it is assigned a value of "1" or the quantity fitted if the book specifies SELECT_QTY_FITTED=TRUE.
--	--	----------------	---

		<p>newline=</p>	<p>terminates kept removed blanked</p> <p>Determines how the selection list treats newline characters when a field is mapped from the selection list.</p> <p>terminates</p> <p>Specifies that this field will only take data from the corresponding parts list field up to the first line break. Any text after the first line break in the parts list will not be included in the Selection List field.</p> <p>kept</p> <p>Specifies that any line breaks in the corresponding parts list field will be retained when the data is added to the Selection List.</p> <p>removed</p> <p>Specifies that any line breaks in the corresponding parts list field will be removed when the data is added to the Selection List.</p> <p>blanked</p> <p>Specifies that any line breaks in the corresponding parts list field will be replaced with blanks when the data is added to the Selection List.</p> <p>The default is kept.</p>
--	--	-----------------	--

		ignore_bracketed_text=	<p>true false</p> <p>When true, specifies that any text within parenthesis in the corresponding parts list field will be removed when the data is added to the Selection List.</p> <p>The default is false.</p>
		criteria=	<p>true false</p> <p>Whether or not the part to be added requires this field to be present. If criteria is true for the column, and the part being added does not have a value for the field then the part can't be added to the selection list.</p>
		export_tag=	<p>The name used to identify the column in exported data. If no export tag is specified and the selection list is export the name value will be used.</p>
<fieldmappings>	Encloses the list of <map> tags.		
<map>	Defines a mapping from fields in the parts list to columns in the selection list.	pub=	<p>A publisher code for the field mapping. If not supplied, the map becomes the default map to which all unmatched mappings apply.</p>
		book=	<p>A book code, allowing the mapping to be specified down to the book level. Applies only in conjunction with a publisher code.</p>
<mapfield>	Defines the mapping for a specific selection list column.	name=	<p>The name of the column to which the mapping applies.</p>

		type=	bookfield listfield entryfield literal picturefield value empty The type of the field being mapped. A type of "empty" means that the column is unmapped.
--	--	-------	--

		<p>field=</p>	<p>The name of the field being mapped.</p> <p>Pre-defined book fields are:</p> <ul style="list-style-type: none"> • publisher • book • title <p>Pre-defined list fields are:</p> <ul style="list-style-type: none"> • title • group • page reference • page number <p>Pre-defined entry fields are:</p> <ul style="list-style-type: none"> • item id • part number • description • fitted • tagnumber • units • weight • alternative • link publisher • link book • link reference • link item <p>Pre-defined picture fields are:</p> <ul style="list-style-type: none"> • picture number • title <p>Pre-defined value fields are:</p> <ul style="list-style-type: none"> • book • book_title • booklocln • date • filename • list_title • option_filters • page • partdescr • partno • picture_title • pub • ref
		<p>32</p>	<ul style="list-style-type: none"> • time • visiblepage • visibleref

Book Libraries

Book libraries for LinkOne WebView are the same libraries that are used by LinkOne WinView and so can be shared by both applications at the same time.

For LinkOne WebView, all libraries are indexed Unicode libraries for high speed access, and use the publisher index file (man.lst) in each library folder and the book index file (book.lst) in each publisher folder. When a library is added using the web site library configuration page (**~/Configuration/ConfigureLibraries.aspx**), the library will be checked for an index, and a warning issued if no index is found. If you add books or publishers manually to an existing library, then you will need to re-index the library using the library configuration page.

Content Encoding

When a book is read from the book library, it requires decoding. The code page for decoding the text within the book is looked for in four places in the following order:

1. The publisher definition file (man.def) in the publisher folder. This requires a CODEPAGE,nnnn entry in that file. A typical entry for decoding Russian might be CODEPAGE,1251.
2. The book definition file (book.def) in the book folder. This requires a CODEPAGE,nnnn entry in that file. A typical entry for decoding Russian might be CODEPAGE,1251.
3. The book file (book.bbi) in the book folder. This is a binary file that can't be edited and was created with LinkOne Publisher when the book was published. If the book was published with LinkOne Publisher version 5.0 SP5 or later, then this file includes the code page used by the publisher and which is used to decode the book during loading.
4. The web site default global code page. This is configured with the **~/Configuration/ConfigureLibraries.aspx** page.

NOTE: If manual changes like CODEPAGE,nnnn entries are made to the publisher definition file (man.def) and or the book definition file (book.def), then you will need to re-index the library using the library configuration page.

Error Logging

Errors are caught by the error event handler in **~/Global.aspx** and redirected to LinkOne WebView Error Service. This service generates a unique error file containing the error details, and saves the file in the **~/Data/Log** folder (this folder can be chosen using the Reporting configuration page at **~/Configuration/ConfigureReporting.aspx**). The user is then redirected to the **~/Error.aspx** error handling.

Event Logging

Event Logging has been implemented to provide audit trail of significant sensitive administrative related events occurring within Linkone WebView. This logging has been implemented utilising the log4net logging library and the configuration can be found in the web.config file under <log4net> directive. It can be turned off/disabled by setting the setting <level value="OFF" />

The default location of the log files is the **~/Data/Log** directory with the log files being changed/rolled over every 2MB in size.

The following log events are recorded.

User Events;

- Login

- Logout
- AddUser
- DeleteUser
- UnlockUser
- AddUserRole
- SetUserAllowedAllBooks
- AddUserAllowedBooks
- DeleteUserAllowedBook
- SetUserAllowedAllNotes
- AddUserAllowedNote
- DeleteUserAllowedNote
- SetUserPolicy
- SetUserPassword (Administrator changing users password)
- ChangeUserPassword (User changing their password).

Role Events;

- SetRoleAllowedAllBooks
- AddRoleAllowedBooks
- DeleteRoleAllowedBook
- SetRoleAllowedAllNotes
- AddRoleAllowedNotes
- DeleteRoleAllowedNotes
- SetRolePolicy

Backing Up Backup

Generally, you should perform periodic backup of:

- The LinkOne WebView configuration file (**~/Mincom.LinkOne.WebView.config**)
- The LinkOne WebView membership, role, and profile database (**~/App_Data/ASPNETDB.mdf**)
- Any web site files that have been customized or added

Backup of the database can be performed on the web server with suitable changes to the following:

```
osql -E -Q "BACKUP DATABASE aspnetdb TO DISK='C:\aspnetdb.bak'"
```

To backup a specific server, you can add:

```
-S <databaseservername>
```

Restoration

Restoration of the database can be performed on the web server with suitable changes to the following:

```
osql -E -Q "RESTORE DATABASE aspnetdb FROM DISK='C:\aspnetdb.bak'"
```

To restore for a specific server, you can add:

`-S <databaseservername>`

Tools

Clear Cache Tool

Clearing selected parts of the cache can be done manually from a command prompt on the web server, by running the `~/bin/ClearCacheTool.exe` tool. Using this tool allows the cached contents of a book, to be deleted without affecting the rest of the cache contents. By default the tool will not delete anything unless the `-delete` argument is provided, it will only report the root directory that will be deleted. The command line arguments are described below.

Argument	Description
-library=path	<p>The path of the library containing the book to be removed from cache. The path must be in the same form as specified in the Configuration\Libraries setting in WebView.</p> <ul style="list-style-type: none"> - Absolute path (eg: C:\library, or \\server\share) <p>The argument is mandatory and there is no default value.</p>
-publisher=PublisherCode	<p>The publisher code of the publisher whose book is to be removed from the cache.</p> <p>The argument is mandatory and there is no default value.</p>
-book=BookCode	<p>The book code of the book whose content is to be removed from the cache.</p> <p>The argument is mandatory and there is no default value.</p>
-report=file	<p>The report file to be used. The file can be in the form:</p> <ul style="list-style-type: none"> - Absolute path (eg: C:\reindex.log, or \\server\share\reindex.log) - Relative path to the web site (eg: Data\Log\reindex.log, or ~/Data/Log/reindex.log) <p>The argument is optional and there is no default value.</p>
-delete	Delete the specified part of the cache.

Library Re-Indexer

LinkOne WebView uses indexed LinkOne book libraries for faster access to the libraries. Whenever books are added to or removed from a library, the library must be re-indexed, otherwise any added book won't be seen, and any removed book will continue to be seen.

Re-indexing can be done manually through the web site, by logging in with an Administrator role, and navigating to `~/Configuration/ConfigureLibraries.aspx`, and clicking the **Re-Index** tool next to the library that needs re-indexing.

Re-indexing can also be done manually from a command prompt on the web server, by running the `~/bin/ReindexTool.exe` tool. Using this tool allows re-indexing from scripts or scheduled tasks. The command line arguments are described below.

Argument	Description
-library=path	<p>The path to the library that is to be re-indexed. The path can be in the form:</p> <ul style="list-style-type: none"> - Absolute path (eg: C:\library, or \\server\share) - Relative path to the web site (eg: Data\Libraries\Mincom, or ~/Data/Libraries/Mincom) <p>The argument is mandatory and there is no default value.</p>
-reporting=None Errors Summary Detailed	<p>The reporting level used to write to the console and to any specified report file.</p> <p>The argument is optional and the default value is None.</p>
-report=file	<p>The report file used when the reporting level is not None. The file can be in the form:</p> <ul style="list-style-type: none"> - Absolute path (eg: C:\reindex.log, or \\server\share\reindex.log) - Relative path to the web site (eg: Data\Log\reindex.log, or ~/Data/Log/reindex.log) <p>The argument is optional and there is no default value.</p>
-codepage=#	<p>The code page for decoding any old books.</p> <p>The argument is optional and the default value is the current system code page from Code Page for Non-Unicode Programs in Control Panel > Regional Settings.</p>

LinkOne Book Indexing Service

The Advanced Search Facility uses a SOLR database to hold indexes to the documents in the LinkOne libraries. Before the Advanced Search Facility can be used, the LinkOne libraries must be indexed. This is done by the System Administrator starting the `~/bin/LinkOneBookIndexingService` Windows Service on the web server.

Note that when any book in the libraries is modified, or a book is deleted, or a new book is added, the **LinkOneBookIndexingService** will automatically reindex the books as expected.

More detailed information, e.g how to enable logging, can be found in the **How to Configure Advanced Search** section

Migrating User Notes From WebView 2.2

Introduction

LinkOne provides the ability for users to create notes, both private and public, relating to particular entities, such as parts or a book. In LinkOne WebView 2.2, these notes are stored in the database, whereas in WebView 3, they are stored in a file based structure.

A set of tools has been provided with WebView 3 to assist with the transfer of public and private notes from WebView 2.2 to WebView 3.

NOTE: It is highly recommended that the tools are executed and tested in a non-production system prior to running in a production system.

The tools are run as console applications from a Windows command prompt. The Export tool produces an XML file containing the user notes from the WebView 2.2 system and the Import tool reads an XML file produced by the Export tool and loads the user notes into the WebView 3 User Notes area.

Transferring User Notes

The tools are installed with the WebView 3 installation into the Bin directory under the web site. There is an export tool to extract the user notes from the WebView 2.2 database and an import tool to create the notes retrieved by the export tool into the WebView 3 location.

- Export Tool : **NotesMigrationTool.Export.exe**
- Import Tool : **NotesMigrationTool.Import.exe**

Installation of Export Tool into WebView 2.2 Site

For the WebView 2.2 Export tool to run correctly, it needs to be located in the same directory as the WebView 2.2 web server assemblies. This path would be similar to

C:\Program Files\Mincom\Linkone\WebView\WebSite

NOTE: For this release, the tool is compiled for use with WebView V2.2 SP12.

Copy the export tool from the WebView 3 install Bin directory to the correct location for WebView 2.2.

Export the User Notes from WebView 2.2

The Export tool is a console application and is run from a command prompt. Open up a command prompt and navigate to the folder where the Export tool has been copied. The definition for the Export tool follows.

NotesMigrationTool.Export

Syntax :	NotesMigrationTool.Export.exe <output notes file name>
Parameters :	output notes file name - path name to the XML file where the user notes are to be written.
Remarks :	The <output notes file name> is the path name of an XML file to contain the user notes from the WebView 2.2 implementation. If the file does not exist, it will be created. If it does exist, it will be overwritten by the execution of this command.

The following is an example of the export command:

```
NotesMigrationTool.Export.exe c:\data\linkone\notes.xml
```

Export Remarks

The export program is a passive program and only reads data from the LinkOne books database. Therefore it can be run as many times as required without affecting the WebView 2 data.

The database connection details are obtained by the Export program by reading the configuration information of the web server. This directs the program to the Application config file located in the user writeable area for the AllUsers entity.

The user defined in the configuration files for the database connection string must be an administrator of the database.

The user executing the tool must have permission to create/modify the XML file name provided. If the file exists, the contents will be overwritten.

Back Up the WebView 3 User Notes Area

NOTE : It is important to back up the User Notes area before running this tool.

The location of the User Notes are for both Public and Private notes and can be found by selecting the Configuration / Notes item on the WebView 3 main menu.

The user notes, public and private, are normally stored by default in a DATA directory under the web site path. This can be configured by the user to point to a different location. The program reads this configuration information for the notes path from information in the web configuration file associated with the web server.

Import the User Notes to WebView 3

The Import tool is a console application and is run from a command prompt. Open up a command prompt and navigate to the WebView 3 install Bin folder where the Import tool has been installed.

The following is the definition for the Import tool

NotesMigrationTool.Import

Syntax	NotesMigrationTool.Import.exe <input notes file name> <web server path>
Parameters	input notes file name – path name to the XML file produced by the export tool. web server path – path name to the web server install location.
Remarks	The <input notes file name> is the path name of an XML file produced by the export tool containing the user notes from the WebView 2.2 implementation. If the file does not exist, an error is output and the execution is terminated. The <web server path> is the location of the WebView 3 web site.

The following is an example of the import command

```
NotesMigrationTool.Import.exe c:\data\linkone\notes.xml c:\inetpub\wwwroot\LinkOne3
```

Import Remarks

Backing up any User Notes already added to the WebView 3 installation is important in case of any overwriting of existing User Notes by the tool.

Copy the XML user notes file produced by the WebView 2 export program to a location which is accessible to the WebView 3 import program.

The User executing the program must have write permission to where the User Notes are stored which is either the default web site DATA directory or the location defined in the Notes configuration item if different from the default. If the default directory is used in this example, the location might be

```
C:\inetpub\wwwroot\LinkOne3\Data\PublicNotes
C:\inetpub\wwwroot\LinkOne3\Data\PrivateNotes
```

The web server path to be used as the second parameter is the install location of the LinkOne web server. In the example above, the location might be

```
C:\inetpub\wwwroot\LinkOne3
```

Library Synchronizer

LinkOne WebView uses local LinkOne book libraries, either on the local disk, or on network file share. Each of these libraries can be synchronized with a master library to ensure consistency between libraries. For example, synchronizing the a library to the master library of the book publisher.

Synchronizing can be done manually from a command prompt on the web server, by running the `~/bin/BookSyncTool.exe` tool. Using this tool allows synchronizing from scripts or scheduled tasks. The command line arguments are described below.

BookSyncTool performs a complete library synchronize, and the result leaves the local library with a mirror copy of the master library, including adding new books locally that are now found in the master library and removing old local books no longer found in the master library. This is the equivalent of using the LinkOne WinView BookSync application, and setting the local library Synchronization Type to Synchronize All Books, and then performing a Synchronize Library.

Argument	Description
-library=path	<p>The path to the target library that is to be synchronized to. The path can be in the form:</p> <ul style="list-style-type: none"> - Absolute path (eg: C:\library, or \\server\share) - Relative path to the web site (eg: Data\Libraries\Mincom, or ~/Data/Libraries/Mincom) <p>The argument is mandatory and there is no default value.</p>

<p>-sync=path</p>	<p>The path to the source library that is to be synchronized from. The path can be in the form:</p> <ul style="list-style-type: none"> - http or https path (eg: http://www.domain.com/booklibrary) - unc path (eg: \\server\share\library) - local path (eg: D:\masterlibrary) <p>The argument is optional and if omitted it will be taken from the remote.config file found in the target library path. The remote.config file is configured and written by the LinkOne WinView BookSync application.</p>
<p>-reporting=None Errors Summary Detailed</p>	<p>The reporting level used to write to the console and to any specified report file.</p> <p>The argument is optional and the default value is None.</p>
<p>-report=file</p>	<p>The report file used when the reporting level is not None. The file can be in the form:</p> <ul style="list-style-type: none"> - Absolute path (eg: C:\booksync.log, or \\server\share\booksync.log) - Relative path to the web site (eg: Data\Log\booksync.log, or ~/Data/Log/booksync.log) <p>The argument is optional and there is no default value.</p>
<p>-codepage=#</p>	<p>The code page for decoding any old books.</p> <p>The argument is optional and the default value is the current system code page from Code Page for Non-Unicode Programs in Control Panel > Regional Settings.</p>
<p>-proxyusername=username</p>	<p>The user name that the BookSyncTool will use for proxy authentication.</p> <p>The argument is optional and there is no default value.</p>
<p>-proxypassword=password</p>	<p>The password that the BookSyncTool will use for proxy authentication.</p> <p>The argument is optional and there is no default value.</p>
<p>-username=username</p>	<p>The user name that the BookSyncTool will use for site authentication to access the remote library.</p> <p>The argument is optional and there is no default value.</p>

-password=password	The password that the BookSyncTool will use for site authentication to access the remote library. The argument is optional and there is no default value.
---------------------------	---

Samples

Silent Login

Included in the LinkOne WebView web application is a sample silent login page which can be used to bypass the login page. To use this page the caller must pass the user name, password and any login options as query parameters on the URL, or as form parameters in a post. If the passed user name and password can't be authenticated, then the user will be re-directed to the login page.

The silent login page can be found at `~/SilentLogin.aspx`.

Login parameters that can be passed are:

Parameter	Description	Default Value
uid	The user name to login as. For example uid=myusername	
pwd	The password for the given user. For example pwd=mypassword	
language	The preferred book language. For example language=de	en
theme	The web site theme to use. For example theme=blue	default
selectionlist	The selection list configuration to use. For example selectionlist=default	default
book	The book to open, given as <publisher_code>/<book_code>. For example book=mypublisher/mybook If a book is specified then it will be opened upon a successful login, otherwise the browser will be directed to the browse books page.	
page	The page to open, given as <page_reference>. For example page=page1 If a page is specified then a book must also be specified. The page will be opened upon a successful login.	
item	The item to highlight, given as <item_identifier> For example item=11223344 If an item is specified then a book and page must also be specified. The item must be present on the specified page, or no item will be highlighted.	

part	<p>The part to highlight, given as <part_number></p> <p>For example part=123.456-789</p> <p>If a part is specified then a book must also be specified, and page and item are ignored. If the part is found more than once in the book, then the first part found is highlighted.</p>	
option	<p>Any filter to apply to the book. Filters are formatted as <option_code>:<option_value>.</p> <p>For example option=s:off</p> <p>For multi-select options, values are separated by a '/' character.</p> <p>For example to set values a, b and c for option myoption, the parameter is option=myoption:a/b/c</p> <p>Supply one option= parameter for each option to be set.</p> <p>For example option=s:off&option=sn:12345</p>	

Examples of query parameters on a URL:

```
http://server/linkone/silentlogin.aspx?uid=admin&pwd=admin&theme=blue
http://server/linkone/silentlogin.aspx?uid=user&pwd=user&language=de&book=pub
/book&option=sn:12345&model:coldweather
```

Example of query parameters in a post:

```
<html>
  <body>
    <form action="SilentLogin.aspx" method="post">
      <input name="uid" value="user"/>
      <input name="pwd" value="user"/>
      <input name="language" value="de"/>
      <input name="book" value="pub/book"/>
      <input name="option" value="sn:12345"/>
      <input name="option" value="model:coldweather"/>
    </form>
  </body>
</html>
```

Pricing Selection List

One of the standard selection lists installed with LinkOne WebView is a sample pricing selection list. To use this selection list, choose it at the login web page from the available selection list types.

When any part is added to the selection list, a constant price of \$100 is shown. There are two parts to this operation:

1. A Price field in the selection list field mapping fields found at **~/SelectionLists/SamplePricing/selectionlist.xml** which is declared as a simple text field (not mapped from a field in the added part).
2. Server-side code which sets the value of the Price field to \$100 when a part is added. This code is in **~/SelectionLists/SamplePricing/Default.aspx.cs**.

Extensions to this simple sample would be to use a price lookup through an external system, or to add other fields to the selection list such as Stock On Hand.

Walkthroughs

How To Authenticate Using Active Directory

LinkOne WebView is capable of using Forms Authentication to authenticate a user against an Active Directory. To use Active directory to authenticate users, edit the `~/web.config` file and make the following changes:

1. Un-comment the Active directory connection string, and set the connection string to your active directory:

```
<add name="ADService" connectionString="-Insert Connection String Here-" />
```

2. Comment out the **AspNetSqlMembershipProvider** provider from the membership providers section.

3. Set the **defaultProvider** attribute in the membership element to **AspNetActiveDirectoryMembershipProvider**:

```
<membership defaultProvider="AspNetActiveDirectoryMembershipProvider" >
```

4. Un-comment the **AspNetActiveDirectoryMembershipProvider** provider in the membership providers section. For information regarding configuration settings for the Active Directory Membership Provider please see <http://msdn.microsoft.com/en-us/library/system.web.security.activedirectorymembershipprovider.aspx>. The following is a sample **AspNetActiveDirectoryMembershipProvider** connection element:

```
<add name="AspNetActiveDirectoryMembershipProvider"
type="System.Web.Security.ActiveDirectoryMembershipProvider, System.Web,
Version=2.0.0.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a"
connectionStringName="ADService"
connectionUsername="CN=ADUser,OU=Users,O=WebView"
connectionPassword="ADUserPassword" connectionProtection="None"
enableSearchMethods="true" requiresUniqueEmail="false"
maxInvalidPasswordAttempts="5" minRequiredPasswordLength="4"
minRequiredNonalphanumericCharacters="0" passwordStrengthRegularExpression=""
enablePasswordReset="true" requiresQuestionAndAnswer="true"
attributeMapPasswordQuestion="pwdQuestion"
attributeMapPasswordAnswer="pwdAnswer"
attributeMapFailedPasswordAnswerCount="badPwdCount"
attributeMapFailedPasswordAnswerTime="badPwdTime"
attributeMapFailedPasswordAnswerLockoutTime="badPwdLockoutTime" />
```

5. Un-comment the **machineKey** element in the system web section. Note: for security purposes it is recommended that the encryption and decryption key pair are replaced with a unique set. In order to use Active Directory authentication the machine key cannot be auto generated. For more information regarding the **machineKey** element please see <http://msdn.microsoft.com/en-us/library/w8h3skw9.aspx>

In order to use Active Directory to administer users, the web application requires an account with privileges to add, remove and update users in the Active Directory data store. Additionally, custom configuration of the Active Directory schema may be required in order to map required attributes.

NOTE: If Active Directory is used for membership, then in order to allow administrators to reset user passwords via the `~/Configuration/ConfigureUsers.aspx` web page, a security question and answer pair must be provided for the user and the administrator will be required to correctly answer the question. If this is not possible, then the **allowPasswordReset** and **requiresSecurityQuestionAndAnswer** should be set to false. This will remove the Authorization pane from the `~/Configuration/ConfigureUsers.aspx` web page.

How To Determine Chosen Microsoft .Net

Version

For performance reasons, the use of .Net 3.5 is currently recommended. However using different .Net versions requires different web configurations. At install time the installer will install a `~/web.config` file that is compatible with the Microsoft .Net version that is configured for the Application Pool on IIS7, 8, or the selected web site on IIS 6. Note that when running with Microsoft .Net 3.5 SP1, the CLR version is actually 2.0.

The table below outlines what version of the web.config file will be used by default for different configurations.

IIS Version	Microsoft .Net 3.5 SP1 Installed?	Web Site Default Asp.Net Version	Default Application Pool CLR Version	Installed web.config file
6	Yes	2	-	Microsoft .Net 3.5 configuration
6	Yes	2	-	Microsoft .Net 3.5 configuration
7, 8	Yes	-	2	Microsoft .Net 3.5 configuration
7, 8	Yes	-	2	Microsoft .Net 3.5 configuration

In addition to installing the web.config file, the installer also adds a web.net35.config file to the application folder to allow for simple re-configuration when changing between versions of Microsoft .Net.

How To Authenticate Using Windows Authentication

LinkOne WebView is capable of using Windows Authentication to authenticate users. To use Windows authentication to authenticate users, edit the `~/web.config` file and make the following changes:

1. Replace the Authentication tag with:

```
<authentication mode="Windows" />
```

Note: In order for a Windows user to be able to log into LinkOne WebView the domain qualified user name must be added to the allowed users roles. This can be performed by navigating to the `~/Configuration/ConfigureUsers.aspx`.

Creating a Web Site Theme

LinkOne WebView supports using themes to create a customized look and feel for the web site. To achieve a customizable look and feel LinkOne WebView utilizes the Microsoft Asp.Net theme architecture. More information about using Asp.net themes can be found at <http://msdn.microsoft.com/en-us/library/wcyt4fxb.aspx>.

To modify the default theme to show a new organizational banner:

1. Copy the `~/App_Themes/Default` folder to another folder in `~/App_Themes`, and give the copied folder the name of the new theme, e.g. **MyTheme**.
2. Replace `~/App_Themes/MyTheme/images/product.png` with the desired corporate logo.
3. Edit the `~/web.config` file to use the new theme by replacing the word **Default** with the name of your new theme in the following 2 lines e.g.

```
<add name="SiteTheme" type="System.String" defaultValue="MyTheme"/>
```

```
and
<pages theme="MyTheme">
```

The theme colors can also be modified by editing the CSS colors such as #005EA8, #C8D9ED or #005EA8 found in `~/App_Themes/<new_theme>/StyleSheet.css`.

How To Add New Translations

LinkOne WebView supports adding translations to web site pages. Standard Microsoft Asp.net practices are used to manage web site translations, and more information can be found at <http://msdn.microsoft.com/en-us/library/c6zyy3s9.aspx>.

To add or modify translations to a web page (.aspx file):

1. Add the desired content to the .aspx web page and add the **meta:resource** key attribute to the element.

For example to include a translatable label in a page add the following element:

```
<asp:Label runat="server" id="MyLabel" meta:resourceKey="MyLabel" Text="The
Label Text" />
```

2. In the same folder that contains the .aspx web page ensure there is a folder called **App_LocalResources**. Also ensure that the folder contains a resource file (.resx file) that has the naming convention of **<aspx_file_name>.<language_code>.resx**.
3. Add the translated values to the resource file as **data** elements with a **name** attribute value of the form **<resource_key>.<property>**.

For example to translate the text value of the label above, add the following content:

```
<data name="MyLabel.Text" xml:space="preserve">
  <value>My Translated Label Text</value>
</data>
```

How To Change the Book or Selection List Menus

The menus on the LinkOne WebView book viewer web page can be changed by adding, removing, or modifying menu items. This allows adding new functionality or removing unwanted functionality.

Each **asp:MenuItem** in the book viewer web page at `~/Viewer/book.aspx` and the selection list web pages at `~/SelectionLists/<name>/default.aspx` has these special attributes:

asp:MenuItem Attribute	Purpose
Text	The text is translated within the associated App_LocalResources/<page_name>.<language_code>.resx for the associated language.
ImageUrl	This attribute references an image in the current theme's web site folder.
Value	<p>This attribute is used to check access to the menu item through authorization. If the user is not authorized to access the menu, then the menu item is removed.</p> <p>The values can be:</p> <ul style="list-style-type: none"> • ? for any user to access; • * for any authenticated user to access; • a specific role for only those users with in that role to access

How To Change the Mapped Selection List Fields

LinkOne WebView supports a customizable selection list field map to suit the individual needs of the organization.

To modify a selection list field map, firstly locate the **selectionlist.xml** file in the folder of the selection list being used. For example for the default selection list, this will be **~/SelectionLists/Default/selectionlist.xml**. The selectionlist.xml file contains the field definitions and the field mappings that map part fields to selection list fields.

To change a part field that is mapped to a selection list field:

1. Locate the <selectionlist> element for the wanted language (different fields can be mapped depending on the user language).
2. Now within that, locate the <map> element for the wanted source book (different fields can be mapped depending on the viewed publisher or book).
3. Now within that, locate the <mapfield> element that names the destination selection list field.
4. Now update the source part field name to the new source name.

Multiple <map> elements can be added, to target specific publishers or books.

Note: Localized selection list field mappings do not inherit from the selection list definition from the neutral language, so each column and field mapping to be used must be defined for each available language in order to be universally used.

How To Create a Custom Selection List

To create a customized selection list:

1. Create a new folder to contain the selection list in the **~/SelectionLists** folder.
2. Implement your custom selection list behavior in the new folder. **Tip:** Copy the contents of the **~/SelectionLists/Default** selection list folder and use it as a starting point.
3. Login to LinkOne WebView and browse to **~/Configuration/ConfigureSelectionLists.aspx**.
4. Add the newly created selection list, giving it a unique name, and using the folder path created in step 1.
5. Specify whether or not the selection list is persisted across sessions.
6. Click **Add** to add the new selection list configuration.

How To Create a Custom Security Provider

Identifying a user and what access rights and policies they have, is the functionality of the security provider. A default security provider is supplied, with a class name of **Mincom.LinkOne.WebView.WebServer.Security.Principal**. This class provides:

- the name of the current user, as identified by **System.Web.HttpContext.Current.User.Identity.Name**
- the allowed books to access, as identified by **Mincom.LinkOne.WebView.Services.SecurityService.Current.AllowedBooks**
- the allowed notes to access, as identified by **Mincom.LinkOne.WebView.Services.SecurityService.Current.AllowedNotes**
- the policy settings, as identified by **Mincom.LinkOne.WebView.Services.SecurityService.Current.GetUserPolicyResolved()**

LinkOne WebView determines the class name of the provider to use, by looking at the **Mincom.LinkOne.WebView.WebServer.Services.SecurityService.PrincipalType** appsetting in the **Mincom.LinkOne.WebView.config** file. The default security provider is used if no class name is given, or the given class can't be constructed.

The given class must implement **Mincom.LinkOne.Dom.IPrincipal2**.

Following is a sample implementation of this interface.

```
using System;
using System.Collections.Generic;
using System.Web;
using Mincom.LinkOne.Dom;
using Mincom.LinkOne.Dom.Notes;
namespace Mincom.LinkOne.WebView.Samples
{
    public class Principal : Dom.IPrincipal2
    {
        private string _userId;
        public Principal()
        {
            _userId =
HttpContext.Current.User.Identity.Name.ToLowerInvariant();
        }
        public string UserId
        {
            get { return _userId; }
        }
        public bool CanAccessBook( string publisherCode, string bookCode,
IList<string> categories )
        {
            bool canAccessBook = false;
            // check if the user is "admin"
            if( _userId == "admin" )
            {
                canAccessBook = true;
            }
            // check if the user is "fred"
            else if( _userId == "fred" )
            {
                // if the publisher code is "Kubota"
                // bookCode is "101a0001"
                // number of categories = 1
                // first category = "tb"
                if( publisherCode.ToLowerInvariant() == "kubota"
                    && bookCode.ToLowerInvariant() == "101a0001"
                    && categories.Count == 1
                    && categories[ 0 ].ToLowerInvariant() == "tb" )
                {
                    // set the book as accessible by the user
                    canAccessBook = true;
                }
            }
            return canAccessBook;
        }
        public IList<string> Licenses
        {
            get
            {
                List<string> result = new List<string>();
                // if the user is "admin"
                if( _userId == "admin" )
                {
                    result.Add( "Administrators-License1" );
                    result.Add( "Administrators-License2" );
                    result.Add( "ABC-ABC-ABC-ABC-ABC-ABC" );
                }
                // if the user is "fred"
                else if( _userId == "fred" )
            }
        }
    }
}
```

```

        {
            result.Add( "LicenseValue1-A1" );
            result.Add( "LicenseValue2-A2" );
            result.Add( "LicenseValue3-A3" );
        }
        return result;
    }
}
public bool CanAccessNote( INote note )
{
    return true;
}
public bool CanAccessCategory( ICategory category )
{
    return true;
}
public bool IsPolicyAllowed( string policy )
{
    return false;
}
public object Security
{
    get { return null; }
}
public string CacheKey
{
    get { return String.Empty; }
}
public string PrivateNotesDirectory
{
    get { return
Mincom.LinkOne.WebView.WebServer.Services.NoteService.Current.PhysicalPrivateNotesD
irectory; }
}
public string PublicNotesDirectory
{
    get { return
Mincom.LinkOne.WebView.WebServer.Services.NoteService.Current.PhysicalPublicNotesDi
rectory; }
}
}
}

```

How To Create a Custom Policy

Additional policies may be added to LinkOne WebView for whatever reason you need. To create additional policies, complete the following steps:

1. Create the policy for all roles and users by adding tags to `~/Web.config` in the section **configuration/system.web/profile/properties**.

The policy name must be prefixed with **Policy**.

The type must be **System.Boolean**.

For example, adding a policy named **AllowPartPurchasing** looks like:

```

<?xml version="1.0"?>
<configuration>
  ...
  <system.web>
    ...
    <profile enabled="true">
      ...
      <properties>
        ...
        <add name="PolicyAllowPartPurchasing" type="System.Boolean"
defaultValue="False" />
      </properties>
    </profile>
  </system.web>
</configuration>

```

```

    ...
  </system.web>
  ...
</configuration>

```

2. Provide a user readable name for policy configuration by adding a localized policy message to the local resource files (.resx files) of the Configure Users and Configure Roles controls.

The English resource files for these controls are

~/Controls/App_LocalResources/ConfigureUsers.ascx.resx and
~/Controls/App_LocalResources/ConfigureRoles.ascx.resx.

The controls can be viewed on the **~/Configuration/ConfigureUsers.aspx** and
~/Configuration/ConfigureRoles.aspx web pages.

If no localized policy message is found then the policy name itself will be displayed.

The policy message resource key must be prefixed with **PolicyMessage_**.

For example, the **AllowPartPurchasing** policy will have a resource key of **PolicyMessage_AllowPartPurchasing** and will look like:

```

<data name="PolicyMessage_AllowPartPurchasing" xml:space="preserve">
  <value>Allow purchasing of parts</value>
</data>

```

3. To make use of the policy programmatically, use the name of the policy and make a call to:

```

Mincom.LinkOne.WebView.WebServer.Services.SecurityService.Current.IsPolicyAllowed("AllowPartPurchasing")

```

How To Share Notes with LinkOne WinView

The notes folders for LinkOne WebView are configured on the **~/Configuration/ConfigureNotes.aspx** web page.

The private notes folder should be:

- For Windows XP, and Windows 2003:
C:\Documents and Settings\{0}\My Documents\LinkOne\Notes
- For Windows Vista, Windows 2008, and Windows 7:
C:\Users\{0}\Documents\LinkOne\Notes

The public notes folder should be:

- For Windows XP, and Windows 2003:
C:\Documents and Settings\All Users\My Documents\LinkOne\Notes
- For Windows Vista, Windows 2008, and Windows 7:
C:\Users\Public\Documents\LinkOne\Notes

Additional steps required are:

- Set permissions on each Notes folder to allow the ASPNET (XP, 2003) or IUSR (Vista, 2008, 7) account to read and write the folder contents.
- Ensure the Windows account name of each user matches the WebView account name, this is because the {0} in the folder name above is substituted with the user account name.

How To Use SQL Server Instead of SQL Express

The standard installation makes use of SQL Express for its database server. The Express editions are cut down versions of the Server editions, but are ideally suited to the membership, role, and profile management required for LinkOne WebView.

In cases where SQL Server really is preferred, or in cases where a shared instance is required for multiple LinkOne WebView web servers, then follow the steps below

1. Complete a standard installation of LinkOne WebView.

2. Move the databases to SQL Server.
 1. Start SQL Server Management Studio
 2. Attach the database `~/App_Data/ASPNETDB.mdf` as ASPNETDB
 3. Attach the database `~/App_Data/LINKONEDB.mdf` as LINKONEDB
 4. Create a login for **NT AUTHORITY\IUSR** and **NT AUTHORITY\NETWORK SERVICE** (i.e. the user or group that the IIS application Pool is running as)
 5. Modify the login **User Mappings** and map ASPNETDB and LINKONEDB, giving each mapping the **dbo** role
3. Update the location of the databases in the LinkOne WebView configuration.
 1. Modify `~/web.config`.
 2. Uncomment the SQL server connection string for LocalSqlServer (modify the data source to the SQL Server instance name if not the default instance)
 3. Uncomment the SQL server connection string for LinkOneSqlServer (modify the data source to the SQL Server instance name if not the default instance)
 4. Comment out the SQL Express connection string for LocalSqlServer
 5. Comment out the SQL Express connection string for LinkOneSqlServer

How to Install Mass Library Replication

In this section, you will find information on configuring the replication of LinkOne Libraries from one location to another. An example of this intended usage might be the replication of whole libraries between remote LinkOne WebView web servers. Rsync will be used via an ssh shell on the server machine.

This is accomplished by using the defacto standard in mass file replication, Rsync. Rsync allows only file changes to be transmitted across the wire, enabling fast and efficient transfer.

If you need further guidance, contact your LinkOne reseller or **ABB Enterprise Software Customer Support mailto:CustomerCare.ES@abb.com** (if you purchased LinkOne directly from ABB). You can get help in answering your questions as well as provide feedback and suggestions on the **OCC** (<https://enterprisesoftware.force.com/customerportal/login>).

Cygwin, Rsync and OpenSSH Installation

These instructions should be followed to install Cygwin, Rsync and OpenSSH on both the client and server machines. These instructions were compiled using Cygwin setup.exe version 2.769 and cygwin.dll version 1.7.11-1. These instructions may not be accurate for other versions.

Download Cygwin's setup.exe from <http://www.cygwin.com>.

1. On the server machine: Login under a **LOCAL** Administrator account (**NOT** a domain account).
On the client machine: Login using any account.
2. Run setup.exe as administrator. If you are not part of the Administrator group, on setup.exe use "Right click -> run as Administrator".
3. Once in the installer press **Next** 3 times.
4. Select a local package directory choose a temporary folder to store the packages which can be deleted after install is complete.
5. Press **Next** twice.
6. Select any mirror.
7. Press **Next**.
8. When asked what components to install select the following by clicking on "**Skip**" so that a cross appears in the "**Binary**" column:
 - +Admin -> Cygrunsrv
 - +Net -> Rsync

+Net-> OpenSSH

9. Press **Next**.
10. Press **Finish**.
11. Add the following to the end of the PATH system environment variable ";c:\cygwin\bin"

OpenSSH Server Configuration

These instructions configure an SSH server on the server machine.

1. Open the Cygwin terminal by selecting "Start -> All Programs -> Cygwin -> Cygwin Terminal".
2. Type "**ssh-host-config**" and press **Enter**.
3. Answer "**yes**" to all questions until the next question.
4. When the script asks for the value of CYGWIN for the daemon, type "**ntsec**" and press **Enter**.
5. Answer "**no**" to the question "**do you want to use a different name?**".
6. Answer "**yes**" to all further questions until the next step.
7. Enter an appropriate password when asked

NOTE : Please be sure that this password matches the password rules given on your machine.

8. Type "**cyglsa-config**" and press enter.
9. Reboot the computer
10. Open the Cygwin terminal again.
11. Type "**net start sshd**" and press **Enter**.
12. If the above command fails go into "Control Panel -> Administrative tools -> Services", double click on the "CYGWIN sshd" service and reconfigure the username and password under the "Log On" tab. Try step 11 again.
13. Type "**mkpasswd -cl > /etc/passwd**" and press **Enter**.
14. Type "**mkgroup --local > /etc/group**" and press **Enter**.
15. Type "**ssh localhost**" and press **Enter**.

This should connect to the local ssh server.

SSH Public Key Authentication

1. Open a Cygwin terminal on the client machine.
2. Type "**ssh-keygen -t rsa**" and press **Enter**.
3. Press **Enter** when asked for any inputs.
4. Type
Tscp /.ssh/id_rsa.pub **serverusername@hostname:/home/serverusername/.ssh"**

Where:

serverusername is any username on the server machine. (this is the account Rsync will run under).

hostname is the server machine name.

5. Open an RDP session to the server machine.
6. Open an explorer window and browse to "c:\cygwin\home**serverusername**\.ssh".
7. Create two new text files called "**authorized_keys**" and "**authorized_keys2**"

There is no file extension!

8. Open the file "**id_rsa.pub**" in notepad and copy all contents to the above two files.
9. On the client machine type "ssh **serverusername@hostname**" and press **Enter**.

You should now be connected to the server via ssh public key authentication without having to enter a password.

Rsync Client Configuration

The following instruction can be configured as a scheduled task on the client machine to replicate the library at "`c:\linkone\library`" from the server to the client directory "`c:\linkone`" via an ssh shell on the server.

```
rsync -az --delete serverusername@hostname:/cygdrive/c/linkone/library
/cygdrive/c/linkone/
```

Where:

serverusername is any username on the server machine. (this is the account Rsync will run under).

hostname is the server machine name.

How To Install on x64 Windows

LinkOne WebView is a 32bit web application, and can't run on 64bit Windows 2003 or 64bit Windows XP without enabling 32bit operation on the server.

Once LinkOne WebView is installed, follow the instructions below to enable 32bit operation. These instructions and how to disable 32bit operation can be found at <http://support.microsoft.com/kb/894435>.

NOTE: This only applies to Windows Server 2003 x64 and Windows XP x64.

```
cscript %SYSTEMDRIVE%\inetpub\adminscripts\adsutil.vbs SET
W3SVC/AppPools/Enable32bitAppOnWin64 1
%SYSTEMROOT%\Microsoft.NET\Framework\v2.0.50727\aspnet_regiis.exe -i
```

How To Configure the Web Site for HTTPS

The standard installation assumes a web site address with a HTTP address. If you alter the web site and serve it from a HTTPS address, then follow the steps below

1. The Bindings for the site must be modified to include HTTPS. As HTTPS requires a SSL certificate, you must acquire one from a signing authority or create a Self Signed Certificate. Note: Self Signed Certificate may have a limited lifespan and may not be automatically accepted by the end user browser.
2. For **IIS6 only** update the behaviourConfiguration attribute for the service connections in the LinkOne Webview configuration:
 1. Modify `~/web.config`
 2. Comment all serviceMetadata for IIS7, 8 (HTTP and HTTPS) and IIS6 (HTTP) containing behaviourConfiguration="serviceBehaviour"
 3. Uncomment all serviceMetadata for IIS6 (HTTP) without a behaviourConfiguration attribute
3. Update the service connection bindings in the LinkOne WebView configuration:
 1. Modify `~/web.config`.
 2. Comment serviceMetadata and all endpoints that are preceded with `<!-- Http web site support -->`
 3. Uncomment serviceMetadata and all endpoints that are preceded with `<!-- Https web site support -->`

The following are the only valid combinations of system configuration for HTTPS:

HTTPS configured in IIS?	Browsing with HTTPS address?	HTTPS configured in web.config?
--------------------------	------------------------------	---------------------------------

no	no	no
yes	no	no
yes	no	yes
yes	yes	yes

How To Configure IIS 7

The standard installation assumes IIS 7 is in use. If you are using IIS 8 or later, then follow the steps below.

- Update the staticContent section in web.config by removing mimeMap attributes for .woff and .woff2:
 - Open `~/web.config`
 - Locate the `<staticContent>` section
 - Remove the comments around : `<mimeMap fileExtension=".woff" mimeType="application/woff"/>`
 - Remove the comments around: `<mimeMap fileExtension=".woff2" mimeType="application/woff2"/>`

How To Configure for Multiple Web Sites

The standard installation assumes your web Server is providing a single web site. If it is providing multiple web sites, then follow the steps below to edit the `serviceHostingEnvironment` data.

- If you are using ASPNET 4:
 - Add the line


```
<serviceHostingEnvironment multipleSiteBindingsEnabled="true" />
```
- If you are running ASP.NET 2.0, 3.0 or 3.5:
 - Uncomment the lines


```
<!--baseAddressPrefixFilters>
<add prefix="http://localhost"/>
</baseAddressPrefixFilters-->
```

 and change the prefix as required, e.g.


```
<baseAddressPrefixFilters>
<add prefix="http://LinkOne_Prd"/>
</baseAddressPrefixFilters>
```

How To Configure DBMan Configuration Manager

- Start LinkOne WebView 3 and login with administrator privileges.
- Browse to the **Configuration > Integration**.
- Check the "Enable Configuration Manager" check box.
- In the "Configuration Manager Assembly" box enter **Mincom.LinkOne.DbMan**.

5. In the "Configuration Manager Connection String" box enter **SampleDB**. For DBMan the connection string is a path to where the sample CSV files are located. This path is relative to the **Mincom.Linkone.DBMan** assembly.
6. Click the Save button.
7. DBMan is now ready to be used as a Configuration Manager.

Note: If no connection string is entered, then DBMan will default to using **SampleDB** anyway.

How To Use DBMan Configuration Manager to Open a Book Via Equipment Id

This sample assumes that DBMan has been configured (see How To Configure DBMan Configuration Manager) and that the DBMan database is the default installed database, and the demo library shipped with LinkOne WebView 3 is enabled.

Before undertaking this sample ensure that a user account is available that has access to view the all books from the "komatsu" publisher in the demo library.

To open a book via equipment id, LinkOne WebView use the `SilentTest.aspx` page in the root web site (this allows the user to open by codes or equipment id):

1. Browse to `~/SilentTest.aspx` (e.g `http://localhost/LinkOne/SilentTest.aspx`).
2. Enter the user credentials that have access to the komatsu books.
3. In the **Equipment Id** field enter "dozer".
4. Click the **Login** button.
5. The komatsu / d355-a dozer book will open.
Notice that the TCSN filter has been set to K1000 as this is a filter defined in the DBMan database.
6. Browse again to the `~/SilentTest.aspx` as in step 1.
7. Enter the user credentials that has access to the komatsu books
8. In the **Equipment Id** field enter "dozer1".
9. Click the **Login** button.
10. The komatsu / d355-a dozer book will open.
Notice that the TCSN filter is now set to K1500.
11. On the parts list click the "Engine" link.
12. The komatsu / sa6d155 engine book will open.
This book opens because the "Engine" link has a slot reference to "Engine2" in the DBMan slot table. Notice that the "SN" filter is set to "20500" because this is a temporary filter that is specified in the DBMan database for the "Engine2" Equipment Id.

How To Configure Book Libraries on a Network Share

In order to access a book library stored on a network share, WebView requires read access to that network share. If the book library requires re-indexing, WebView also requires write access.

In some circumstances security restrictions may prevent the default account that WebView runs with from accessing the network share. In this scenario WebView needs to use an identity that is known to both the server running WebView and the server hosting the network share. There are two methods that can be used to achieve this.

The first method simply involves granting access to the Everyone account on the network share. This allows access to the service account used by WebView.

The second method involves running WebView using a different identity. After installing WebView:

1. Decide on an identity that will be used to run WebView. It must be known to both the server running WebView and the server hosting the network share.
 - a. Give the selected identity read and write privileges to the WebView installation folder.
 - b. Give the selected identity read and write privileges to the files in the network share.
 - c. Give the selected identity read and write privileges to the network share itself.
2. Reconfigure the WebView application pool to use the selected identity:
 - a. Open IIS Configuration Manager.
 - b. Identify the application pool being used by WebView (usually LinkOne_AppPool).
 - c. Change the process identity to the selected identity for the identified application pool.
3. Configure the WebView databases to allow access for the selected identity:
 - a. If using Microsoft SQL Express:
 1. If the SQL Express Management Studio has not previously been installed, install it by running the Microsoft Web Platform Installer. Select **Web Platform Database Customize**, select **SQL Express Management Studio**. Click **Install**.
 2. Open SQL Express Management Studio and connect to the SQL Express instance (usually <machine name>\SQLEXPRESS).
 3. Attach the database **~/App_Data/ASPNETDB.mdf** as ASPNETDB
 4. Attach the database **~/App_Data/LINKONEDB.mdf** as LINKONEDB
 5. Create a login to the SQL Express database for the selected identity.
 6. Modify the login User Mappings and map ASPNETDB and LINKONEDB, giving each mapping the dbo role.
 - b. If using Microsoft SQL Server see the "How To Use SQL Server Instead of SQL Express" walkthrough and configure for the selected identity.
4. Update the location of the databases in the WebView configuration:
 1. Modify **~/web.config**.
 2. Uncomment the SQL server connection string for LocalSqlServer (modify the data source to the SQL Server / SQL Express instance name if not the default instance).
 3. Uncomment the SQL server connection string for LinkOneSqlServer (modify the data source to the SQL Server / SQL Express instance name if not the default instance).
 4. Comment out the SQL Express connection string for LocalSqlServer
 5. Comment out the SQL Express connection string for LinkOneSqlServer

How to Configure SharePoint Integration

WebView integrates with SharePoint by requesting documents that are relevant to the current viewing context. Documents may be associated with one of three Link One contexts:

- The Publisher
- A book or set of books
- A page or set of pages

Publisher level documents are retrieved when viewing any content from the specified publisher. Book level documents are retrieved when viewing one of the specified books. Finally, page documents are retrieved when viewing any of the specified pages. Page level documents may also specify the set of books in which case both page and book code must be matched, but this is optional.

Requirements

This section outlines how to set up SharePoint integration for WebView. This integration requires:

- WebView 3.7 or above
- SharePoint 2010

NOTE : This section provides some basic instructions on how to configure SharePoint 2010. These instructions are provided for guidance only. LinkOne support takes no responsibility for resolving SharePoint configuration issues.

ALSO NOTE : Some paths may differ from those specified due to different system configurations. For example, "Program Files" may need to be "Program Files (x86)".

Set Up Process

1. Install WebView 3.7 or higher.
2. Configure a SharePoint list.
3. Configure WebView for SharePoint support.
4. Add and maintain SharePoint documents.

Install and Verify WebView

Follow the standard WebView installation process. In order to integrate with SharePoint, you must use Windows Authentication. You can specify this setting in the web.config file. For example:

```
<authentication mode="Windows" />
```

You may also need to enable impersonation. This allows ASP.NET to execute using the identity of the client on whose behalf they are operating. You usually impersonate for resource access control. You can specify the settings in the web.config file. For example:

```
<identity impersonate="true" userName="accountname" password="password" />
```

Please see the following Microsoft KB article for further details:

<http://support.microsoft.com/kb/306158>

Note: Verify that the system is fully operational before proceeding with any of the following steps.

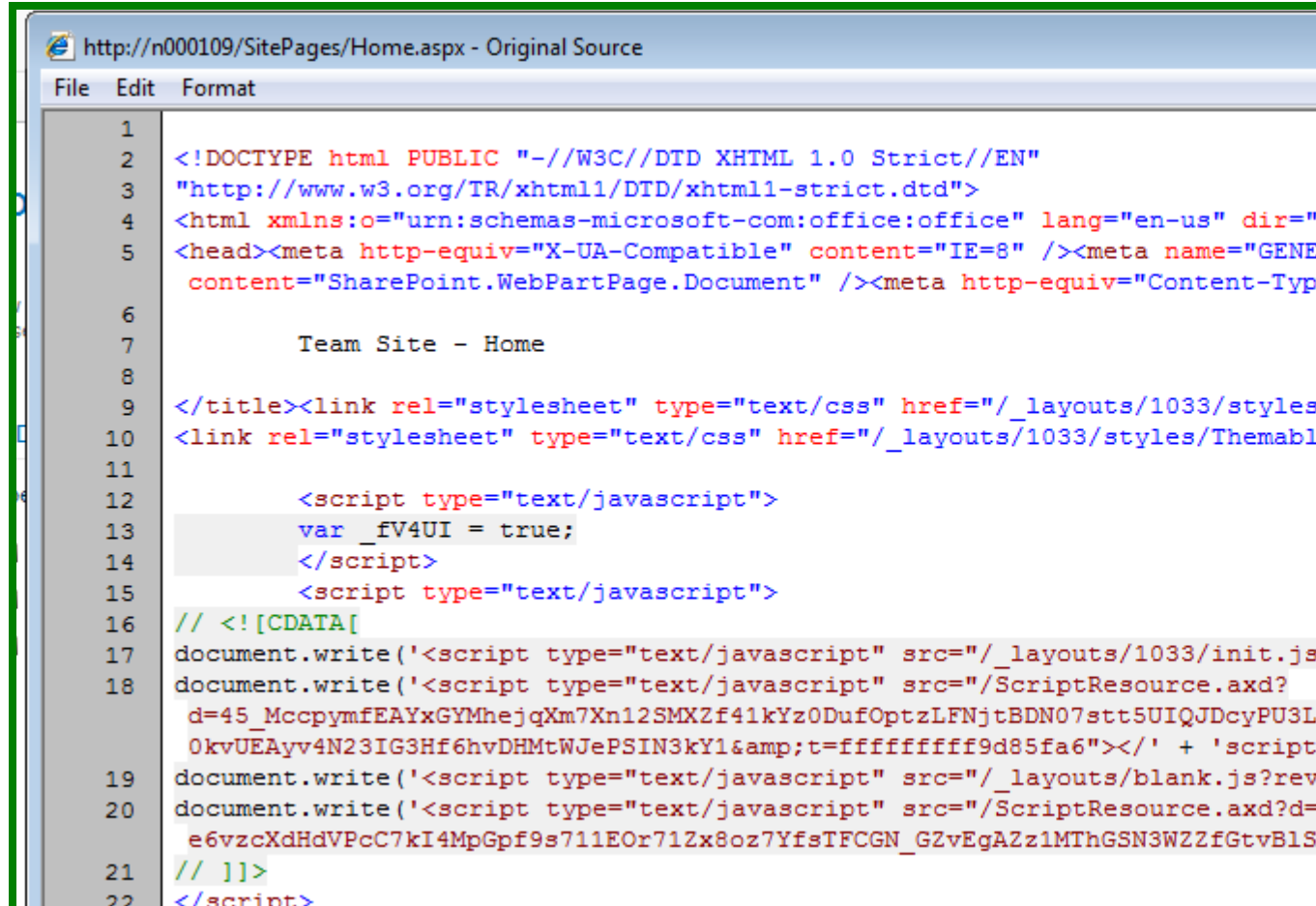
Configure a SharePoint List

WebView retrieves documents for the current context by querying a SharePoint list. This list must have some additional custom columns added in order to identify the context to which a document belongs.

1. Log in as an administrator to the SharePoint site used to store WebView data.
2. Select the SharePoint list of interest.
3. Add the following columns to the SharePoint list. Each column must support text entry. All of these are required.
 - Publisher Code
 - Book Code
 - Page Reference
 - Language
 - Model
4. Ensure that the "Publisher Code" requires a value to be entered.

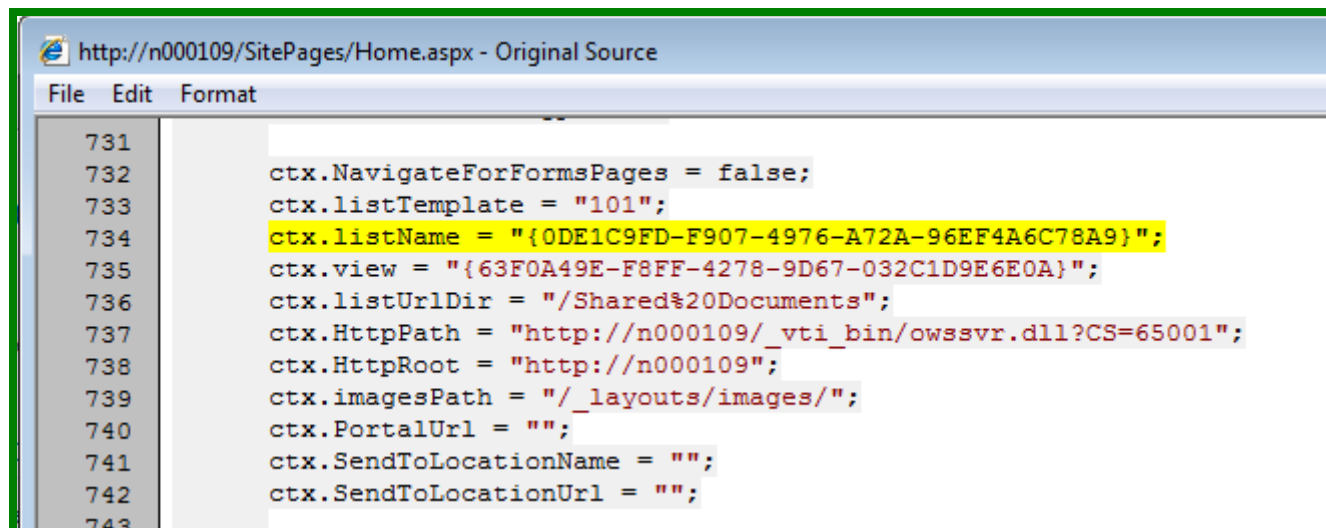
Note: These are the default column names used by WebView.

5. Record the GUID name of the SharePoint list. Instructions for finding the GUID of a SharePoint list are available on the internet. The simple way is to do the following:
 - a. Open Internet Explorer browser and navigate to your SharePoint site.
 - b. Make sure you are on the Shared Documents page.
 - c. Right click on the page, and select "View Source". This opens up the original source of the page in a browser window. It looks something like this:



```
1
2 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
3 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
4 <html xmlns:o="urn:schemas-microsoft-com:office:office" lang="en-us" dir="
5 <head><meta http-equiv="X-UA-Compatible" content="IE=8" /><meta name="GENE
6 content="SharePoint.WebPartPage.Document" /><meta http-equiv="Content-Typ
7
8 Team Site - Home
9
10 </title><link rel="stylesheet" type="text/css" href="/_layouts/1033/styles
11 <link rel="stylesheet" type="text/css" href="/_layouts/1033/styles/Themabl
12
13 <script type="text/javascript">
14 var _fv4UI = true;
15 </script>
16 <script type="text/javascript">
17 // <![CDATA[
18 document.write('<script type="text/javascript" src="/_layouts/1033/init.js
19 document.write('<script type="text/javascript" src="/ScriptResource.axd?
20 d=45_MccpymfEAYxGYMhejqXm7Xn12SMXZf41kYz0DufOptzLFNjtBDN07stt5UIQJDcyPU3L
21 0kvUEAyv4N23IG3Hf6hvDHMTwJJePSIN3kY1&amp;t=ffffffff9d85fa6"></' + 'script
22 document.write('<script type="text/javascript" src="/_layouts/blank.js?rev
23 document.write('<script type="text/javascript" src="/ScriptResource.axd?d=
24 e6vzcXdHdVPcC7ki4MpGpf9s711EOr71Zx8oz7YfsTFCGN_GZvEgAZz1MThGSN3WZZfGtvB1S
25 // ]]>
26 </script>
```

- d. Search in this window for the text "listName". The search should find an item with a GUID assignment. That is the GUID you want to copy for this value.



```
731
732 ctx.NavigateForFormsPages = false;
733 ctx.listTemplate = "101";
734 ctx.listName = "{0DE1C9FD-F907-4976-A72A-96EF4A6C78A9}";
735 ctx.view = "{63F0A49E-F8FF-4278-9D67-032C1D9E6E0A}";
736 ctx.listUrlDir = "/Shared%20Documents";
737 ctx.HttpPath = "http://n000109/_vti_bin/owssvr.dll?CS=65001";
738 ctx.HttpRoot = "http://n000109";
739 ctx.imagesPath = "/_layouts/images/";
740 ctx.PortalUrl = "";
741 ctx.SendToLocationName = "";
742 ctx.SendToLocationUrl = "";
743
```

Configure WebView

The WebView configuration file and the web.config must be modified to enable SharePoint integration. The default configuration file is Mincom.LinkOne.WebView.config

1. Open the WebView configuration file.
2. Add and configure the following settings:

Mincom.LinkOne.WebView.WebServer.Services.Cms.Enabled

A complete list of configuration values for the Mincom.LinkOne.WebView.Config file is described below:

Configuration Key	Description
Mincom.LinkOne.WebView.WebServer.Services.Cms.Enabled	"True" to enable SharePoint integration, "False" to disable. Default value is "False".

1. Open the web.config file.
2. Add and configure the following AppSetting values:
 - Mincom.LinkOne.Adapter.Cms.Assembly
 - Mincom.LinkOne.SharePoint.BaseUrl
 - Mincom.LinkOne.SharePoint.ListTitle

An example of these config values is shown here:

```
<add key="Mincom.LinkOne.Adapter.Cms.Assembly"
      value="CmsProvider.SharePointProvider,Mincom.LinkOne.Sharepoint"/>
<add key="Mincom.LinkOne.SharePoint.BaseUrl"
      value="http://myserver:8080/" />
<add key="Mincom.LinkOne.SharePoint.ListTitle"
      value="{0DE1C9FD-F907-4976-A72A-96EF4A6C78A9}" />
```

A complete list of web.config settings used for SharePoint integration are described below:

AppSetting Key	Description
Mincom.LinkOne.Adapter.Cms.Assembly	Class Name and Namespace of the custom CMS assembly. The value for SharePoint should be: "CmsProviderSharePointProvider,Mincom.LinkOne.WebView.SharePoint"
Mincom.LinkOne.SharePoint.BaseUrl	Identifies the URL used to access the SharePoint site. Include a trailing '/' character. For example http://SharePointHost:8080/
Mincom.LinkOne.SharePoint.ListTitle	The GUID of the SharePoint list to access – e.g., {A611A2FE-4FDC-49BD-9E28-973444BCDB80}

Adding SharePoint Documents and Associating them to LinkOne

Documents can now be added to SharePoint for display by WebView. Documents must be associated with:

- A single publisher code
- Any number of book codes (or none)
- Any number of page references (or none)

These are specified by entering data into the respective fields in the SharePoint list. A document may only be associated with one publisher code (exact match, although case insensitive). Any number of book codes and page references may be assigned to the document. Multiple codes are separated by the pipe '|' character.

Sharepoint Alerts (Notifications)

The LinkOne Book Change Notification service does not notify when documents stored in SharePoint are modified. To be notified or alerted when changes are made to SharePoint documents, it is recommended to configure SharePoint Email Alerts.

The following references are included to assist with this process:

- SharePoint Email Alerts Configuration
<http://technet.microsoft.com/en-us/library/cc263462.aspx>
- Configuring alerts on documents (files)
SharePoint 2010 Help topic "Create an alert for a folder, file or list item"

How to Configure Advanced Search Facility Administration

Setup

Prior to users accessing the Advanced Search Facility, a number of tasks must be completed by the LinkOne WebView System Administrator.

Prerequisite Software

SUN java should be installed on your Web server. We recommend that you download and install the most recent version of Java that is available.

Settings

The `~/web.config` file must be edited to enable a connection to the SOLR Search Index DB. This connection is used by the LinkOneBookIndexingService and LinkOne WebView.

Modify the following line from the Connection Strings section (under the comment '**SQL Server connection strings**') and change `connectionString=.` to point to the appropriate SOLR instance. For example:

```
<add name="LinkOneSearchIndexServer"
connectionString="http://localhost:8983/solr" />
```

Modify the LinkOne.LinkOne.WebView.config Settings

By default, the Advanced Search Facility is not enabled. To enable the Advanced Search Facility, edit the `~/bin/Mincom.LinkOne.WebView.config` file by adding the `LibrarySearchEnabled` key and setting its value to true and enabled to true as below:

```
<add key="Mincom.LinkOne.WebView.WebServer.LibrarySearchEnabled" value="true"
enabled="true" />
```

The Advanced Search Facility can be disabled by either removing the key or by setting the value of LibrarySearchEnabled to false or by setting enabled to false.

By default, the maximum rows returned by a search is limited to 100. This can be changed by adding the **LibrarySearchMaxRows** key to the `~/Mincom.LinkOne.WebView.config` file and setting its value to true and enabled to true as below:

```
<add key="Mincom.LinkOne.WebView.WebServer.LibrarySearchMaxRows" value="100"
enabled="true" />
```

Modify the LinkOneBookIndexingService.exe.config Settings

The LinkOneBookIndexingService configuration settings are as follows:

- The BookIndexingService must point to the LinkOne installation directory by modifying the LinkOneDir key. For example:

```
<add key="Mincom.LinkOne.WebView.LinkOneDir"
value="c:\inetpub\wwwroot\LinkOne" />
```

- The EventDelayMinutes key determines how long (in minutes) it waits before checking for book changes:

```
<add key="Mincom.LinkOne.WebView.EventDelayMinutes" value="5" />
```

- The SolrConnectionTimeout key determines how long (in milliseconds) it waits before timing out on an operation:

```
<add key="Mincom.LinkOne.WebView.SolrConnectionTimeout" value="3600000" />
```

- The VerboseLoggingEnabled key determines if the Service should generate output to the log file:

```
<add key="Mincom.LinkOne.WebView.VerboseLoggingEnabled" value="false" />
```

- The SolrConnectionTimeout key determines what folder the log file will be located:

```
<add key="Mincom.LinkOne.WebView.LogDirectory" value="Data\Log" />
```

- The IndexEmbeddedDocuments key is used to enable indexing of embedded documents:

```
<add key="Mincom.LinkOne.WebView.IndexEmbeddedDocuments" value="false" />
```

- The NumBooksPerSolrCommitDuringInitialSync key is used to determine how many books the Service will process before committing the results into SOLR:

```
<add key="Mincom.LinkOne.WebView.NumBooksPerSolrCommitDuringInitialSync"
value="10" />
```

Installation and configuration of the SOLR Instance

The Advanced Search Facility uses a SOLR server instance to hold indexes to the documents in the LinkOne libraries. This server does not need to be the same exact server as your LinkOne instance. ABB supplies a compressed ZIP file that contains the entire contents required to utilize the SOLR Database and the LinkOne schema.

For LinkOne WebView, we recommend that your SOLR service be hosted inside of a Jetty HTTP Server instance that runs as a Windows Service.

NOTE: Because the SOLR installation is a manual process and is not a part of the automated WebView installer package, the WebView installer will not uninstall the Jetty Service or clean up the files that are manually installed\configured.

This version supports two basic configurations for SOLR:

1. Basic configuration, which will support most deployments, and
2. Replication configuration, which supports larger, more demanding deployments.

The Basic configuration can be setup by either:

- using the prepared deployment package (**SOLR-Basic.zip, found in the ~\Distributions folder**) and deploying it onto your server or

- manually performing the configuration steps individually.

The Replication configuration can only be setup by manually performing the configuration steps individually.

We highly recommend using the prepared deployment package for the Basic configuration instead of manually performing the detailed steps, as those configurations have been fully qualified by our QA department. The manual instruction steps were compiled using SOLR version **3.6.1** and jetty version **6.1.26**. These instructions may not be accurate for other versions.

NOTE: Confirm that Java has been installed on the server before using SOLR. We recommend that you download and install the most recent version of Java that is available. NB: Jetty 6.1.26 only runs on Java 7.

Basic Configuration Setup

To install using the prepared deployment package, perform the following steps:

- Copy the **SOLR-Basic.zip** file to your SOLR server (this file can be found in the **~\Distributions folder**). This server may also be your LinkOne WebView server.
- Unzip the file. Your destination folder can be wherever you want it to be. It is suggested that if you are deploying onto a LinkOne WebView server that your destination folder be a subfolder under your LinkOne WebView folder structure. For example:

If LinkOne WebView is installed to **c:\inetpub\wwwroot\LinkOne**, then we suggest that you deploy the SOLR instance to the following subfolder:
c:\inetpub\wwwroot\LinkOne\SOLR
- Install the SOLR instance as a Windows Service. To do this, open a DOS Command Prompt with Administrative privileges.
- Change directory to the **SOLR\WebView\bin** folder. Type in the following command:


```
Jetty-Service.exe --install jetty-service.conf
```

You should see a success message like the following:

```
wrapper | Jetty6-Service installed.
```
- Close the DOS window.
- You may now run your SOLR instance as a windows service and perform a start/stop/restart anytime you like.
- Once the service has been installed, you will see it in your Services Console with the name **Jetty6-Service**. Start the service and you are now ready to populate the SOLR indexes by configuring the **LinkOneBookIndexingService**.

To manually perform the configuration steps individually, perform the following:

- Create a new directory under **WebView\LinkOne\Main** named "**SOLR**". Although this directory can be created anywhere, we suggest that you create it in this location.
- Download a copy of the SOLR server package v3.6.1 from here: <http://www.apache.org/dyn/closer.cgi/lucene/solr>
- Decompress the SOLR server into the SOLR directory (this will result in a new folder, **apache-solr-3.6.1** being created)
- Copy the folder **SOLR\apache-solr-3.6.1\example** to **SOLR\apache-solr-3.6.1\WebView**.
- Open a DOS Command prompt and navigate to the **SOLR\apache-solr-3.6.1\Webview** directory.
- From the command window, execute the command "**java -jar start.jar**". This should launch the SOLR server.
- Test the SOLR instance by opening a web browser to the address **http://servername:8983/solr/admin/**. Stop the SOLR server by entering **CRTL-C** at the command prompt window.

8. Copy the file `schema.xml` into the folder `SOLR\ apache-solr-3.6.1\Webview\solr\conf`. This will replace an existing file. You can locate the most recent `schema.xml` file in the `\bin` folder of your LinkOne WebView installation.
9. Open the file `SOLR\ apache-solr-3.6.1\Webview\solr\conf\solrconfig.xml` in a text editor. Find the following element:

```
<maxBooleanClauses>1024</maxBooleanClauses>
```

and modify the value to this:

```
<maxBooleanClauses>1048576</maxBooleanClauses>
```

10. Download a copy of **jetty v6.1.26** from <http://dist.codehaus.org/jetty/jetty-6.1.26/>
11. Unzip the jetty package on your server.
12. Copy the folder `jetty-6.1.26\jetty-6.1.26\bin` from the jetty package into a new folder under your SOLR instance named `SOLR\ apache-solr-3.6.1\WebView\bin`.
13. Copy the folder `jetty-6.1.26\lib\win32` from jetty into a new folder called `SOLR\ apache-solr-3.6.1\WebView\lib\win32`.
14. Copy the file `etc\jetty-win32-service.xml` from jetty into the `SOLR\ apache-solr-3.6.1\WebView\etc` folder.
15. Copy the file `SOLR\ apache-solr-3.6.1\WebView\solr\conf\stopwords_en.txt` into the `SOLR\ apache-solr-3.6.1\WebView\bin` folder.
16. Open the file `SOLR\ apache-solr-3.6.1\Webview\bin\jetty-service.conf` in a text editor. Locate the following line:

```
wrapper.java.additional.2=-Djetty.logs=./logs
```

And insert this new line immediately following that entry:

```
wrapper.java.additional.3=-Dsolr.solr.home=./solr
```

17. Create a new folder `SOLR\ apache-solr-3.6.1\Webview\lib`
18. Copy the file `SOLR\ apache-solr-3.6.1\dist\apache-solr-cell.jar` into the new lib folder.
19. Copy all of the files in the folder `SOLR\ apache-solr-3.6.1\contrib\extraction\lib` into the new lib folder.

20. Optional Steps:

1. If you want to disable logging, put a "#" sign before the entry

```
wrapper.logfine=./logs/jetty-service.log.
```

You can also modify the logging by using the SOLR administrative console (<http://localhost:8983/solr/admin/> on the server).

2. Assign more memory available to the Jetty service that is running under Java by changing the setting `wrapper.java.maxmemory` from **64** to **512**.

21. To install the Jetty Service as a Windows Service, open a DOS Command Prompt with administrative privileges and navigate to the `WebView\bin` folder.

22. Run the command:

```
Jetty-service.exe -i jetty-service.conf.
```

You should see a success message similar to "**wrapper | Jetty6-Service installed.**"

23. You may now start\stop the Jetty Service from the Services Console.
24. Test the SOLR instance again by opening a web browser to the address <http://servername:8983/solr/admin/>.

Replication Configuration Setup

To manually configure the Replication installation, perform the following steps:

1. Follow steps **1 – 20** above under the prior section "**To manually configure the Basic installation, perform the following steps**".

- Copy the entire **SOLR\ apache-solr-3.6.1\Webview** folder under **SOLR** to a new folder named **SOLR\ apache-solr-3.6.1\Webview_Slave**
- Add the following request handler to the **WebView\solr\conf\solrconfig.xml** file:

```
<requestHandler name="/replication" class="solr.ReplicationHandler">
  <lst name="master">
    <str name="replicateAfter">commit</str>
    <str name="replicateAfter">startup</str>
    <str name="confFiles">schema.xml, stopwords.txt</str>
  </lst>
</requestHandler>
```

- Edit the **Webview_Slave\etc\jetty.xml** and change the port number to **8984** per this example:

```
<Call name="addConnector">
  <Arg>
    <New class="org.mortbay.jetty.bio.SocketConnector">
      <Set name="host"><SystemProperty name="jetty.host" /></Set>
      <Set name="port"><SystemProperty name="jetty.port"
default="8984" /></Set>
      <Set name="maxIdleTime">50000</Set>
      <Set name="lowResourceMaxIdleTime">1500</Set>
      <Set name="statsOn">>false</Set>
      <Set name="headerBufferSize">1048576</Set>
    </New>
  </Arg>
</Call>
```

- Add a request handler to the **WebView_Slave\solr\conf\solrconfig.xml** file, making sure to use the new port **8984**:

```
<requestHandler name="/replication" class="solr.ReplicationHandler" >
  <lst name="slave">
    <str name="masterUrl">>
    <str name="pollInterval">00:00:60</str>
  </lst>
</requestHandler>
```

- Edit the file **Webview_Slave\bin\jetty-service.conf**. Locate the lines in the file for the **wrapper.ntservice.name** and the **wrapper.ntservice.displayname** properties. Modify them to be the following:

```
wrapper.ntservice.name=JettySlave
wrapper.ntservice.displayname=Jetty6-Service-Slave
```

- Install the SOLR Master instance and the SOLR Slave instance as Windows Services. To do this, open a DOS Command Prompt with Administrative privileges. Change directory to the **SOLR\ apache-solr-3.6.1\Webview\bin** folder. Type in the following command:

```
Jetty-Service.exe --install jetty-service.conf
```

You should see a success message like the following:

```
wrapper | Jetty6-Service installed.
```

- Change directory to the **SOLR\ apache-solr-3.6.1\Webview_Slave\bin** folder. Type in the following command:

```
Jetty-Service.exe --install jetty-service.conf
```

You should see a success message like the following:

```
wrapper | Jetty6-ServiceSlave installed.
```

- Close the DOS window.
- You may now run your SOLR Master and Slave instance as Windows services and perform a start/stop/restart anytime you like.
- Once the services have been installed, you will see them in your Services Console with the names **Jetty6-Service** and **Jetty6-ServiceSlave**. Start the **Jetty6-Service**. Next start the **Jetty6-ServiceSlave** service. Verify that the services are running.
- Test the SOLR instance again by opening a web browser to the address **http://servername:8983/solr/admin/**.

13. You are now prepared to populate the SOLR indexes by configuring and initiating the **LinkOneBookIndexingService**.

Operation

Indexing the LinkOne Books

Before the Advanced Search Facility component of the WebView application can be used, the LinkOne libraries must be indexed into the SOLR repository. This is done by the System Administrator installing and then starting the Service. Note that the **LinkOneBookIndexingService** will index all of the LinkOne libraries specified in the **Services.BookService.Libraries** key found in the `~/Mincom.LinkOne.WebView.config` file.

To install the service, you may run the batch file `~/bin/InstallLinkOneBookIndexingService.bat` with Administrative privileges or open a DOS Command Prompt with Administrative privileges and change directory to the `~/bin` folder. Enter in the following DOS command (note that the `installutil` executable must be in your Windows path for this to work):

```
installutil LinkOneBookIndexingService.exe
```

To Start the service, open a DOS Command prompt and change directory to the `~/bin` folder. Enter in the following DOS command:

```
net start LinkOneBookIndexingService
```

To Stop the service, open a DOS Command prompt and change directory to the `~/bin` folder. Enter in the following DOS command:

```
net stop LinkOneBookIndexingService
```

To remove the service, you may run the batch file `~/bin/UninstallLinkOneBookIndexingService.bat` or open a DOS Command prompt and change directory to the `~/bin` folder. Enter in the following DOS command (note that the `installutil` executable must be in your Windows path for this to work):

```
installutil /u LinkOneBookIndexingService.exe
```

Removal\Uninstallation of the SOLR Instance

The WebView installer package will not uninstall or remove any of the files or services configured in the manual steps documented above. In order to remove these, perform the following steps prior to running the Uninstall of WebView:

1. Stop the **LinkOneBookIndexing Service** from the Services console.
2. Remove the **LinkOneBookIndexingService** as documented above.
3. Stop the **Jetty Service** from the Services console.
4. Open a command prompt and change directory to the `SOLR\WebView\bin` folder. Type in the following command to remove the **Jetty Service**:

```
Jetty-Service.exe -r
```

You should see a success message like the following:

```
wrapper | Jetty6-Service removed.
```

5. As a final step, you should delete the SOLR directory structure from the hard drive.

Using the Advanced Search Status Page

The administrator may use the LinkOne Book Indexing Service page to review the overall Book Indexing status information. This page displays the status of the Windows Service (i.e. running or stopped). It also reports how many books still need to be indexed as well as how many books have completed the indexing process. If there are any books that generated an error during indexing, then this page displays a table that includes: The Book, Library, Exception Message, and Status of all failed books. The list may be sorted and filtered. When there are failures listed, the administrator may issue a "Retry" request or a "Retry Ignoring Errors" request.

The administrator is also given the ability to Reset the LinkOne Book Indexing Service Windows Service:

NOTE: Resetting the LinkOneBookIndexing Service will permanently destroy all of the current Book Index data that has previously been created by the Service. Performing this operation is not reversible and should only be done when required.

To Reset the indexer, you may perform the following steps:

1. Login to the WebView application using the administrator account and navigate to **Configuration -> Advanced Search Status**.
2. Verify that the Indexing Service status is **"Running"**. There will be a green **"Check Mark"** icon located next to the Indexing Service status label. If instead, there is a yellow **"Caution"** icon displayed next to the label, then you will need to **manually restart** the LinkOneBookIndexingService as outlined above.
3. Click on the **"Reset Book Indexing"** link, and when prompted to continue select the **"OK"** button to allow the Book Indexing Reset to proceed.
4. A few minutes may elapse before any indication of the reset action is noticed.

Overriding Publisher defined Parts List Formatting

With the new user interface introduced with WebView 3.20, a number of changes to the user interface "look and feel" were incorporated. These changes may render the parts list formatting of some book pages to appear sub-optimal. When this occurs, LinkOne WebView provides a mechanism to be used by the publisher/administrator to easily override the original/default formatting of the books, on a per publisher library/folder basis. This mechanism can only be applied when the book is fully expanded state, not stored as a single file book, also known as sfb file.

In order to do so, the publisher/administrator must include a CSS file, that will be treated as an addendum to the default CSS. Two simple examples are included in WebView samples folder (~/Samples/Overriding Formats) to be readily applied to any publishers, but a deeper knowledge in CSS will be required to do more complex modifications.

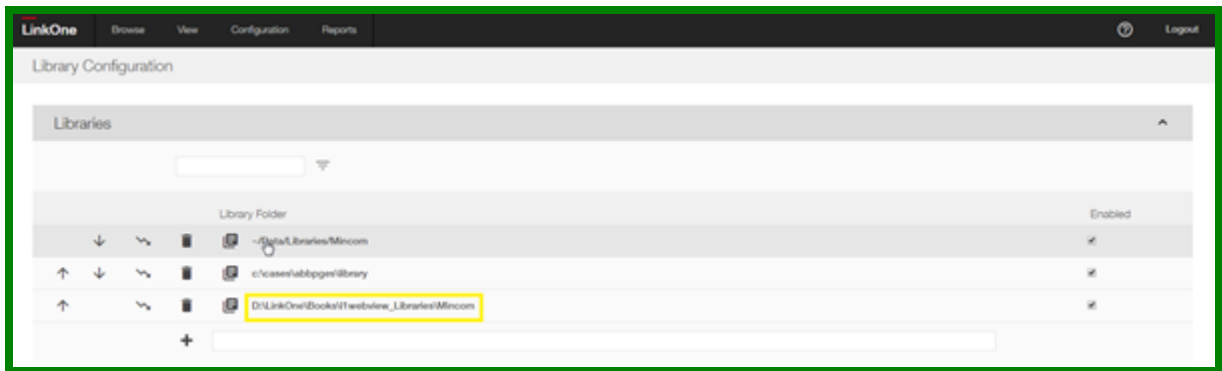
The sample CSS files are as follow:

- stylesheet.01.css contains an example to vertically aligned the parts list items to the middle of the line.
- stylesheet.02.css contains an example to vertically aligned the parts list items to the middle of the line, changing parts list header background color to white and bottom border of the parts list header to grey.

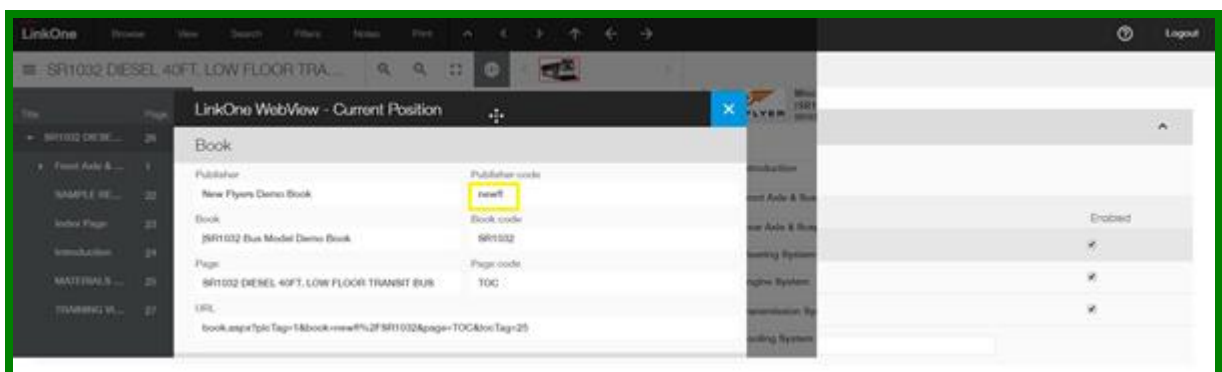
Here are the steps on how to apply the CSS

1. Determine the location of the library on the filesystem. To do this, you can go to Configuration > Libraries to check all the current registered libraries.

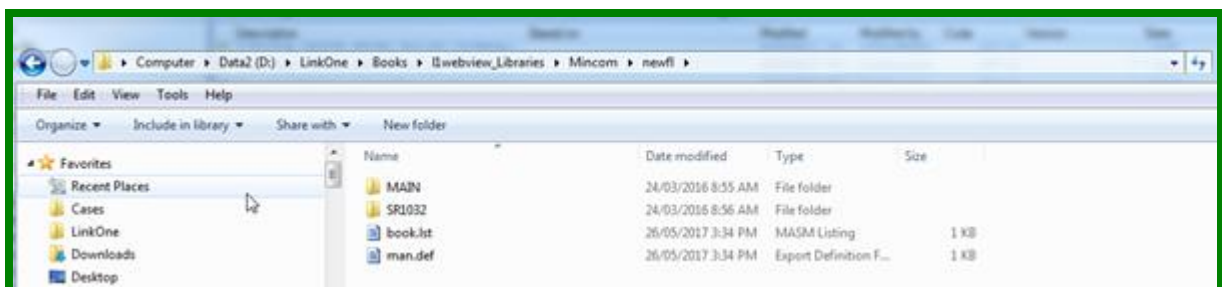
- The following example will use the library highlighted below:



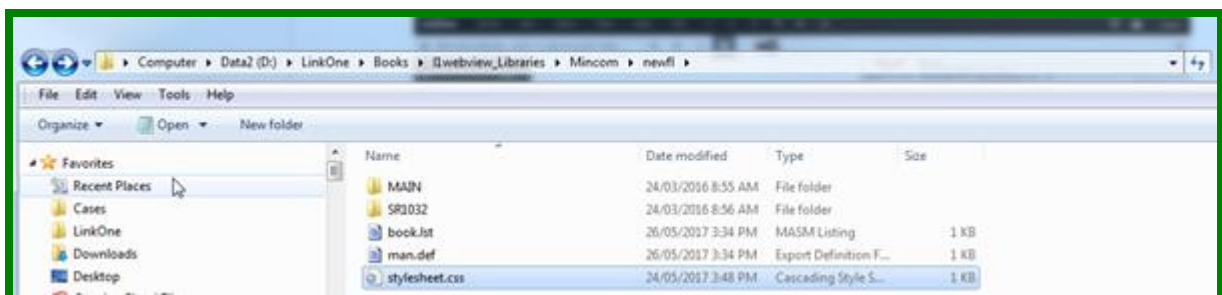
- Open the filesystem location of the library.
- To check if it's the correct publisher, you can match the publisher code against your target book, by opening the book in the browser, and open the View->Current Position option:



- In this example, the publisher code is newfl.
- Open the publisher (newfl) folder:

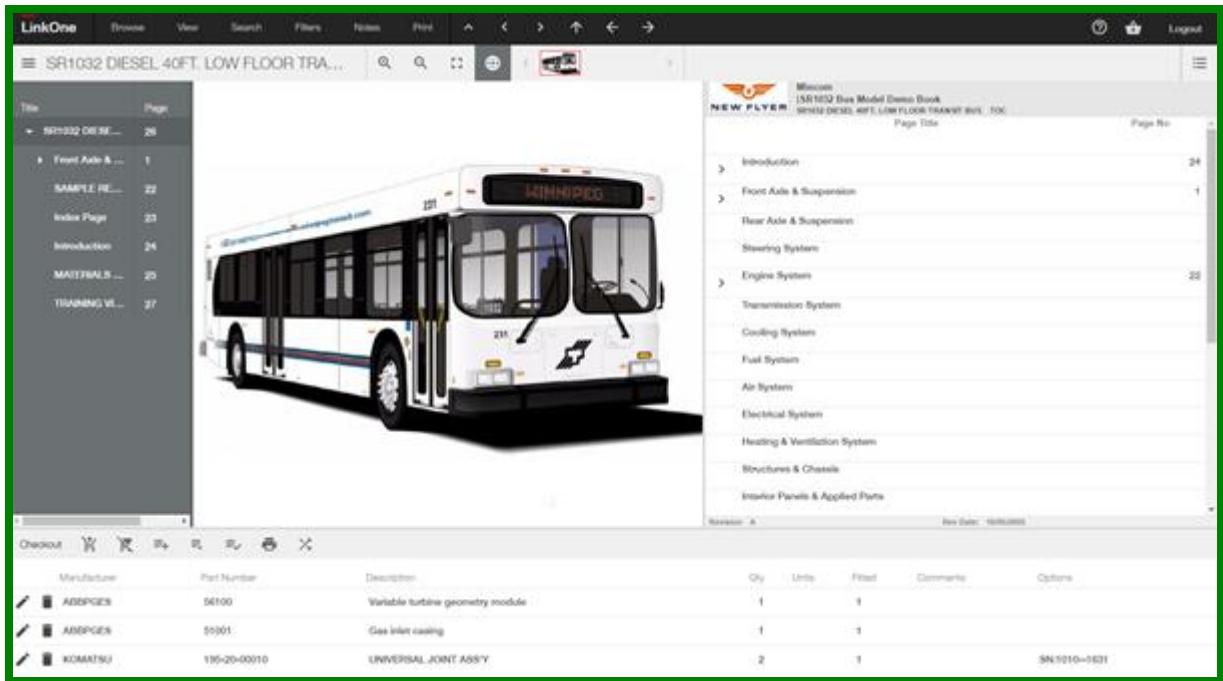


- Choose one of the CSS files to be applied. This example uses stylesheet.02.css. Rename it to stylesheet.css and copy it to the publisher folder.

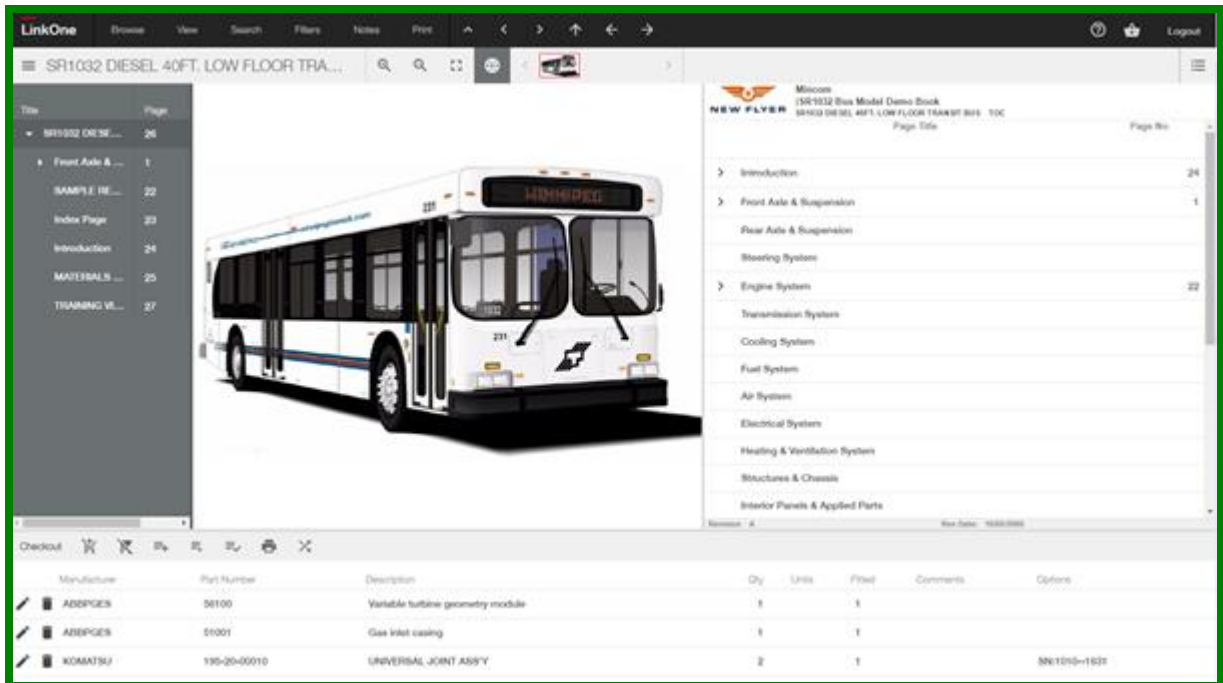


- For the change to take effect, the IIS application pool will need to be restarted.
- Now, reload the page in the browser and review the change

10. Original (default style)



11. Custom



Note the background color of the parts list header and the vertical alignment of the parts list item have changed.

The CSS is applied per publisher's folder basis, not to all books belonging to that publisher. This means that if the publisher is introduced in two different libraries, each of the publisher's folders in those libraries need to have their own copy of the custom CSS.

If the publisher has the same books with the same name in different folder, the order of the books being listed in the libraries will be in effect.