

Open eLectos

Version 0.73

An Open source Content Management System for WebApp

A technology preview

*By Wil van Antwerpen
Antwise Solutions
June 16, 2003*

Table of Contents

Table of Contents	2
Introduction	3
License	4
Why is Open eLectos released as open source?	4
Under what kind of license is Open eLectos released?	5
Installation	6
Installation requirements	6
Components	7
Open eLectos Management Console	10
Session Management	19
Feedback Page	21
Database Model	22
Adding textelements to a page	25
Adding a link to a element	25
Adding image elements to a page	26
Delete a Page	27
Displaying the menu	28
Contributions	29
Open eLectos list of files	30

Introduction

Open eLectos is primarily based on the open source webapplication called "eLectos" originally written by Folkert Klemme. Over time this project has been adapted and based on work done in several projects including the VDF-GUIDance area.

Other ideas have been included are based on the various other Content Management System (CMS) engines readily available on the web.

The goal of the application is to be able have a Content Management System which:

- Is very easy to use, maintain and setup.
- Works with any browser (not just the latest Internet Explorer) right now and in the future.
- Very secure,
- Outputs pages based on the current standards and has its layout entirely in Cascading Stylesheets.
- Provides easy access for everyone
- Is easy to extent to new standards without having to rewrite your content....
- Is highly extensible
- Can be used as a framework for all your WebApp projects as it tackles many of the standard problems in which everyone will bump into.
- A system that has as much code as possible in compiled WebApp code as this code is a) much more secure because it is compiled and b) is much easier to use by DataFlex developers as that is what we know best.

Since all the text of the webpages is served from a database they can be changed more easily by making the changes in the database instead of in a static HTML page.

Open eLectos is a true open source application in that the majority part of its code is ugly, there's not much subclassing going on and it does not make much use of relationships between the datafiles. A result of this is that the source of the application looks more complicated as it should have been in the first place.

Although I don't like this part, it is not something to be changed in this revision, stability is our primary concern for the moment, code cleanup will only happen after we are satisfied with any potential stability issues.

There was no documentation available when we started with this project and as a result of this our current documentation should also be regarded as work in progress.

License

Why is Open eLectos released as open source?

There are several reasons for choosing the open source model.

1. A very good reason is that the main part of the source is based on work from Folkert Klemme who has donated eLectos as open source.
2. We are using Open eLectos for our own customers, the end-user is willing to pay for the service to use the product. In the end, our customers will get a better product if more people are using it.
3. Although we think that it is a very good product with a lot of potential, it cannot sold for a reasonable price on which we could also provide support and still make money. If we would have decided to do that, then it is to be expected that the support overhead would eventually kill the product.
4. There are currently not enough companies using WebApp, the market is too small.
5. It is released under the condition that if a developer improves the code, he/she is obliged to send these improvements back to the community so that we ALL can profit from improvements supplied by the community.
6. I just happen to love open source software ;-) The goal for Antwise Solutions is not how-to make as much money as possible but how-to develop good products with many happy users. The more happy users there are, the marrier I am.
7. Learning new techniques and extending technical boundaries is a personal hobby of mine. The Open Source community (not just DataFlex) has always provided me with knowledge that was otherwise very hard to attain.
8. To have a real life webapp example

There is a peer-to-peer support newsgroup available at <news://news.mijnwebapp.nl/open-electos>
We will be monitoring this newsgroup on a daily bases and if possible answer questions there.
This way everyone can benefit from the answer to a question.
Please do not ask questions directly by email, unless you prefer payed support.

**ANTWISE SOLUTIONS DOES NOT PROVIDE FREE SUPPORT, ONE CAN CONTACT US FOR
PAYED SUPPORT ONLY**

Under what kind of license is Open eLectos released?

Basically the LGPL license holds, but in more detail and without the usual legislation mumblejumble it comes down to:

- You cannot claim any damages caused by Open eLectos neither directly or indirectly. This software is provided as is and no costs can be traced back to Antwise Solutions or its affiliates.
- You are not allowed to sell the sources as a whole or any part of the source individually.
- You are allowed to use Open eLectos either commercially or non-commercially, in this case you are in fact the “service provider” of Open eLectos for your customers.
- You are allowed to integrate Open eLectos into your web applications
- If a developer improves the sourcecode, then he/she is obliged to send these improvements back to the community so that we ALL can profit from these improvements.

If you do not agree to use Open eLectos according to these rules then you are not eligible to use this software.

Contact Antwise Solutions at info@antwise.com in case of any doubt.

Installation

At this stage we do not yet have installation programs for the Open eLectos project. So you will have to install it the hard way, using winzip or equivalent and a few extra steps for setting up the workspaces.

If you don't have the time, knowledge or installation requirements to install Open eLectos on one of your own machines then be notified that you can also testdrive the product in an online sandboxed version of the product.

See http://www.vdf-guidance.com/Open_eLectos.asp?Page=OELSANDBOX for more info on this feature.

Installation requirements

- A computer running MS Windows and..
- a webapp 2.1 or webapp 3.0 studio license.
- IIS 5.0

The application might also work on IIS 4.0, but we haven't tested this. Please note that we do not support IIS 4.0 as we want to be able to use IIS 5.0 specific ASP script syntax.

Additional optional requirements are:

- for upload support we use the standard Microsoft's Posting Acceptor, Dennis Piccioni wrote a really good article on how-to install this which can be found [here](#):

http://www.dataaccess.com/ChangePage.asp?PageCode=SUP_G_22WP

- for email support we currently use Aspmail from persits software, see <http://www.aspmail.com>
- for image support we will be using ImageMagick, see <http://www.imagemagick.org/>

Later revisions of OEL are expected to also support Apache under Linux.

This of course will only happen if DAW decides to complete and release WebApp For Linux.

Components

Open eLectos has been designed to support multiple websites from a single executable. Because of this we have tried to move all setup information for a specific website into a meta-database (in our case a DataFlex systemfile).

We can identify two basic components:

- * Management Console
- * The Web application itself

The management console is a Visual DataFlex application and is to be used to setup your meta-database.

You are not required to deploy this application on your WebApp server, but it sure is handy if you have some means to administer the websites.

A required component to identify is the wiki DLL (antwiki4.dll) this is the DLL used for the wiki formatting and generating your session IDs.

This dll needs to be copied into your webapp bin folder, without this DLL your webapp will not work.

To install Open eLectos on WebApp 2.x/3.0 follow these steps.

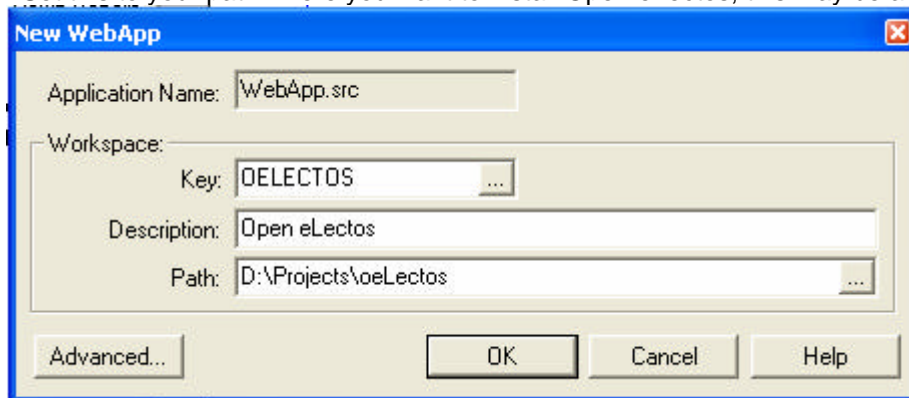
Note:

If you want to run both versions of the webapp then we suggest to you to append a '3' to the virtual and physical paths for the WebApp 3 version. For example in that case the folder "oeLectos" should read "oeLectos3"

Before installing anything, you will first need to use the WebApp Studio to setup the default settings in order to be able to run and compile the application.

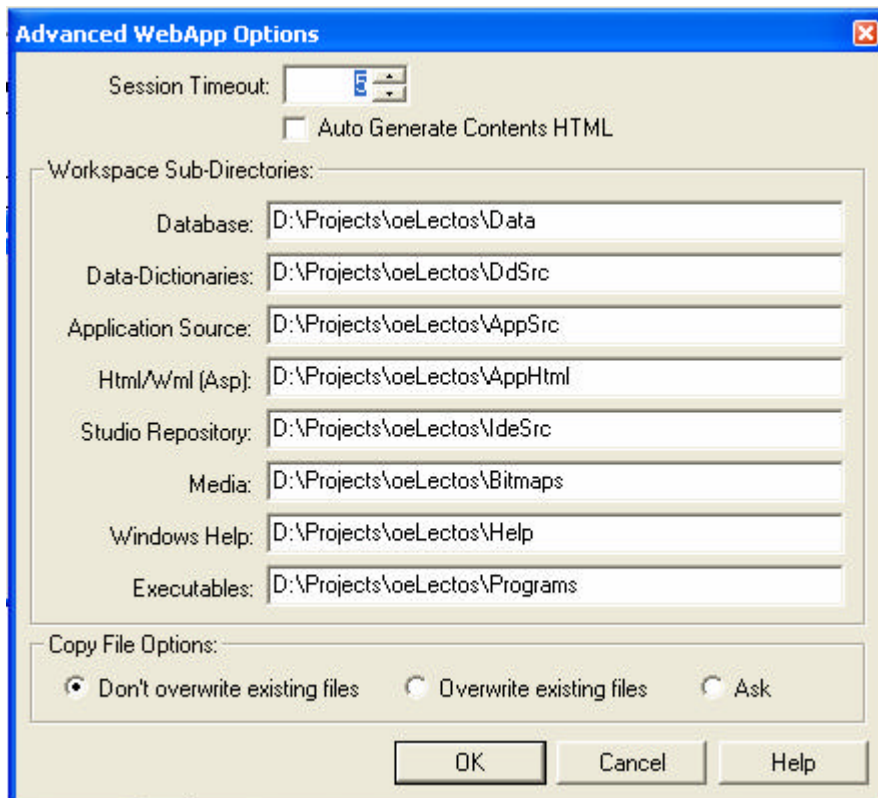
- Start the WebApp Studio
- From the File menu, select New -> WebApp
- Key: OELECTOS
- Description: Open eLectos
- Path: C:\Dev\www\oeLectos

Set this to your path where you want to install Open eLectos, this may be a different path as our path.



- Don't click Ok just yet, instead go to the advanced button and change some of the defaults:
- Set the session timeout to 5 minutes
- Unclick the generate "Auto generate Content HTML" checkbox as it will only help crackers when they try to gather information about your website.

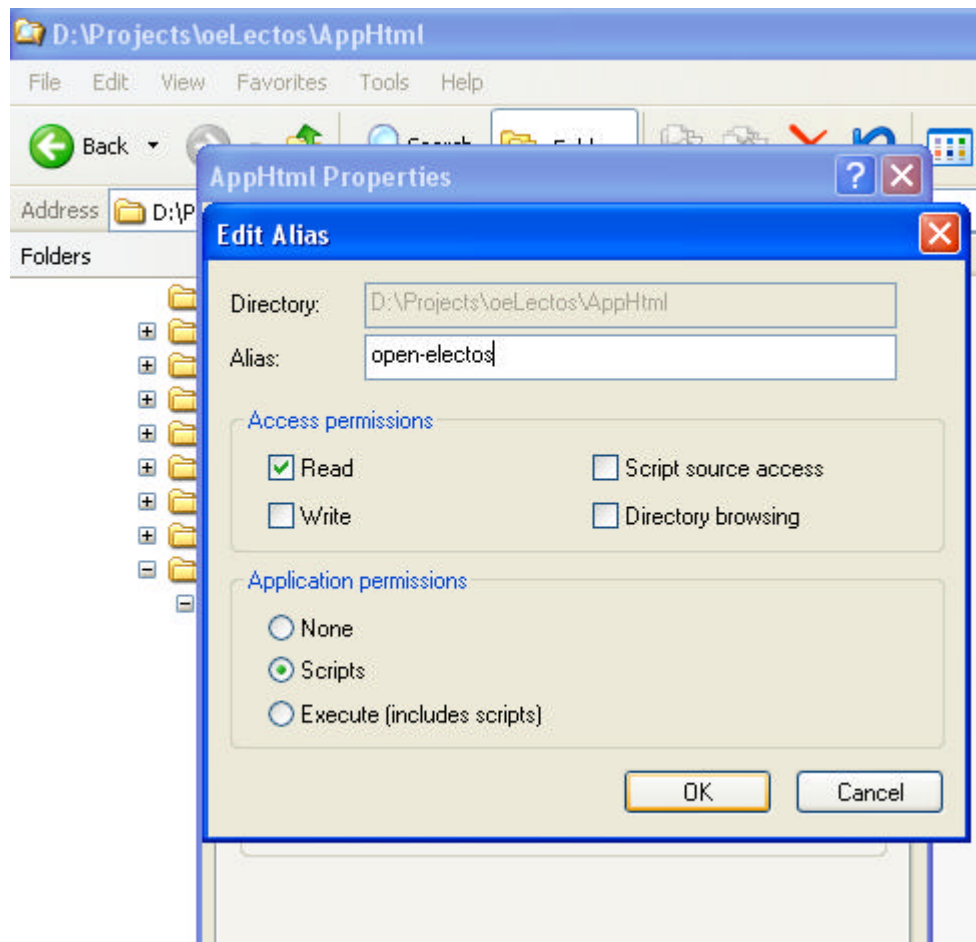
Open eLectos – A Content Management System for WebApp
by Wil van Antwerpen, Antwise Solutions



- Click the Ok button to close the advanced dialog
- Click Ok to create the WebApp
- Close the Studio and all other programs which may have database files open in our freshly created webapp folder.
- Open the WebApp Server Administrator and stop the service by using the File Menu -> Stop Service option.
- Open Windows explorer and unzip the contents of the file Open-eLectos-0.7.3.zip over the folder which you selected to install eLectos in and Click "Yes To All" when you get the "Confirm folder Replace" dialog from windows explorer
- Move the file Antwiki4.dll from the electos folder to the .bin folder of where you installed WebApp. This dll is mainly used for the wiki formatting but it is required for Open eLectos to run. It needs to be in the same folder as your DataFlex Virtual Machine (dfrun.exe etc. in WebApp2.x/3.0)

We are about to create a virtual folder in IIS and we will use windows explorer in order to achieve this. Navigate in windows explorer back to your .\oeLectos\AppHtml folder and rightclick on the AppHtml folder, select Properties from the menu, and select the Web Sharing tabpage, Click Share this folder and enter open-electos as your Alias.

Open eLectos – A Content Management System for WebApp
by Wil van Antwerpen, Antwise Solutions



- Return to the WebApp Server Administrator and start our webapp server service again.
- Select OELECTOS in the WebApp Server Administrator's treeview
- Rightclick mouse and select properties from the menu
- Uncheck the "disable Web Application" checkbox and click Ok

This is it!

To check if your install works, launch a browser and enter the following url in the address:

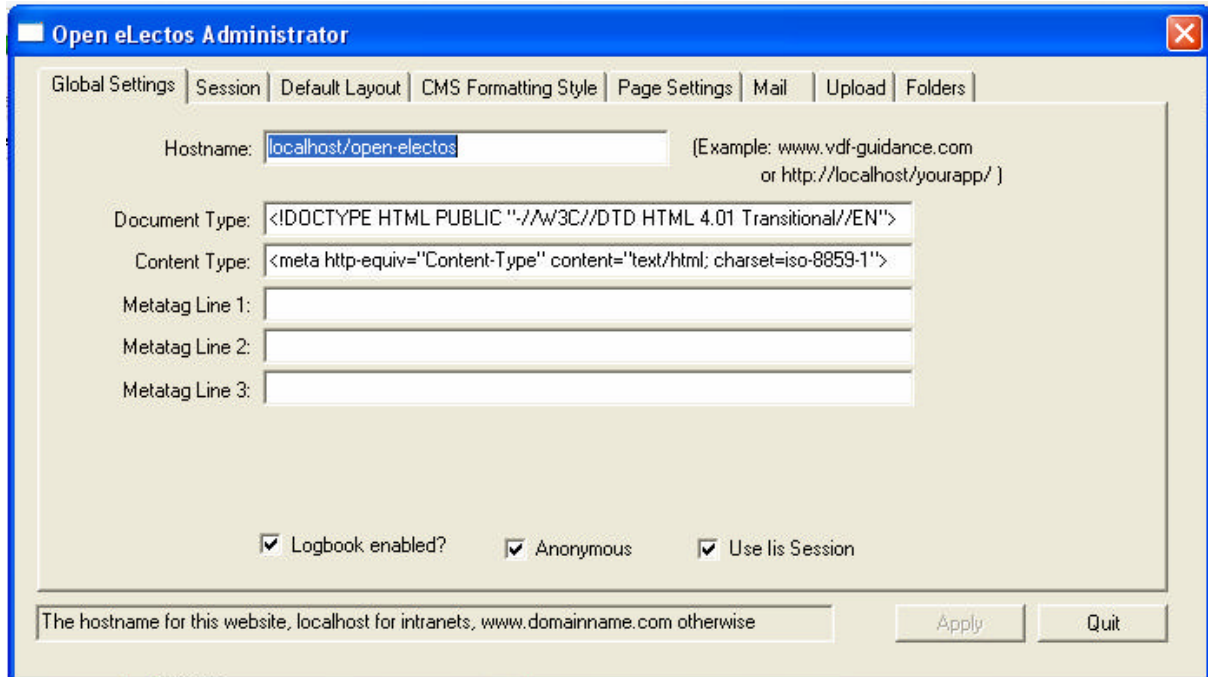
[Http://localhost/open-electos](http://localhost/open-electos)

At this stage you don't need to change any settings with the management console. The database has been setup to work well with these defaults. You might however want to take a look at the current settings to finetune them a bit for your needs.

Open eLectos Management Console

The Open eLectos Management Console merely exists to make it easier to setup your website. In the management console you can setup your virtual folder for the webapp for now, this is a requirement in order for the wiki engine to work correctly and generate the correct hyperlinks to pages within your website.

If you are planning to use the WebApp on a server then the qualified domainname should do.



The screenshot shows the 'Open eLectos Administrator' window with a blue title bar and a standard Windows XP-style interface. The window has a tabbed menu at the top with 'Global Settings' selected. The main area contains several input fields: 'Hostname' (with 'localhost/open-electos' entered), 'Document Type' (with a full HTML DOCTYPE declaration), 'Content Type' (with a meta tag for text/html), and three empty 'Metatag Line' fields. At the bottom, there are three checked checkboxes: 'Logbook enabled?', 'Anonymous', and 'Use Iis Session'. A status bar at the very bottom explains the hostname format. 'Apply' and 'Quit' buttons are in the bottom right corner.

Open eLectos Administrator

Global Settings | Session | Default Layout | CMS Formatting Style | Page Settings | Mail | Upload | Folders

Hostname: (Example: www.vdf-guidance.com
or http://localhost/yourapp/)

Document Type:

Content Type:

Metatag Line 1:

Metatag Line 2:

Metatag Line 3:

☒ Logbook enabled? ☒ Anonymous ☒ Use Iis Session

The hostname for this website, localhost for intranets, www.domainname.com otherwise

Apply Quit

Select if you want to log the browsing behaviour of the user, if anonymous users are allowed and if you want to make use of the IIS session object.

This last one needs some explanation. We only have a requirement for Session IDs being passed around in the url if the user is logged in.

Users browsing anonymously can use the session object to pass parameters and thus also session IDs. If you allow this then you might consider to extend the duration of your IIS session to at least reflect the eLectos Time Out settings.

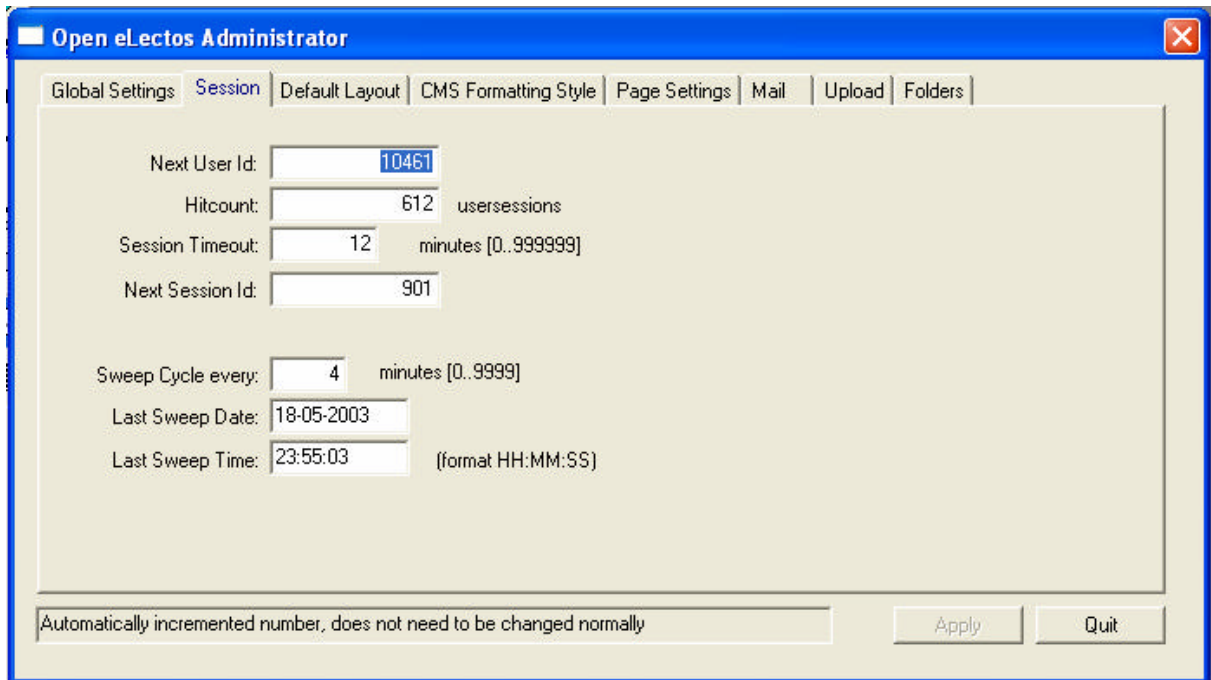
One of the reasons that you don't always want the session ID to be passed on the links is when you email someone a link.

Why is this you might ask?

The reason for this is that you can continue a session within the set timeout period on another browser. If you cut and paste a url with an embedded session ID from browser X into browser Y you can continue from where you currently are.

Understanding this feature is necessary in order to understand on how-to keep your application secure.

Open eLectos – A Content Management System for WebApp
by Wil van Antwerpen, Antwise Solutions



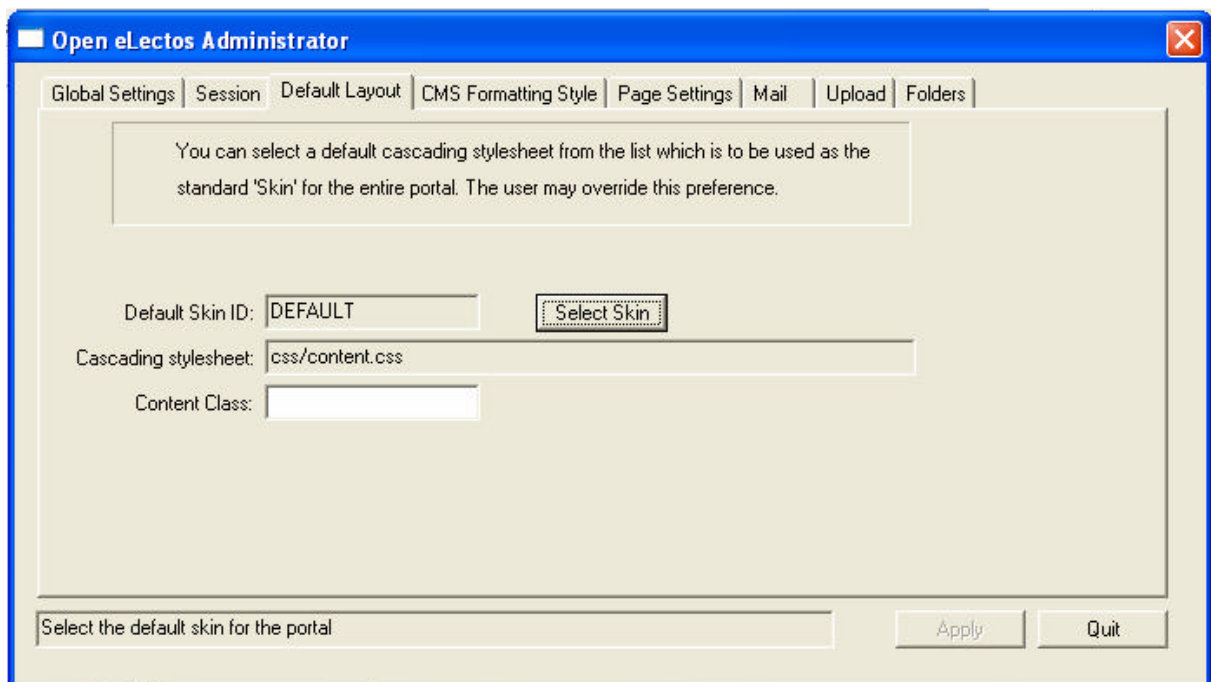
The screenshot shows the 'Open eLectos Administrator' window with the 'Session' tab selected. The window has a blue title bar and a tabbed interface. The 'Session' tab contains several input fields for session management settings. At the bottom, there is a status bar with a description and 'Apply' and 'Quit' buttons.

Field	Value	Unit/Notes
Next User Id:	10461	
Hitcount:	612	usersessions
Session Timeout:	12	minutes [0..999999]
Next Session Id:	901	
Sweep Cycle every:	4	minutes [0..9999]
Last Sweep Date:	18-05-2003	
Last Sweep Time:	23:55:03	(format HH:MM:SS)

Automatically incremented number, does not need to be changed normally

Apply Quit

Hitcount is equal to the number of times a new usersession has been created for your website. This would normally equal to the number of times that your website has been visited. The sweeptime is that amount of time that has passed since the session table has been swept. Sweeping the session table just means that all timed out sessions will be removed from the file session.



The screenshot shows the 'Open eLectos Administrator' window with the 'Default Layout' tab selected. The window has a blue title bar and a tabbed interface. The 'Default Layout' tab contains a text box with instructions, a 'Default Skin ID' field, a 'Select Skin' button, a 'Cascading stylesheet' field, and a 'Content Class' field. At the bottom, there is a status bar with a description and 'Apply' and 'Quit' buttons.

You can select a default cascading stylesheet from the list which is to be used as the standard 'Skin' for the entire portal. The user may override this preference.

Default Skin ID:	DEFAULT	Select Skin
Cascading stylesheet:	css/content.css	
Content Class:		

Select the default skin for the portal

Apply Quit

Open eLectos – A Content Management System for WebApp
by Wil van Antwerpen, Antwise Solutions

The screenshot shows the 'Open eLectos Administrator' window with the 'Global Settings' tab selected. The window has a blue title bar and a tabbed interface. The 'Global Settings' tab contains several options: 'Html enabled?' (checked), 'Outputformat:' (HTML 4.0), 'Wiki enabled?' (checked), 'Emoticons enabled?' (checked), 'Url History' (checked), and 'Css Class for CMS text:' (Content). At the bottom, there is a checkbox for 'Is Html allowed in this web?' and 'Apply' and 'Quit' buttons.

Open eLectos Administrator

Global Settings | Session | Default Layout | CMS Formatting Style | Page Settings | Mail | Upload | Folders

☒ Html enabled?

Outputformat: HTML 4.0

☒ Wiki enabled?

☒ Emoticons enabled?

☒ Url History

Css Class for CMS text: Content

Is Html allowed in this web?

Apply Quit

You can (and should) disable html editing if you allow an anonymous user to make changes to the website.

We recommend you to not uncheck the Url-History checkbox for the moment.

The screenshot shows the 'Open eLectos Administrator' window with the 'Page Settings' tab selected. The window has a blue title bar and a tabbed interface. The 'Page Settings' tab contains several options: 'Website Friendly Name:' (Open eLectos Preview Release 0.72), 'Page Header:' (HTML code for a banner), 'Company Logo:' (HTML code for a logo), 'Page Footer:' (HTML code for a footer), and 'Page Counting' (checked). At the bottom, there is a text field for 'Userfriendly title for the website to be displayed in each page's caption bar' and 'Apply' and 'Quit' buttons.

Open eLectos Administrator

Global Settings | Session | Default Layout | CMS Formatting Style | Page Settings | Mail | Upload | Folders

Website Friendly Name: Open eLectos Preview Release 0.72

Page Header: `<div class="PageHeader"></div>`

Company Logo: ``

Page Footer: `<div class="PageFooter">Copyright © 2003 Your Company Name Here.</div>`

☒ Page Counting

Userfriendly title for the website to be displayed in each page's caption bar

Apply Quit

These settings can be used from within in the asp-templates.

Open eLectos – A Content Management System for WebApp
by Wil van Antwerpen, Antwise Solutions

The screenshot shows the 'Open eLectos Administrator' window with the 'Mail' tab selected. The window has a blue title bar and a tabbed interface with tabs for Global Settings, Session, Default Layout, CMS Formatting Style, Page Settings, Mail, Upload, and Folders. The Mail tab contains a text box explaining that the email information is used for sending automated emails or posting in a support newsgroup. Below this are four input fields: 'Smtp Hostname:', 'Nntp Hostname:', 'Nntp Login:', and 'Nntp Pass:'. The 'Admin Emailaddress:' field is pre-filled with 'info@yourdomain.com'. At the bottom, there is a status bar showing 'Smtp-hostname for sending of emails' and two buttons: 'Apply' and 'Quit'.

These are the settings which are used from within the pages that use the asppmail component.

The screenshot shows the 'Open eLectos Administrator' window with the 'Upload' tab selected. The window has a blue title bar and a tabbed interface with tabs for Global Settings, Session, Default Layout, CMS Formatting Style, Page Settings, Mail, Upload, and Folders. The Upload tab contains a text box explaining that enabling uploads to an internetserver is a thing to consider seriously and that the application will only allow authenticated (registered) users to do this. Below this are two checkboxes: 'Upload enabled?' (checked) and 'User Upload enabled?' (checked). There are two file filter lists: 'Upload Image Filter' with a list of image and document formats, and 'Upload File Filter' with a list of various file formats. Below these are two input fields: 'Antivirus Program' (pre-filled with 'C:\Program Files\Norton AntiVirus\NAVW32.EXE') and 'Antivirus Parameters' (pre-filled with '*/s'). At the bottom, there is a status bar showing 'Upload enabled?' and two buttons: 'Apply' and 'Quit'.

Uncheck the 'upload enabled' checkbox to completely disable the ability to upload files into your web applications repository.

At this stage the antivirus check has not yet been implemented.

Open eLectos – A Content Management System for WebApp
by Wil van Antwerpen, Antwise Solutions

The screenshot shows a window titled "Open eLectos Administrator" with a blue title bar and a close button. The window contains a tabbed interface with the following tabs: Global Settings, Session, Default Layout, CMS Formatting Style, Page Settings, Mail, Upload, and Folders. The "Global Settings" tab is active. Inside the tab, there are four text input fields with labels to their left: "Emoticon Path:" with the value "images\icons\", "Upload Path:" with the value "C:\Dev\www\uploads\", "User repository:" with the value "Contrib\", and "Images Path:" with the value "SiteImages\". To the right of the "Images Path:" field is a small text label "(Example: images\)". At the bottom of the window, there is a status bar and two buttons: "Apply" and "Quit".

Emoticon Path:	images\icons\
Upload Path:	C:\Dev\www\uploads\
User repository:	Contrib\
Images Path:	SiteImages\

(Example: images\)

Apply Quit

Multicompany Deployment

An idea from Knut Sparhell

For the moment this only works for WebApp 2.x, Knut already managed to find a workaround to implement this in WebApp 3 but it is delayed to be implemented in the first upcoming build.

We want to run the same application on different databases (hosting the same WebApp for several customers on the same server). In order to be able to use different databases you'll need different workspaces, says the book. It also says that the workspace of a WebApp cannot be changed.

Ok, so the database to use by the webapp can not be dynamically changed, but does it need to be hard coded?

The problem with not hard coding them is where to get the database location from in the initial phase of the program. Deployed VDF applications can be started with either the workspace or a registry key passed as a parameter.

Why is this not the standard way of invoking a WebApp?

Ok, so if the workspace really needs to be hard coded, then we would need several independent developed WebApps. That would mean several development environments and several compiles of the same logical code, just in order to support one application! No Way.

We must have one and only one VDF-file out there, or the whole project becomes too complicated. MySolution, free to use, as a tip, if you find it usable for you:

Step A:

In the very upper code of the WebApp, the piece of code which is normally generated is changed from :

```
Object oWorkspace is a cWebAppWorkspace
    Set WorkspaceName to "myworkspace"
End_Object
```

to this:

```
Object oWorkspace is a cWebAppWorkspace

    // Lazy man's mixin
    #Include DoLoadWorkspace.pkg
End_Object
```

The include file contains code which looks similar to:

```
String gsCmdline // Global String
Cmdline gsCmdline
Set WorkspaceName to gsCmdline
```

This allows us to add a parameter on the commandline, if no parameter has been added, then the workspace constant defined by CURRENT\$WORKSPACE will be used.

Compile your WebApp and deploy it to the Programs folder as usual.

Step B:

Deploy the different databases you have to their own folders, wherever you want them to reside. (The default location is the Data folder for each WebApp. You may have several Data folders residing under your application's main folder by calling them Data-A, Data-B or whatever is desired.

You may find it wise to give them names corresponding to a workspace you intend to make for accessing them.)

Step C:

Regedit on the server. By exporting the actual workspace, you may modify it in the file and then merge it back to the registry. Remember to change the name of the workspace too, not only the paths. Now you have the opportunity to have different paths for your data and Filelist.cfg. Keep the ProgramPath the same for all workspaces of the same application.

Go to the WebApp Administrator on the deployment server and make new WebApps for the new workspaces. Before applying, take control over the properties the new WebApps and:

1. Edit the VDF Program Name field and _append_ the workspace name as a parameter. Then **add quotes** around the whole string. It will not work without the quotes! Example:

Original: C:\App\MyApp\Programs\WebApp.vd7
New: "C:\App\MyApp\Programs\WebApp.vd7 myworkspace"

2. Change the Logfile name if logging is desired or used. You will probably need to have separate logs. Example: C:\App\MyApp\Programs\ myworkspace.log

Apply the new WebApps and close the properties view by Ok.

Keep track of the names of your WebApps. The WebApp name is by default the same as the workspace name. It does not need to be so, and you may change the name of them. The name of a WebApp is a reference for the ASP-processor on your IIS, and is referenced in the file global.asa. A new global.asa is written every time you compile your WebApp. The name of the WebApp used is always set to the same name as the workspace. This is just a convention introduced by the WebApp concept. Do not confuse the WebApp name with the name of the VDF-file, which is at all times WebApp.vdf (f=6 or f=7 for now).

However, to keep it clean, use the same name for the workspace and the WebApp itself.

Then Stop and Start the WebApp Service.

Step D:

Make several AppHtml-folders called A-Html, B-Html or whatever. It is probably wise to keep the names in some way in correspondence with the workspaces created and the databases it points to.

Copy the total contents of the current AppHtml to these new folders. You may rename AppHtml to fit your naming scheme. If you want to rename AppHtml, first stop the Web Sharing of it (in IIS or the folder properties.)

Create new virtual directories pointing to these folders, either by editing the Web sharing properties of these folders in windows explorer, or by using the IIS Admin. Remember to enable scripts.

Be careful when creating a _name_ for the share.

This share will be used in the address when accessing the corresponding database from the web.

Name it like: "/myworkspace "

Step E:

Edit the file global.asa in each Html folder.

Look for the following line:

Open eLectos – A Content Management System for WebApp
by Wil van Antwerpen, Antwise Solutions

```
Application("hSessionMngr")=VDFSessionManager.Initialize("<webappname>")
```

(where <webappname> is the actual WebApp name in use, taken from Administrator, and this name will still correspond to a special workspace name, probably the same name.) Like this:

```
Application("hSessionMngr")=VDFSessionManager.Initialize("myworkspace")
```

Use:

First database: <http://servername/myws-1/>

Second database: <http://servername/myws-2/>

Step F:

Update the management console

Copy the Open eLectos management console (OEMMC.VDx) to the programs folder.

This Visual DataFlex application can be used to apply specific settings for each website.

It can be started from the commandline to edit the data in any of the workspaces by appending the workspace name. Below is an example of how to set up a shortcut for a separate website.

In this example the website myws-1 is used.

- go to your desktop
- right-click mouse
- select new shortcut
- For the location, type the path to the oemmc.vdx file to use or alternatively use the Browse button to locate the file, for example: P:\Electos\Programs\oemmc.VD7
- Append the workspace name which you want to edit to this location, example:
P:\Electos\Programs\oemmc.VD7 myws-1
- Click Next and type the name of your website, myws-1
- Click Finish

If you like you can also use the icon, OELlogo.ICO in the bitmaps folder for your shortcut, but this is not mandatory.

Note: Make sure that you test that the management console picks up on the new workspace. We have seen a problem with this when using a default VDF7/WebApp3 setup.

The reason for the console not recognizing commandline parameters is in an unexpected area.

If you experience this then fire up your registry editor and check the following registry entry

[HKEY_CLASSES_ROOT\Dfrun7.Document\Shell\Open\command]

Now check the (default) value, you're likely to find a value similar along to these lines:

"C:\VDF7\Bin\Dfrun.exe" "%1"

Because the program itself is the first parameter for the VM, the program will not identify the workspace parameter and simply ignore it. Changing your (default) value for the registry key into the suggestion below should fix this.

"C:\VDF7\Bin\Dfrun.exe" "%1" "%2" "%3" "%4" "%5"

The path where the runtime component is located may be different on your machine.

Precautions and development issues:

The two (or as many as you like) addresses above will now run logically two WebApps, and not two sessions of the same app. It is very important to understand this. Even though the WebApps will be identical in code. If you have not put any dependencies in the database folder, they will also be identical in behaviour.. Of course, you may have these dependencies, if you like. The system file (if any) or just a sequential file there among the database files will do. What you will probably read somewhere is the owner or the customer name.

Version 0.73	Page
June 16, 2003	17 of 33

Dependencies of the ASP pages may easily be put in as special files in either the Html folder, the inc subfolder or any. I would advise not to put such special files in either of the standard folders. It is better to make a new folder under the Html folder.

This way you are decreasing the risk of copying over the entire content to another workspace area.

Deployment of application updates, where there is no new WBOs:

Copy (deploy) the vdf file as usual. Copy (deploy) the AppHtml content except global.asa to each of the Html folders which you have for this application.

If you have remote control: The fastest way is of course to deploy to one folder and then copy the content (without the global.asa) to the other Html folders.)

Deployment of application updates, where there is new WBOs compiled in:

Include the global.asa in the deployment and then edit them remotely as in step D above. The reason for this is that each WBO has an entry in global.asa, as a defined OBJECT for the ASP.

If you do not have remote control you may edit them locally and then deploy them. Local access to the server or remote control is only needed when setting up your WebApps.

Session Management

A complete story of its own.

Each page request should be considered as a separate process, the time inbetween 2 page requests is unknown. As a reason of this we cannot expect that our WebApp process still lives between 2 page requests. Since WebApp 3.0 we also have process pooling which makes all of this even less predictable as a single WebApp process can be shared by multiple IIS sessions.

The user of our webapp however does expect the application to remember his selections and settings during a session or maybe even over multiple sessions.

So we need a consistent and easy to maintain solution for persistence, preferably written in WebApp itself as that can be compiled (security) and is easy to work with for the most of us as it still is plain DataFlex code.

We identify several types of users, but the most important ones from a session specific point of view are the anonymous user and the logged in(=authenticated) user of our system.

IIS has a build-in object which can be used for storing session specific information, the session object. However this object does not really fit our needs as it has several drawbacks:

1. You do not have much control over how long a session lives
2. Each session is identified by storing a temporary cookie on the user's machine. If the user disabled storage of cookies, each page will be regarded as a new session.
3. This session object is an object only accessible directly from within ASP, we would prefer to have a session object in WebApp.
4. An IIS session object can relatively easy be hijacked by a malicious user.

There are many ways in how one can solve this problem. We are not going into detail on this as you can find much discussion about this on the web. The solution we opted for to use does generate our own unique session ID and stores the data in the datafiles named Session and User.

The data stored in there is retrieved at the top of each page and publicly made available in the oSession object in our WebApp.

Advantages of using our local session object:

1. It is directly available from within WebApp, so much easier to work with as from the IIS object where you would have to define methods to transfer data from and to your program.
2. Our session object contains more useful properties, for example if the user agent * his browser *
3. Use of our own session object is much more secure due to a) the strong encryption scheme used and b) there are currently no known defects/exploits that a malicious user could use
4. The session object is directly linked to a user either anonymous or authenticated. If not an anonymous user, then it is really easy to store new preferences that can be stored over multiple sessions.
5. We have total control over the time our session object lives as it is actually a physical record in the session database. This session database is cleaned up (swept) at a preset time interval.
6. You can easily add new properties to our session object by adding new fields to the session datafile.
7. Since we store the Querystring you can also parse this and get the variables passed on a GET interface. (Note: You can also use DoGetHtmlFormValue for this)
8. It is possible to restart IIS (and thus WebApp) without having users noticing this or breaking their session. This also means that you can recompile your application without your session being interrupted so you don't have to start from the begin again after fixing a bug.

When logged in (authenticated) then the encrypted SessionID is passed in the Querystring on each page. If not logged in, then this SessionID is by default stored in the IIS session object.

The encrypted SessionID is a strong encrypted version of a combination of the Session and User Record Ids and a dynamic public key.

We have used 128 bits AES (American Encryption Standard) encryption as our method of choice to encrypt the session ID.

We are almost there, but - as usual - there's always some reason that complicates things. In the solution described above, we would have to embed the session id within every URL on the entire site. This does not really create that big of a problem for us, but it does not help users of eLectos much when they want to share URLs with friends or colleagues. Having a URL that wraps over multiple lines in an email ain't transparent. It is also not required (nor recommended) to share your session id with multiple users. Even while such a session id is only valid for a limited time that can be set in the management console, it does not improve our security.

In addition to this it is most of the times not really a problem if a session times out for an anonymous visit to our site. The user just gets a new session id, inherits some defaults and that's it. No biggie. As a result of this it has been decided to store the session id in the IIS session object for anonymous use and embed the session ID in the URL for authenticated use. If the page is called and passes an encrypted session ID in the URL, then this session ID will have priority over any Session ID stored in the IIS session object.

These are the different scenarios that can happen:

Note for self, need to check if this is still the case

1. New user,
UserID=meaningless, there will be no record of him/her nowhere.
SessionID is created for an anonymous user.
2. Existing user, no cookies stored but sessionID passed
3. Anonymous user which is succesfully logging in.
4. Existing user with permanent cookie
The cookie contains the encrypted Session ID. We can retrieve a meaningfull user-ID from this. The decrypted Session.ID may or may not have a meaning, depending if the session has timed out or not.

Feedback Page

NOT YET IMPLEMENTED

Most website's want to implement a page which one could call a feedback page.

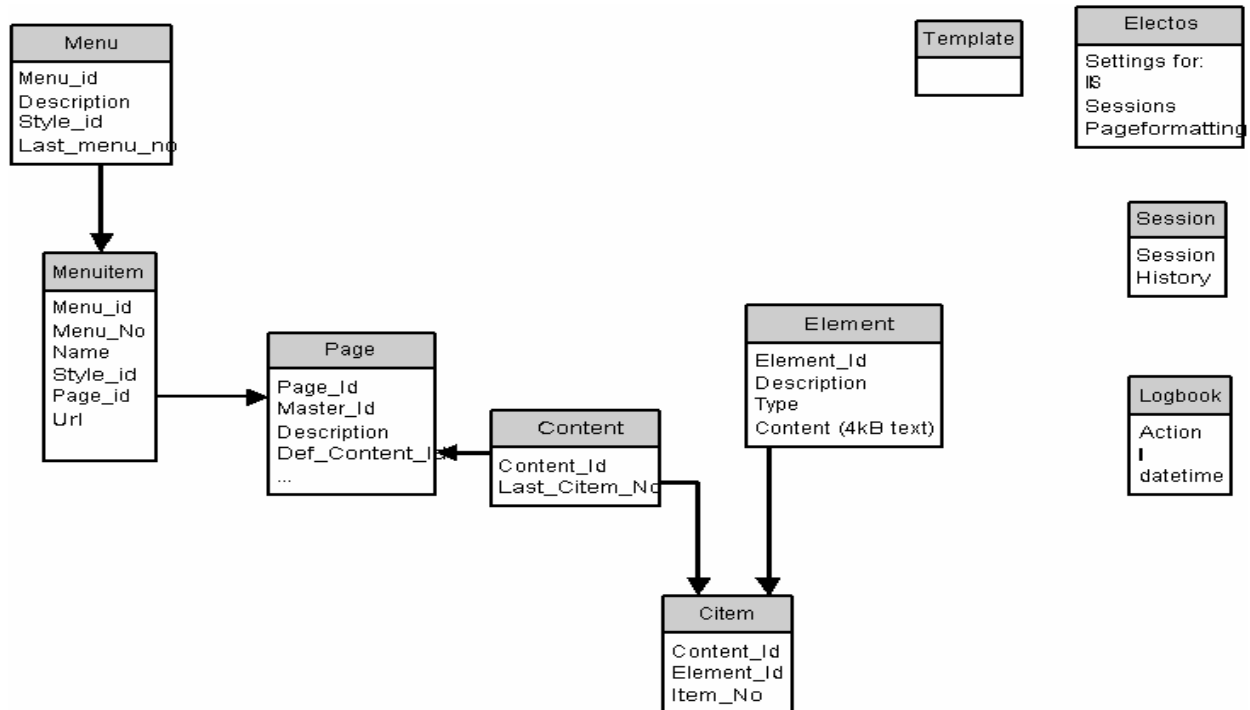
We have supplied a basic page which resembles this, it is called feedback.asp and this page is supposed to be configured based upon settings that have been made from within the administrator module.

So what's on the page?

- A CMS area in which the website admin can add some comments
- Email address from the user (required or not)
- A text area in which the visitor can write his/her comments about the website
- A Submit and Reset button

Database Model

The database model is still pretty much the same as the original electos.
This is likely to change over time, but here is the 'map' of how it looks like right now.



The template file must contain all the templates from which you want to be able to choose.

Create a new page

This is an "analysis" of the changes made to the database when creating a new page.
The reason this is written out like it is, is because of the complete lack of documentation when picking up on the project a long time ago.

Creating a New Page makes changes to several files.
For example adding a page with the ID "VDFQUERY" as a child of the projects page (OSDPROJECTS) makes the following changes to the database:

File CONTENT

New Record, recnum = 18
CONTENT_ID: [VDFQUERY]
DESCRIPTION: [VdfQuery]
STYLE_ID: [CONTENT]
LAST_CITEM_NO: [0]
REVISION: [1]
PUBLISH: [Y]
CREATED: [29/08/2001]
CREATED_USER: [WvA]
UPDATED: [29/08/2001]

File MENU

Record Changed for record with field MENU_ID = [OSDPROJECT]
LAST_MENU_NO: [3] <> [2]
UPDATED: [29/08/2001] <> [03/05/2001]

File MENUITEM

New Record, recnum = 23
MENU_ID: [OSDPROJECT]
MENU_NO: [3]
NAME: [VdfQuery]
STYLE_ID: [SUBMENU_ITEM]
PAGE_ID: [VDFQUERY]
URL: []
FROM_DATE: []
UNTIL_DATE: []
REVISION: 0
PUBLISH: [Y]

File PAGE

New Record, recnum = 13
PAGE_ID: [VDFQUERY]
MASTER_ID: [OSDPROJECT]
DESCRIPTION: [VdfQuery]
STYLE_ID: []
CATEGORY: [0]
TYPE: [0]
URL: [ViewPage.asp?Page=VDFQUERY]
PAGETYPE: [D]
DEF_CONTENT_ID: [VDFQUERY]
DEF_TEMPLATE_ID: [CONTENT]
DEF_MENU_ID: []
USERGROUP: [0]
FROM_DATE: []
UNTIL_DATE: []
COUNT: [Y]
REVISION: [1]
PUBLISH: [Y]

Notes:

PAGE.DEF_MENU_ID contains only data if it is a menu

Adding textelements to a page

Then if we add some text to a page like this, we get the next changes:

File ELEMENT
New Record, recnum = 47
ELEMENT_ID: [VDFQUERY0]
DESCRIPTION: [OSD VdfQuery Header]
TYPE: [3]
STYLE_ID: [HEADER]
CONTENT: [VdfQuery]
REVISION: [1]
PUBLISH: [Y]
CREATED: [29/08/2001]
UPDATED: [29/08/2001]

File CITEM
New Record, recnum = 39
CONTENT_ID: [VDFQUERY]
ELEMENT_ID: [VDFQUERY0]
CITEM_NO: [1]

File CONTENT
Record Changed for record with field CONTENT_ID = [VDFQUERY]
LAST_CITEM_NO: [1] <> [0]
REVISION: [2] <> [1]

Adding a link to a element

The CMS allows you to have an entire element to be a hyperlink.
Previously the system did go to the link if you clicked on the hyperlink part of the element while in editmode. So the editbox pops up, but the content page just moved away to its new destination. We have experimented a little bit with this to have the user the best functionality and to avoid confusion. Creating an element that was only partly a link was one of the solutions.

Now we figured out that if the content page goes to its next destination and the user tries to delete a page, he will get a script error that is really hard to get rid off (we are trying to do a javascript reload on the opener page that points to a url which does not have to be in our domain. The result is a permission denied error)
So in the end, the solution is to have a link not be a link when in edit-mode.

Adding image elements to a page

The field ELEMENT.CONTENT contains the name of the image that is to be loaded into the html page.

When the page is constructed in Content.wbo, the relative path that is going to be used is added in front of the filename.

This relative path is normally stored in the systemfile Electos, field ImagesPath.

In some occasions this will not suffice and we will add a different path to this.

An example of this for the vdf-guidance site are images that are stored in the repository.

The NewImageField.asp file uses the CreateImageCombo method in the EditPage wbo file.

This method allows for the dynamic setting of a physical folder where the images to select from are to be retrieved. If no folder is selected the global images folder is its default.

Ditto for the EditImagesFolder.asp file as it uses the same method.

The content wbo is used to display the correct images and will need the virtual folder of where the images are stored in order to make up the page.

Delete a Page

Deleting page with PAGE_ID [FOO] should go as follows:

- 1) First we check if page [FOO] is not a Masterpage for some other pages, to do this, we check if in file MENU a record exists with MENU_ID = [FOO] (optional)
If it does, we have a masterpage and we'll check for childpages by traversing the MENUITEM file and see if there's a record with MENU_ID = [FOO], if there is, we have a masterpage with at least one child and we cannot delete the page.
At least that is how the masterpage check should have been done. For now, we just check if PAGE.DEF_MENU_ID <> "", because it contains only data if it is a menu
- 2) Search MENUITEM for a record with PAGE_ID equal To [FOO] and if found delete it.
- 3) Search PAGE for a record with PAGE_ID = [FOO] and if found delete it.
- 4) File CONTENT, search for CONTENT_ID = [FOO] and delete the record.
- 5) File CITEM, search for all records that have a CONTENT_ID = [FOO] and take the value of the field ELEMENT_ID (FOO0, FOO1, FOO2) to search the file ELEMENT with and check if there is a record in there with ELEMENT_ID = [FOO0] and if there is, delete the record.
After this one can delete the CITEM record.

If we want to keep the contents of the page, but not the layout, the above procedure still counts, but the file ELEMENT should be untouched.

Displaying the menu

The menu is build with the use of several files. Files involved are:

MENU, MENUITEM, PAGE and STYLE.

It is not much more as a simple report in the way we present it now.

Realize however that because of our javascript detection you will be able to send out a much fancier menu to the users browser once we detected that he or she has scripting enabled.

Function DisplayMenu String sRoot String sDisplayMode String sCurrentPage String sMasterID
Returns String

// Build submenu

Get DisplayMenu Page.Def_Menu_ID sDisplayMode sCurrentPage sMasterID to sDummy

If sRoot = "", it will get the default value of "DEFAULT" (Our ultimate rootpage)

Contributions

We want a restricted number of end-users to be able to edit their own texts into the system.
For this to be able to do so, we will make it possible for the end-user to create his own page, but this will be added automatically in the background.

The masterpage for this is the page that contains the contribution.

Ideas to limit the users ability to navigate over the site and make changes to it are:

If we take as a new contribution of the type "Package" as an example.

- In file menu, set the PUBLISH field to "N"
- Do not give the end-user the following options:
 - 1) New Page (this may be added in the future)
 - 2) Sitemap
 - 3) Edit Sitemap
 - 4) New Contribution Container Object
 - 5) New User Contribution Container Object
 - 6) Preview

Contributions can be added to every page you want.

You do this by adding a Contribution Container Object. This object is linked to a normal text-element and it can contain several contributions.

The Container Object has several properties. The most important one of these is the Category.

There is also a Userspecific Contribution Container Object. This object contains only the contributions of the current user.

Creating a new contribution

New contributions will get their own repository on the website.

This means that during the creation of the contribution a new folder will be created to store optional files for the contribution.

The name for this folder is always constructed based on the type (How-To's, Packages...), category (Classes, Datadictionaries, Windows API...) and the projectname.

Once the folder is created it will not be able to change.

The name of the folder is stored in the field CONTRIBFOLDER of the file CONTRIBC.

Open eLectos list of files

\AppHtml

_Logout.asp	Log out of the current session
_Preview.asp	N.A. old leftover
_Redirect.asp	Used to redirect pages
ChangeSkin.asp	Change the current Cascading Stylesheet
Cms.asp	N.A. old leftover
Cmsmenu.asp	N.A. old leftover
Debug.asp	
Default.asp	Standard ASP start page, will redirect to our homepage
DeleteElement.asp	Deletes CMS elements
DeletePage.asp	Delete the entire selected CMS page and contents
EditHTMLField.asp	Allows the user to edit the HTML text in the selected HTML-element
EditHyperLinkField.asp	Edit hyperlink.. expected to be removed at a later stage
EditImageField.asp	Allows the user to edit the image properties for the selected Image-element.
EditMode.asp	Toggle between Edit Mode and Preview Mode
EditSitemap.asp	Edit the sitemap
EditTextField.asp	Allows the user to edit the selected text. Parameters: RecId, Citem_No, Last_Citem_No, CItemNo.....
ElectosTemplate.asp	N.A. old leftover
Error.asp	N.A. old leftover
Help.asp	Help page template
ListRepository.asp	N.A. old leftover (will be used again)
Login.asp	Login page
Mailpassword.asp	Mail the password to the users registered emailaddress by using the Persits send mail component.
MoveElement.asp	Move element Up/Down within the current page
MovePage.asp	Move the page Up/Down in the current menu
NewContribField.asp	The page that is used to add a new contribution.
NewPage.asp	Is activated from the "New Page" button while in admin mode. No Parameters.
NewTextField.asp	Is activated from the "New TexField" button in administration mode. Parameters: RecId, PageId
PageProps.asp	Allows you to edit the menu title, the template used by the page and to move up and down in the menu and to delete the page from the system. Parameters: PageRecId, MenuItemRecId, LastMenuNo
Profile.asp	Template page to edit profile information
Repost.asp	Shows the status of an uploaded contribution file.
Repostimage.asp	Shows the status of an uploaded image
TimeOut.asp	Same as the Login page except it is shown when a session times out.

Open eLectos – A Content Management System for WebApp
by Wil van Antwerpen, Antwise Solutions

Upload.asp	Page to upload a contribution
UploadImage.asp	Page to upload an image to

ViewPage.asp	Template page for normal textpages
--------------	------------------------------------

\AppHtml\inc

cmsinit.asp	Asp SSI file to be used in normal CMS pages
umsinit.asp	Asp SSI file to be used in the user menu system pages
editinit.asp	Asp SSI file to be used in one of the edit content pages

\AppSrc

Content.wbo	This is the business process that functions as a page-server. Its main method is Function Display.
ContribContent.wbo	This business process is called from the content BPO to display and edit the contributions.
ContribEntry.wbo	Process to create a new contribution, move an uploaded contribution to the contributions repository.
EditPage.wbo	Move an uploaded image to the images folder.

Menu.wbo	Contains logic for the menu
----------	-----------------------------

