



CAB Local Information System Technical Overview



**Constructive
IT Advice**

Andy & Margaret Henderson
Constructive IT Advice

Preface

This document provides a technical overview of the Local Information System originally developed for Epsom & Ewell Citizens Advice Bureau.

As such, the document expects readers to have some technical expertise.

The document is aimed at people that:

1. Install the software on new machines
2. Install new versions of the software
3. Set up access controls
4. Back up data
5. Make changes to the equipment and software infrastructure in the office
6. Want to explore problems with the system
7. Want to investigate ways to enhance the software.

It provides additional information to that provided in the local information system installation guide.

Table of contents

1	Overview	3
1.1	Components and locations	3
1.2	Technology.....	9
2	Access control	10
3	Configuring the system	13
3.1	Reader program	13
3.2	Maintenance program	15
4	Maintenance tasks	17
4.1	Desktop version	17
4.2	Intranet version.....	18

1 Overview

1.1 Components and locations

The CAB Local Information System has the following main components:

- The desktop 'reader' program
- The maintenance program
- The database
- An intranet version that is accessed by a web browser
- 'Synchronisable' databases
- Supporting programs

The following sections provide more information on each one.

Note that most desktop objects listed here are held in the system's program directory. By default this is C:\Program Files\CITA\CABLocalInformation. However, some objects – such as the configuration files – are copied to the user's application data directory so that changes to those objects do not cause data access problems with Windows access control. The location of the application data directory (and which one the system uses) depends on the operating system you use and how it is set up. To find out the directories the reader or maintenance program is using, double-click the version number shown at the bottom left of the window.

Desktop 'reader' program

The reader program allows advisers to access the local information system. They can use search terms or alphabetic look-ups to find entries in the system. Having found a relevant entry, they can see the available information and click links to relevant web sites and documents to see additional information without leaving the reader program.

The program comprises the following objects:

- Program objects:
 - CABLocalInformation.exe - the program
 - AxInterop.SHDocVw.dll - ActiveX encapsulation of...
 - Interop.SHDocVw.dll - Encapsulates Internet Explorer
 - Microsoft.mshtml.dll - HTML document processing
 - CAB.ico - CAB icon
- CABLocalInformation.chm - help file
- LocalInformation.xml - configuration file
- Objects used to show matching items:
 - twistiedown.gif
 - twistieup.gif
 - LocalInformation.css
 - Popular.gif – highlights popular items
 - Obscure.gif – indicates obscure items
 - MatchingItems.html (recreated by the program)

- Objects used to show CITA credentials:
 - CITA.gif
 - CITACredentials.html (recreated by the program)

Maintenance program

The maintenance program allows authorised administrators to add, change and delete entries in the local information system. The program also provides utilities to:

- Check links from the database
- Repair and compact the database.

The program comprises the following objects:

- Program objects:
 - CABLIMaintenance.exe - the program
 - Interop.ADOODB.dll - encapsulates Microsoft data access software
 - Interop.JRO.dll - encapsulates the Microsoft jet and recovery objects
 - VbPowerPack.dll - .NET supplementary software
 - CAB.ico - CAB icon
- CABLIMaintenance.chm - help file
- LIMaintenance.xml - configuration file

Database

The database holding the local information for the desktop programs is an Access database called LocalInformation.mdb and is stored by the system's central PC. See the accompanying Installation Guide for more information on the role of the central PC in the system.

This directory also has several subdirectories:

- systemLogs - Reader program usage logs
- systemInstall - the PC install program
- systemObjects - used to update C:\Program Files\CITA\CABLocalInformation as necessary
- systemDocumentation – contains documentation for the system.

An intranet version

The intranet version of the Local Information System allows people to work with the database using the same features as the desktop version, but using a web browser.

The system is installed either on the internet or in an application folder referenced by Windows' built-in web server: IIS. The Intranet Installation Guide explains how to decide the location of the application folder. The folder contains:

- **Pages you can invoke directly**
 - LIS.aspx – the main search page
 - ValidateUser.aspx – the system's log on page normally requested via a button on the main search page but can be requested directly

- ChangePassword.aspx – allows someone that has logged on to change their password
- MaintainUsers.aspx – page used to add, change or delete system users and to assign users to groups
- MaintainGroups.aspx – page used to add, change or delete user groups; can also list users and database entries related to the group
- MaintainAttachments.aspx – page used to add, change, overwrite and delete documents in the document library
- GetAttachment.aspx – used to retrieve documents from the document library
- Help.aspx, Help2.aspx, Help3.aspx and Help4.aspx – pages that display help information
- DatabaseUpdater.aspx – used to bring the database up-to-date, to change its status or rebuild search data
- ShowConfigurationValues.aspx – a diagnostic tool that displays the current configuration details used by the system
- TestFileReferences.aspx – a diagnostic tool that tests the way the system is configured and provides diagnostics of any problems it finds (should be deleted once the system is set up)
- **Pages that are invoked by the system**, they are not intended to be called directly:
 - app_offlinex.htm – renaming this file to app_offline.htm takes the system offline
 - ConfirmUpdate.aspx – displays the confirmation page following an update to an item in the database
 - EditAttachment.aspx – Ajax processor used to update documents in the document library by MaintainAttachments.aspx
 - EditGroup.aspx – Ajax processor used to update groups by MaintainGroups.aspx
 - EditItem.aspx – page used to add, change or delete items in the database
 - EditUser.aspx – Ajax processor used to update users by MaintainUsers.aspx
 - GetAttachmentList.aspx – Ajax processor provides document details to MaintainDocuments.aspx
 - GetGroupList.aspx – Ajax processor provides group details to MaintainGroups.aspx
 - GetUserList.aspx – Ajax processor provides user details to MaintainUsers.aspx
 - Global.asax – supports code that runs against all web requests
 - InternalError.aspx – reports any errors detected by the system
 - LISService.asmx – a set of web services provided to support the intranet tools
 - SearchResults.aspx – generates a CSV file containing details of a search result
- **Configuration files:**
 - LocalInformation.config – optional configuration file
 - Web.config – the standard .NET configuration file
- **Other system objects:**
 - bin folder – contains class libraries that support the system
 - Images folder – contains images used by the system
 - Scripts folder – contains JavaScript, including the JQuery library, used by the system; the images subfolder contains images used by JQuery
 - Styles folder – contains CSS style sheets used by the system
 - CKEditor folder – code to support editing the text of items in the database

- **Data folders:**
 - LogDirectory – can be used to store logs of system searches in which case there is one comma-separated values (csv) file per month
 - Database folder – can be used to store a local copy of the Access database

‘Synchronisable’ databases

We provide tools that use the LISService.asmx web services to synchronise a local copy of the database with the one held by the intranet version of the system. The target database cannot be exactly the same as the standard one, because the standard database includes mechanisms to generate database ids. The target needs to be able to define ids explicitly so they match the local version.

We provide two databases in the ‘Intranet\Synchronisable databases’ folder on the distribution media:

- LocalInformation.mdb – is an Access database
- SQLServer.bak – is a backup copy of a SQL Server database

Both databases are empty apart from a single LIUser record defining id TEST and password TEST. You can use this id to populate the database for the first time with the synchronisation utility.

We also provide a script called CreateLISDatabaseScript.sql that you can use to build a new SQL Server database for the Local Information System that includes the initial TEST user.

Supporting programs

The system comes with several programs that help you install and manage the local information system:

- Network install program
- Local information updater
- Local install program
- User id and maintenance program
- Database synchronisation tool
- Intranet database backup tool
- Check intranet web links tool
- Browser fix tool

The following sections describe each one in turn.

Network install program

This program helps you set up the central PC. It sets up the system database and directories ready for distributing files and programs down to PCs that need to use the system.

The program comprises the following objects:

- CABNetworkInstall.exe – the program
- CABNetworkInstall.chm – help file

Local information updater

This program helps you update the configuration on the Server PC in such a way that changes are automatically distributed to the PCs that run the local information system.

The program comprises the following objects:

- CABLIUpdater.exe – the program
- CABLIUpdater.chm – help file

Local install program

This program runs whenever you run the Reader program or the Maintenance program. It checks for updates and, if necessary, implements them before running your chosen program.

It accepts a single command line parameter:

- “L” runs the Reader program
- “M” runs the Maintenance program
- A blank parameter forces the program to update the programs and reset the program icons.

The program comprises the following objects:

- CABLocalInstall.exe – the program
- wshom.ocx – encapsulates the Windows scripting host object module
- ICSharpCode.SharpZipLib.dll – supports unzipping of zip files.

User id and maintenance program

This is a simple utility to maintain the list of people allowed to update the Local Information database.

The program comprises the following objects:

- CABLIAccess.exe – the program
- VbPowerPack.dll – .NET supplementary software.

Database synchronisation tool

This is tool compares two versions of the Local Information database:

- A local Access database kept up-to-date by the maintenance program
- A target database held by the intranet version

It looks for:

- Entries with a modification date in the local version that is later than the one in the target database
- Entries in the local version that don't exist in the target
- Entries in the target version that don't exist in the local version

The tool makes changes in the target version to bring it into line with the local one. It's unlikely to happen, but there's a possibility that 'modified dates' will get out of step. We therefore provide an option to force changes regardless of modified dates.

The program comprises the following objects:

- SynchLIData.exe – the program
- VbPowerPack.dll – .NET supplementary software.

There's more information in the intranet installation guide.

Intranet database backup tool

This tool extracts data from an intranet version of the Local Information System and makes a backup copy in an Access database.

The program comprises just one object: BackupLIData.exe

There's more information in the intranet installation guide.

Check intranet web links tool

This tool extracts links to the web from items held in an intranet database and tests to see if the linked documents are still available.

The program comprises the following objects:

- CheckLIWebLinks.exe – the program
- VbPowerPack.dll – .NET supplementary software.

There's more information in the intranet installation guide.

Browser fix tool

By default encapsulated copies of Internet Explorer (as used by the Desktop 'Reader' program to display search results and internet pages) emulate IE7 which is now dated. Changing this behaviour requires an update to the registry to match the version of Internet Explorer installed on a PC. This tool provides options to do that – it needs to be run using Administrator privileges.

The tool can also override the user agent used by the link checking tool and the link checking routine of the maintenance program.

The program comprises just one object: CABLISBrowserFix.exe

1.2 Technology

The prime development tools are vb.NET (a derivative of Visual Basic) and C#.

The database is Access '97 although any version of Access could be used (see on). We also provide tools to allow the intranet version to use Microsoft SQL Server (see on).

Help text is developed using the Microsoft HTML Help Workshop.

The desktop programs run on any of these operating systems:

- Windows 10 (preview version tested at the time of writing)
- Windows 8 (all versions)
- Windows 7 (all versions)
- Windows Vista (all versions)
- Windows XP Professional
- Windows XP Home Edition
- Windows 2000
- Windows NT 4.0 (Service pack 6a or later)
- Windows ME
- Windows 98

They also require:

- Internet Explorer 5.01 or later
- .NET framework version 1.1. Note that later versions **do not** supersede version 1.1 and cannot be used instead. There is no problem installing version 1.1 alongside any other version of the framework.
- Microsoft data access software.

We recommend installing 'Microsoft Access Database Engine 2010 Redistributable' – a free Microsoft download – as the data access software. If you have installed a Microsoft Office product, you might already have this installed. For backwards compatibility, the programs can also use Microsoft Data Access Components (MDAC) 2.7 (note that version 2.8 will not work, it removes the data access software we need).

The intranet version runs on any PC capable of running IIS 5.1 or later and .NET 2 (also installed by .NET 3 and .NET 3.5). It runs under Windows desktop and Windows server operating systems. To work with an Access database, the intranet can use MDAC, but we recommend using 'Microsoft Access Database Engine 2010 Redistributable' instead. Neither provider is necessary for the intranet version to use SQL Server.

.NET 1.1 and Microsoft Access Database Engine 2010 are available from www.microsoft.com/downloads. We also link to them from www.SeelTA.com/CAB/Downloads.shtml

2 Access control

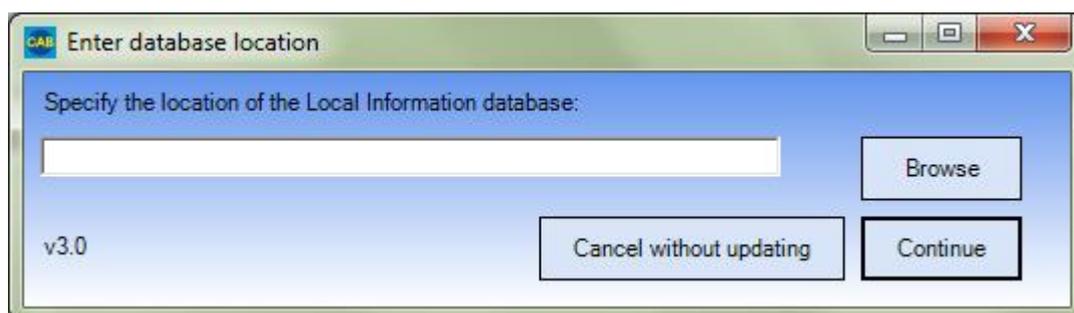
The system includes a simple access control system that checks authorised access to system components that make changes to the local information system database.

The intranet version includes a page that allows maintenance users to add, delete or change ids and passwords held by the system. This section describes the tool provided for desktop use.

You can add, delete or change authorised users by running program CABLIAccess.exe which you can find in the system's program directory on any PC set up to run the system. To use CABLIAccess, you must already be an authorised user. The initial database we supply contains a single user: TEST that uses password: TEST. You should delete this Id as soon as you have created one of your own.

The access control system will automatically log someone on if their Windows user id exists in the local information system database – regardless of what password is specified. The system assumes that if the user has already authenticated with Windows, there is no need to re-authenticate them.

When you start CABLIAccess you will see this screen:



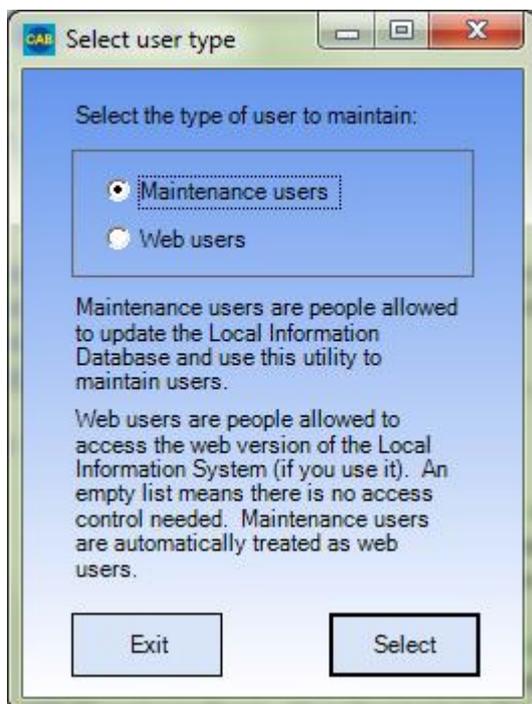
The program will normally show the correct database address automatically.

Click Continue to see this screen:



The screenshot shows a dialog box titled "Enter user id and password". The main text reads: "Enter a user id and password to allow updates to the local information database:". Below this, there are two input fields: "User Id:" and "Password:". Both fields are empty and have a red exclamation mark icon to their right, indicating an error. At the bottom of the dialog, there are two buttons: "Cancel without updating" and "OK".

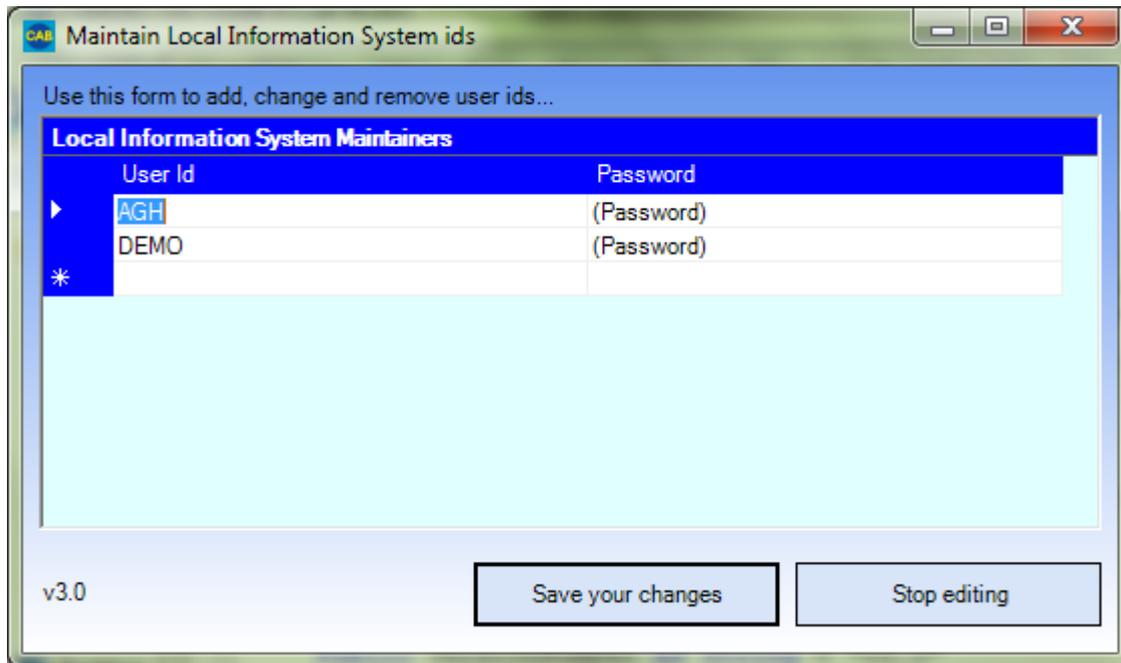
Enter a valid maintenance id and password and click OK to see this screen:



The screenshot shows a dialog box titled "Select user type". The main text reads: "Select the type of user to maintain:". Below this, there are two radio button options: "Maintenance users" (which is selected) and "Web users". Below the options, there is explanatory text: "Maintenance users are people allowed to update the Local Information Database and use this utility to maintain users." and "Web users are people allowed to access the web version of the Local Information System (if you use it). An empty list means there is no access control needed. Maintenance users are automatically treated as web users." At the bottom of the dialog, there are two buttons: "Exit" and "Select".

The utility maintains two types of users as described by the form.

Select which you want to maintain and click Select to see this screen (the heading changes depending on which user type you selected):



To add a new user, insert their id and password in the row starting with an asterisk.

To delete an existing user, click in the blue column to the left of their id and press your delete key.

To change a user's id, simply overwrite it.

To change a user's password, click in the password column next to their id and type the password. Note that the system does not show you their existing password.

User ids are not case-sensitive and are automatically converted to upper case for you. Passwords are, however, case sensitive.

When you have made all the changes you need, save them by clicking the 'Save your changes' button.

When updating maintenance users, we suggest you test your changes before clicking the Stop editing' button.

There is no facility for updating user groups or for assigning web users to user groups. Instead you are expected to use the intranet software to maintain user ids if you also want to use user groups (since user groups are used only by the intranet version). You will see a warning if you try to update web users and groups are present in your database.

3 Configuring the system

This section describes how to configure the desktop programs. The Intranet Installation Guide explains how to configure the intranet version.

The local information system includes mechanisms to allow you to tailor the way it works without changing the program.

3.1 Reader program

There are two ways to tailor the reader program:

- The configuration file
- The style sheet

3.1.1 The configuration file

The configuration file is an XML file called LocalInformation.xml and is stored in the program directory C:\Program Files\CITA\CABLocalInformation.

It is created initially by the Network install program. It is updated by the Local information updater.

The configuration file contains the following tags:

databaseName

The name of the database file. Controlled by the installation programs.

databaseDirectory

The directory (including the server's resource name) containing the database. Should end with "\". Controlled by the installation programs.

logDirectory

The directory (including the server's resource name - assuming the logs are to be held on the server) containing log files generated by the program. Should not end with "\". Controlled by the installation programs.

StopNewWindow

If present, activates the reader program's pop-up blocker. Controlled through the reader program.

searchPrefix

If this tag is not present, the Internet search button does not appear in the local information reader program. If present it provides the search URL up to the point where the search text appears. Controlled by the installation programs.

searchSuffix

If present, the supplied text is added after the search text when creating a search URL. Controlled by the installation programs.

BureauName

If present, includes the provided name in the local information system in place of Epsom & Ewell. This parameter is intended to be used by bureaux adopting the program. A trailing “-” suppresses the display of “Citizens Advice Bureau” following the bureau name. Controlled by the installation programs.

hideDocReferences

If present, prevents the reader program from showing the full address next to a document link. Controlled by the installation programs.

documentURLTextToReplace

If present the system automatically searches for the specified text in document references (case is not significant) and replaces it with...

documentURLReplacingText

Text used when replacing text in document references.

zoomLevel

The percentage of the standard display size that web pages should be zoomed by. 150 increases the zoom level to 150%.

softButtonA and softButtonB

These tags describe soft buttons displayed by the reader program. The file can include one, both or none of the tags. A softButtonB tag does not need a softButtonA tag. Each button definition must have all the following children:

- **caption** - The caption that is shown on the button
- **tooltip** - The tooltip that will be shown when someone hovers their mouse over the button
- **action** - The action that will be taken if the button is clicked. One of “Index lookup”, “Web lookup” or “Document lookup”
- **parameter** - The parameter associated with the button action: either one or more search terms, a web address or a document address
- **numberOfDaysForNew** - the number of days the reader program should look back to decide whether to flash “Recent update”. “0” suppresses the alert. Used only if the action is “Index lookup”.

Controlled by the installation programs.

Other tags

The following are all used by the reader program to 'remember' window and pane positions from one use to the next:

- formTop - position in pixels of the top of the window
- formLeft - position in pixels of the left-hand edge of the window
- formHeight - height in pixels of the window
- formWidth - width in pixels of the window (or "MAX" if maximised)
- formSplitterLeft - width in pixels of the window's search pane.

3.1.2 The style sheet

LocalInformation.css is a standard Cascading Style Sheet (CSS) file that determines how results appear when displayed by the reader program. For example, you could use the CSS file to change fonts, colours and default sizes of the text.

3.2 Maintenance program

The maintenance program's configuration file is an XML file is called LIMaintenance.xml stored in the program directory C:\Program Files\CITA\CABLocalInformation.

It is created initially by the Network install program. It is updated by the Local information updater.

The configuration file contains the following tags:

databaseName

The name of the database file. Controlled by the installation programs.

databaseDirectory

The directory (including the server's resource name) containing the database. Should end with "\". Controlled by the installation programs.

engineType

This determines what version of Access is used when repairing and compacting the local information database as follows:

- 4 = Access '97
- 5 = Access 2000/Access 2003

stopList

Lists all the words that should be dropped when the maintenance program constructs search terms for an entry in the database. Note that one-letter words and all punctuation are automatically removed and do not need to be included in the list. Each entry in the list is contained in a 'stop' tag embedded in the 'stopList' tag.

checkLinkTimeout

The number of milliseconds to wait before deciding that a site is not responding when checking web links. The default is 20000 - i.e. 20 seconds.

Other tags

The following are all used by the maintenance program to 'remember' window and pane positions from one use to the next:

- formTop - position in pixels of the top of the window
- formLeft - position in pixels of the left-hand edge of the window
- formHeight - height in pixels of the window
- formWidth - width in pixels of the window (or "MAX" if maximised)
- formSplitterLeft - width in pixels of the window's left pane
- formSplitterRight - width in pixels of the window's right pane

4 Maintenance tasks

The local information system requires little maintenance besides keeping its data complete and up-to-date. This section describes those few tasks that are necessary. It describes tasks needed for the desktop version and for the intranet version.

4.1 Desktop version

4.1.1 Backup

The local information database needs to be backed-up regularly as do the usage logs in the systemLogs subdirectory.

The other contents and subdirectories of the Server PC database directory need be backed-up only if something in them changes. There is no harm, however, in backing them up more frequently.

The contents of C:\Program Files\CITA\CABLocalInformation do not need to be backed up at all because they can easily be reproduced from the Server PC. However, there is no harm in backing them up as well.

4.1.2 Checking links

The local information database includes links to web pages and to documents stored in the network. Over time the items linked-to from the database can move or get deleted. We have therefore provided a utility to check whether links are valid, or not. To run the utility, run the maintenance program and click the 'Check links' button. Press F1 to call up help text to explain the utility in more detail.

We suggest running the check links utility at least once a month.

4.1.3 Compact and repair

This is a standard Access function to repair a corrupted database and reduce the size of the database by releasing space occupied by deleted records. You can run the procedure manually from within Access. Alternatively you can run the maintenance program and click the 'Repair and compact' button. This will create a backup of the database prior to running 'Repair and compact' in case there is a problem with the procedure. The backup copy will be in the database directory called OldLocalInformation.mdb.

Compact and repair must be run if Access detects the database is corrupt.

Otherwise we suggest running the utility once a month. This will keep the file size to a minimum and will repair nascent problems that have yet to be detected by Access.

4.2 Intranet version

4.2.1 Backup

If you maintain your Local Information System using the desktop version and synchronise changes with the intranet version, you don't need to backup the intranet's database because you can recreate it easily.

If you maintain your database directly in the intranet version the database needs to be backed-up regularly. We provide a backup tool called BackupLIData.exe to help you with that – see the Intranet Installation Guide for more detail.

In either case, you should use an FTP program to take regular backups of the contents of the systemLogs subdirectory. You will also need to take extra steps to backup the document library if you use it (see the Intranet Installation Guide).

The other contents and subdirectories of the intranet application folder need be backed-up only if something in them changes. There is no harm, however, in backing them up more frequently.

4.2.2 Checking links

The local information database includes links to web pages and to documents stored in the network. Over time the items linked-to from the database can move or get deleted. We have therefore provided a tool called CheckLIWebLinks.exe to check whether links held by the intranet are valid, or not – see the Intranet Installation Guide for more detail.

We suggest running the check links utility at least once a month.