v3.0

[PDF Suite]

for Lasso Professional 8.5

PDF_FromHTML

User Guide and Installation

[EXECUCHOICE]
software development

Contents

	Page
Introduction ……	3
About the Installer ……	5
[PDF_fromHTML] ……	7
[PDF_GetInfo] ……	9
[PDF_toHTML] ……..	10
[PDF_toTEXT] ……	13
[PDF_SetFieldValue] ……..	15
[PDF_GetFields] ..……...	18
[PDF_MergeFDF] ……	19
[PDF_ExportFDF] ……	21
[PDF_MakeFDF] ……..	22
Examples with plain links and Sessions (by Rich Fortnum) ……	23
Configuring CUPS for faxing and printing with EC PDF Suite …….	27
Included Software and Source Locations ……	29
Disclaimer ……	30
Special Thanks ……	30
GNU License ……	31
HTMLDOC License Agreement ……..	36

Introduction

Thank you for choosing the ExecuChoice PDF tag suite. While this product is primarily free, I would like to call it "Donation Ware". Any money earned will be used to fund future development. The suggested minimal donation is 100$ (USD). Please contact me if you would like to donate or be a sponsor. The work on this project has been quite extensive and time consuming. In this guide we will cover the complete usage and provide some examples of these tags. All tags are completely rewritten from previous versions and have many new features. These tags were designed to bridge the gap between what LassoSoft provides and what is necessary to encompass all PDF functions. There are three basic methods to making PDFs. Originally people tried the ol' string_replace method but this does not work. There is a little thing called the xref table which maps out locations and lengths of everything in the PDF. It is a hit or miss game and not a guarantee. So, below are your only options.

- You can use Lasso's built in PDF tags and manually draw the PDF. This is very time consuming and slow. Lasso uses iText which is Java based. To draw a full PDF and can take ridiculous amounts of code not to mention many CPU ticks to do it. In a web environment this is the least efficient method.

- The [PDF_fromHTML] tag described in this PDF is by far the fastest. It is based on HTMLDOC which is a C/C++ based application. It works by taking a Lasso page which may contain dynamic data and convert it into a PDF. This is VERY fast. It does, however, have a small draw back. The current version of HTMLDOC is limited to basic HTML and has no CSS support as of yet. It is in beta version, but is not stable enough for this release. In the interim, it is necessary for the use of basic HTML font="xxx" type tags. HTMLDOC currently supports the basic and most common fonts used for PDFs. Future plans for HTMLDOC will support CSS and embedding fonts. In the event that EasySW, the maker of HTMLDOC, is not able to get a new release out soon supporting CSS, I will switch to a different back end, "html2ps" to handle HTML pages written with CSS.

- The third option is faster than first option but not as fast as the second option. This one, however, offers a more elegant solution. Simply design a PDF in your favorite application and then use Adobe Acrobat to place form fields on your PDF. Then use the included tags to populate the fields and save it. Options are built in for saving to disk, streaming, printing and faxing and of course returning the raw data. This is by far the most efficient method that guarantees your PDF will look the way YOU want it. An example of this option has been included in the ExecuChoice_Tags folder created in your lasso folder.

Please be sure to read **all** pages in this manual as there are many notes and examples.

About the Installer

This section describes the components of the ExecuChoice PDF Suite that have been installed on your system.

The installer will be looking for the following path:
/Applications/Lasso Professional 8
If you have changed this path then you will need to make an alias of it with the proper name so that the installer will know where to look.

- Custom LCAPI module – EC_Helper
- EC_PDF_Suite_V30.LassoApp
- EC_pdftotext command line tool
- EC_pdftohtml command line tool
- EC_pdfinfo command line tool
- HTMLDOC software/command line tool
- HTMLDOC PDF manual
- EC_PDFSuite.PDF manual
- Example folder containing an example of the [PDF_SetFieldValue] tag.
- Scripts to enable/disable the at/atrun function of MacOS X for [PDF_toHTML]
- Uninstaller script

Items 3, 4 and 5 are custom builds of the open source XPDF package. The installer will also enable the "at" command line tool and SMB printing. In enabling the "at" command, which is an extension of the cron task, a process will be performing every 5 minutes to check for any tasks set to be performed. While on a server this is fine and expected. The provided scripts will allow you to enable/disable this feature for usage of these tags on a laptop while doing development. Disabling this feature will prolong battery charge. However, any temporary folders created by [PDF_toHTML] will not be deleted until "at" is activated again or you manually delete them. It is imperative that the lasso user has read/write permissions on any folders/files where the tags will be accessing.

The following is the default line of the /etc/crontab file on MacOS X 10.3 and older which will be uncommented to make the "-expire" feature of the [PDF_toHTML] tag work correctly.

```
#*/5    *       *       *       *       root    /usr/libexec/atrun
```

If you have removed this line then you will need to add it, uncommented, to make this feature work correctly. If you have never touched this file then you have nothing to fear.

In MacOS X 10.4 and up the OS was changed to use the Launch Daemon for such tools as at/atrun. The installer and scripts will enable/disable this accordingly by the version of OS and your selection of enable/disable. No verification or manual changes are needed for MacOS X 10.4 and up.

When you initially run the installer you will realize it's a meta installer where it is made up of multiple packages. If you click the "customize" button you will see the list of packages. Of course the PDF Suite is required but you will see everything else is optional. Ghostscript is required for the [PDF_toHTML] tag when converting detailed PDFs. For all normal purposes I recommend just doing the complete install.

NOTE: Due to a bug in Lasso 8.5.5, the Java based tags will throw an error that cannot be trapped, This will only occur if you do not properly use the tag. This issue was not present in LP 8.1. It has been reported and a future update of Lasso should rectify this problem. No future action regarding these tags is needed. Once the bug in Lasso has been fixed, error reporting should work as expected.

[PDF_fromHTML]

ex : [PDF_fromHTML(-URL='http://yourdomain.com/page.lasso',
-Output='web',-SaveAs='myfile.pdf)]

-URL and/or –File

Any multiples of these parameters may be used. Keep in mind that with –File you may only read/write to/from your site's directory. A file of /test.html will be assumed to be in the root of your site.

-Output='xxx'

Below the options for –output are as follows as well as the function of –SaveAs.

'Web'

-SaveAs='somefile.pdf'

This is required when force downloading a PDF to the browser.

[PDF_fromHTML(-file='test.html',-saveas='test.pdf', -output='web')]

In the above example this will grab the file test.html in the root of the site, convert it to a PDF and then force download it in the client's browser.

'Disk'

-SaveAs='somedir/somefile.pdf'

This is required when saving a PDF to your disk.

[PDF_fromHTML(-file='test.html', -saveas='pdf/test.pdf',-output='disk')]

The above example will save the html file as a PDF in the directory "PDF" in the root of the site as "test.pdf". You may save files in any subdirectory in your site's root. There is also an option to save the PDF as a PS file. Simply make the name "something.ps" and it will be saved as a postscript file.

'Printer'

[PDF_fromHTML(-file='test.html',-printer='epsonc80n', -output='printer')]

Requires a valid cups name. See the section on configuring CUPS for a how-to approach.

'Fax'

[PDF_fromHTML(-file='test.html', -fax='efax', -number= '132-4567', -output='fax')]

A valid CUPS Fax name.

ex : -Fax='efax', –Number='1234567890'

If none is provided it will assume efax which is installed with MacOS X 10.3 and up. The use of fax requires –Number for the telephone number. It is up to you to ensure that you have a 1 and / or any area codes necessary. This install includes CUPS back ends for efax.

-Options

These are any of the HTMLDOC options seen on page 8-2 of the accompanying htmldoc.pdf manual. Ex: -Options='-option1 –option2 –option3'

-PS

Using this option while printing will print the file as PS and not PDF.

-Raw

This will return the RAW PDF file to the current page.

NOTES:

These tags are NOT for processing already made PDF files. Watch for an upcoming CUPS tag set to handle all processes of printing.

[PDF_GetInfo]

ex : [PDF_GetInfo(-File='somefile.pdf')]

Required:

-File='somefile.pdf'

This can be any PDF file on your drive. The path to the PDF will be from the root of your site.

Optional:

-Raw

This will make the tag dump raw info to your screen and not use the custom type for getting information. If you want you may format the tags output manually. For example:

The code below will format the text with tabs and all.

<pre>[PDF_GetInfo(-file='some.pdf',-Raw)]</pre>

This code will return the information and put breaks between lines.

[PDF_GetInfo(-file='some.pdf',-Raw,-EncodeBreak)]

This tag declares a custom type thereby allowing you methods of extracting exact info about your PDF. The following are available to you.

Title, Creator, Producer, CreationDate, Tagged, Pages, Encrypted, PageSize, FileSize, Optimized and PDFVersion.

Example:

[var('PDF'=PDF_GetInfo(-file='somefile.pdf'))]

[$PDF->Title] will give you the title of the PDF.

[$PDF->Creator] will give you the creator of the PDF.

[PDF_toHTML]

ex : [PDF_toHTML(-File='somefile.pdf')]

Required:

-File

The PDF file to be used. (path from site's root)

Optional:

-Detailed & -Expire

If the option –Detailed is selected it will create a full mirror of the PDF in html including any graphics and all. This will create a temp folder on your server in the root of the site containing all the html files. The option –Expire='xxx' will control how many MINUTES before the html files are erased. This is controlled by the "at" command at the OS level. If you use the EC tools to disable at for a laptop testing situation be aware that the files will not be erased. You may also specify –Expire=Never and the html files will not be deleted. See –Redirect and –ReturnURL

-SaveAs ex: -SaveAs='xxx'

Should you desire to make the PDF into html and save it with a specific path and name you may do so. Be sure to specify the –Expire to control whether or not the folder is deleted.

-Redirect and –ReturnURL

These are rather unique. If you generate a PDF into html and specify it to be saved then you will more than likely use these options. When creating a detailed html file you need to use a temp folder because there are individually linked files with background images. If you specify either of these after the execution of the tag it will perform that action. So, if you use –Redirect it will redirect the browser to the url which will access these files. If you use –ReturnURL it will give you a url to access them. This may be handy to put into a link.

-FirstPage

The first page of the PDF to start conversion.

-LastPage

The last page of the PDF to stop conversion.

-Silent

Stop any error reporting.

-Exchange

Exchange .PDF links by .html **Advanced Users**

-NoImages

This option is for stopping any images from being included in html.

-NoFrames

This option is for stopping the framed version of a multi page PDF from being converted.

-Zoom

Use this option for zooming in on the PDF. Default is 1.5

-XML

This will output the PDF as xml instead of html.

-HiddenText

Used for displaying any hidden text in the PDF.

-NoMerge

Used to stop merging paragraphs. (Use only if needed)

-BGImageType

This is for selecting a background image format. By default the tag uses jpeg which is fastest but you may use others. These are Ghostscript devices. This should **only** be changed by **advanced** users.

-BGColor

You may use this to change the default background color of html pages generated.

-FBGColor

Change the background color in the links frame in a detailed rendering.

-FBGImage

Set a background image for the links frame in a detailed rendering. This will resolve the link for the image from the root of the site.

-UserPW

The user password of the PDF. You will not need this unless it has been encrypted.

-OwnerPW

The user password of the PDF. You will not need this unless it has been encrypted.

Future Plans:

Plan to add an option –URL where it will download the PDF to a buffer and pump it directly into this tag. Request this feature if you need it to make it into the next release.

Notes:

I was asked why this tag. There is a good reason. Some people don't want to download a PDF or don't have the software to view a PDF. In one example you could use [PDF_fromHTML] to build an archive of monthly reports and then use [PDF_toHTML] for looking at this archive of PDFs at a later date. You could use [File_ListDirectory] to get a list of these archived PDFs and then place them in links. On the response page you plug the PDF name into the [PDF_fromHTML] page and view the report for that month.

[PDF_toTEXT]

ex : [PDF_toText(-File='somefile.pdf',-SaveAs='somefile.txt')]

Required:

-File

The source PDF. This may include the and like all other tags is restricted to the path from the site's root

Optional:

-SaveAs

The text from the PDF may be saved to disk in a text file using this option. This option like all other tags is restricted to the path from the site's root. See NOTES for more information.

-FirstPage

The first page of the PDF to start conversion.

-LastPage

The last page of the PDF to stop conversion.

-EOL

This is for **advanced** users. end-of-line convention (unix, dos, or mac)

-UserPW

The user password of the PDF. You will not need this unless it has been encrypted.

-OwnerPW

The user password of the PDF. You will not need this unless it has been encrypted.

Notes:

If you do not specify an option for saving the file it will return the results to the screen. You may also assign this to a variable.

[var('PDF'=PDF_toText(-file='somefile.pdf'))]

Future Plans:

Plan to add an option –URL where it will download the PDF to a buffer and pump it directly into this tag.

[PDF_SetFieldValue]

ex : [PDF_SetFieldValue(-PDFsrc='myfile.pdf',-SaveAs='mynewfile.pdf',
-field='myfield',-value='myvalue')]

Required :

-PDFSrc

The source PDF.

-Field

The field in the PDF you want to populate.

-Value

The value you want to populate the field with.

Optional :

-SaveAs

If you either specify –Output='disk' or do not specify a target –SaveAs will be the path from the site's root to save the PDF.

If you specify –Output='pipe' then –SaveAs will be the application to pipe the results to. For printing it would look like –SaveAs='lp –d printername' or for faxing

–SaveAs='lp –d efax –o phone=*number'*

If you specify –Output='web' then –SaveAs will be the name to save the PDF as on the client's browser.

-Output

-Output can be 'web', 'disk' and 'pipe'. If 'pipe' is specified -SaveAs is the app to be piped, usually lp, with all the options following. The default is 'disk'.

Optionally "raw" can be used as a target but –Raw is preferred.

-Flatten

This will remove the fields from the PDF but place the text in place of the fields. Some PDF viewers are not capable of viewing PDF with fields in the PDF. So, if you don't need the info to be editable then just use this option.

-Force

This option will simulate the "mkdir –p" in a *nix environment where any missing directories specified in the path will be created before the file is saved.

-Raw

This will option will return the raw PDF data back to the Lasso user rather than saving to disk, stream, fax, print etc.

-InPass

This will provide a password to open the PDF. This applies to both owner and user passwords.

-OutPass

The password to be set on a PDF necessary when opening it in a client's PDF reader.

-OutPassOwner

This is a password to restrict certain editable features of a secured PDF to the owner of it only.

-Security

This is a string of 8 numbers, 0 or 1, to enable a particular feature. The permissions are, in sequence:

AllowPrinting,

AllowModifyContents,

AllowCopy,

AllowModifyAnnotations,

AllowFillIn,

AllowScreenReaders,

AllowAssembly,

AllowDegradedPrinting.

For example the permissions to copy and to print would be

-Security="10100000"

By default, unless specified, all features are enabled.

-KeyLen

This can be "40" or "128". It will be "40" by default.

Notes:

If you have multiple fields in the PDF which you want to populate you may do this my specifying multiple fields and values.

–field='xxx',-value='xxx',-field='xxx',-value='xxx'

This tag like some of the others below are based on iText and have a slightly different syntax than the earlier tags but this will be more standardized in future releases.

[PDF_GetFields]

ex : [PDF_GetFields(-PDFSrc='somefile.pdf')]

Required:

-PDFSrc

The source PDF. As all other tags, paths are resolved from the root of the site.

Notes:

This will read in a PDF and return a list of all fields contained within.

[PDF_MergeFDF]

ex : [PDF_MergeFDF(-PDFSrc='somefile.pdf',
-FDFSrc='somefile.fdf',-SaveAs='mynewfile.pdf')]

Required :

-PDFSrc

The source PDF. (path from site's root)

-FDFSrc

The source to the FDF file. (path from site's root)

Optional :

-SaveAs

If you either specify –Output='disk' or do not specify a target –SaveAs will be the path from the site's root to save the PDF.

If you specify –Output='pipe' then –SaveAs will be the application to pipe the results to. For printing it would look like –SaveAs='lp –d printername' or for faxing –SaveAs='lp –d efax –o phone=*number*' If you specify –Output='web' then –SaveAs will be the name to save the PDF as on the client's browser.

-Output

Options can be 'web', 'disk', 'raw' and 'pipe'. If 'pipe' is selected then -SaveAs is the app to be piped to, usually lp, with all the options following. The default is 'disk'.

-Flatten

This will remove the fields from the PDF but place the text in place of the fields. Some PDF viewers are not capable of viewing PDF with fields in the PDF. So, if you don't need the info to be editable then just use this option.

-Force

This option will simulate the "mkdir –p" in a *nix environment where any missing directories specified in the path will be created before the file is saved.

-Raw

This will option will return the raw PDF data back to the Lasso user rather than saving to disk, stream, fax, print etc.

-InPass

This will provide a password to open the PDF. This applies to both owner and user passwords.

-OutPass

The password to be set on a PDF necessary when opening it in a client's PDF reader.

-OutPassOwner

This is a password to restrict certain editable features of a secured PDF to the owner of it only.

-Security

This is a string of 8 numbers, 0 or 1, to enable a particular feature. The permissions are, in sequence:

AllowPrinting,

AllowModifyContents,

AllowCopy,

AllowModifyAnnotations,

AllowFillIn,

AllowScreenReaders,

AllowAssembly,

AllowDegradedPrinting.

For example the permissions to copy and to print would be

-Security="10100000"

By default, unless specified, all features are enabled.

-KeyLen

This can be "40" or "128". It will be "40" by default.

Notes:

This tag is for reading in a PDF and an FDF file and merging them together into a fully populated PDF.

[PDF_ExportFDF]

ex : [PDF_ExportFDF(-PDFSrc='somefile.PDF',-SaveAs='somefile.fdf')]

Required :

-PDFSrc

The source PDF. (path from site's root)

-SaveAs

The name of the FDF file to be saved. (path from site's root)

-Force

This option will simulate the "mkdir –p" in a *nix environment where any missing directories specified in the path will be created before the file is saved.

-Raw

This will option will return the raw PDF data back to the Lasso user rather than saving to disk, stream, fax, print etc.

Notes:

This tag will read in a PDF and extract any populated form fields and placed them into an FDF form.

[PDF_MakeFDF]

ex : [PDF_MakeFDF(-SaveAs='xxx',-field='xxx',-fieldtype='xxx',-value='xxx',
-field='xxx', -fieldtype='xxx',-value='xxx')]

Required:

-SaveAs The name of the FDF file to be generated and saved.

-Field The name of the field to be specified in the FDF form.

-FieldType The type of FDF field to be generated.

-Value The value of the preceding field name in the FDF form.

-Force

This option will simulate the "mkdir –p" in a *nix environment where any missing directories specified in the path will be created before the file is saved.

-Raw

This will option will return the raw PDF data back to the Lasso user rather than saving to disk, stream, fax, print etc.

Note:

Multiple fields and values can be specified. Example:

-field='xxx',-fieldtype='xxx',-value='xxx',

-field='xxx',-fieldtype='xxx',-value='xxx'

This tag is used for make a FDF file for you with the information you provide to it.

More information can be found on the FDF and PDF formats at http://www.Adobe.com/

Examples by Rich Fortnum

ExecuChoice PDF Generation Tags
by Rich Fortnum, Viaduct Productions
http://www.viaduct-productions.com
July 2003
Somewhere in Toronto
27°C

Steffan's tags introduce a very efficient way of generating PDFs from web pages. This little treatease is a description of how you can implement his new v2.x tags to stream PDFs on the fly without having to generate the file first.

Three Page Concept

This solution follows logic through 3 individual pages. These pages can be incorporated into all solutions, including Corral Method Onepage systems, which is a favorite of mine. As Onefiles also need to carry session information and demand we throw parameters around through links, we'll get into that aspect later. First, let's discuss the Three Page Concept to show how all this works.

Because of the imaging technologies Steffan has brought to Lasso with his tags, we are essentially telling our system to make a PDF of a web page we instruct it to image. That page, we will call Page_3. Understandably, these PDFs have LDML code in them, so they probably change depending on what parameters we send to it, or perhaps from other items in our solution.

Page_1 is the page that has the link to initiate such a solution. It would instruct params thrown to Page_2 (discussed after), which would instruct the system to throw a generated Page_3 at us, streamed as a download. Depending on our browser settings, it will show up in a browser window, or appear as a download.

Page_2, is essentially the hidden instructions, constructed of Steffan's tags, to say what page 3 should include, and the fact that it should be served as a stream, on the fly generation, to the client.

Capice? Let's put some code down.

Page_1 Example

```
<a href="http://g4.local/project/page_2.lasso">Generate PDF</a>
```

This is a link that points to the Page_2. Of the Three File Concept, it is the only page we see. Once we click the link, we get a PDF thrown at us.

Page_2 Example

```
[var: 'myHTMLDOC_Options' = '--landscape --left 5 --right 5 --top 5 --bottom 5
--fontsize 8 --headfootfont Helvetica_Bold --headfootsize 10
--no-compression --no-strict –-no-toc --size Letter']
```

HTMLDOC is the background imaging technology that says how the PDF is generated. Options thrown to HTMLDOC essentially tell the system what we'd like to see in the resulting PDF, such as margins, fonts, header and footer requirements, etc. These 'options' are quite full and flexible, and I recommend you review them in the HTMLDOC PDF that comes with the tags. One thing I should say is that the resulting html page (Page_3) has to fit into the requirements we set within these options, or else you'll be trying to fit a watermelon into a Ziploc bag, and that isn't going to work. Dragging your resulting PDF into BBEdit will show you the errors that you come up with. Reducing table widths might be a solution. When in doubt, modify your Page_3 so that it's a simple page, and at that point, send it to PDF generation to test the system. You can always modify your Page_3 after. In fact, it's good practice that your Page_3 is working before you try forcing a PDF, otherwise you'll think there's something wrong with the tags, when you might be messing up your actual image. Deciphering between the two errors is important, and this way is the best way to troubleshoot.

```
[var: 'myURL' = 'http://g4.local/project/mypage.lasso?showpage=bluejays']
[var: 'myName' = ‘pdf_test.pdf']
```

These vars are built so we can do two things: Modify our options that we have to throw to the tags before required, and so that we can build them dynamically using vars and session information. More on that later. In this Onefile example, bluejays is the instruction to insert the bluejays.lasso file into the body of the Onefile. If you are not using a Onefile system, then it would just be mypage.lasso that you would want to image.

```
[PDF_fromHTML: -options=($myHTMLDOC_Options),
-url=($myURL: -encodenone), -SaveAs=($myName)]
```

So now you see the basic structure of the tag. It's that simple. Options are set so the imaging technology knows what to do, the -URL is essentially Page_3, the page we want to turn into a PDF, and a forced name to save it under. Simple.

Page_3 Example

Well, nothing really to put here. It can be anything. As for now, let's just put a static html page in here. So, this time around, we get the same PDF every time. So now you know how to get a PDF to work.

URL VARS

Since our final Page_3 is one that we'd like to have dynamic information showing, we use Lasso to do this. Inlines could have parameters thrown at it, but we'd have to somehow get params through these 3 files to properly show what we wanted. This is how you do it:

Page_1 has a URL with parameters in it:

```
<a href="http://g4.local/project/page_2.lasso?flavour=
                                        strawberry&scoops=2">Generate PDF</a>
```

Page_2 takes those params, and has to send them to Page_3 so it knows what to do. Again, separate variable building is useful to reduce complexity and confusion for the developer:

```
[var: 'todaysFlavour' = (action_param: 'flavour')]
[var: 'todaysScoops' = (action_param: 'scoops')]
[var: 'myURL'='http://g4.local/project/mypage.lasso?showpage=icecream&
        flavourchoice='  + ($todaysFlavour) +  '&scoopcount=' + ($todaysScoops)]
[var: 'myName' = 'pdf_test.pdf']
```

The URL now has the 3 parameters we throw to Page_3 so that the dynamic LDML page can be generated: showpage so we have the Onefile body called, flavourchoice carries what flavour we wanted from Page_1, and scoopcount carries our integer.

Page_3 receives those parameters, and deals with them accordingly, probably changing those action_params into vars first off, so that inlines can use them as often as they need.

SESSION USAGE

As we develop solutions, we find sessions become really useful. Similar implementation is used here as we did with URL vars usage. Building proper syntax is achieved using the same methodologies, so that we can have a PDF page validated by a session. Carrying the valid current session information from Page_1 through to Page_3 is our goal.

Page_1's link now requires a dynamically built session structure:

```
<a href="http://g4.local/project/page_2.lasso?flavour=strawberry&scoops=2&
        -session=[$Session]:[Session_ID: -Name=($Session)]">
                                        <b>Generate Report</b></a>
```

As you can see, we structure our session as we need to: SESSION_NAME:SESSION_ID. These two parameters come from our session variable information previously assigned for session management. Make sure not to leave out neither the session name, nor the session id. It will confuse you when troubleshooting where the error is coming from, because you will request a page where your session is not valid. Again, testing your Page_3 before asking it to be generated as a PDF, is always good practice. You can accomplish this by aiming your Page_1 URL directly to Page_3, bypassing the request of Page_2 to generate the PDF.

Page_2

The URL var changes to incorporate session structure:

```
[var: 'myURL' = ('http://g4.local/project/mypage.lasso?showpage=
        icecream&flavourchoice=' + ($todaysFlavour) + '&scoopcount=' +
                ($todaysScoops) + '&-session=' + ($Session) + ':' + ($SessionID))]
```

End result is a PDF demanded by something 2 pages away, while throwing params and keeping session structure integrity intact.

Good luck with the tags, and let Steffan know what you think of his tags.

Configuring CUPS for printing and faxing.

The efax printer queue has already been set up by the installer but you will need to figure out the name of the printer. There are a couple of ways. You can do it through the Mac OS X Print Center or directly thru CUPS (http://www.cups.org/) the built in print spooler. You will need the printer name for the [PDF_fromHTML] tag to know where to send the document. Don't be fooled by the name of this tag when used for printing. The pages must be converted to PDF before being printed or faxed.

To verify printer name via the Printer Setup Utility (Panther):

Go into the /Applications/Utilities/ folder and open the app called Print Center. If the application does not present a window with a list of printers in there then go to the "Printers" menu and select "Show Printer List". In the window you should see the new "efax" printer and any other printers you already have configured. Let's say that you have a printer called "Foo" then in the [PDF_fromHTML] tag you would have -Printer='Foo'. This should be pretty simple. I have enabled SMB printer sharing upon install of these tags so you can even print directly to printers attached directly to a windows machine! If you just click add you will not have the option to add a SMB printer but do it this way :

1. Hold down the option key while clicking the Add button in the print center.
2. Select Advanced
3. From the device drop down select Windows Printer via Samba
4. Enter a name for the printer (this will also be used in -printer you can have as many printers attached as you want!)
5. In the device URI you have a few ways to enter this. If there are no passwords used then you will have a URI like this "smb://username@machinename/printername" Where you see "username" do not change that to anything. Leave it as it is. On your MAC OS X 10.2+ machine open this URL http://127.0.0.1:631/help.html and read it for more information regarding this.
6. Select a printer model. By a default install of Mac OS X 10.2 there really are FEW choices for printers so I have included the latest GIMP-Print drivers in the isntaller. You now have over 300 options on PPD drivers and there are some really cool ones in there! More can be found at http://www.linuxprinting.org .
7. When done with that click Add and then you are ready to go!

To verify by using CUPS (the built in print spooler) :

1. Open the following URL in a browser http://127.0.0.1:631/printers
2. Here you will see a list of all your printers.
3. Choose one and remember the name so you can use it in the [PDF_fromHTML] tag when printing.

There is a great deal of documentation about CUPS there for you to read. Its really cool knowing that you can print to ANY printer! Hey, you could even print directly to mine!

More info on CUPS : http://127.0.0.1:631/help.html

Included Software and Source Locations

HTMLDOC is from Easy Software -

http://www.easysw.com/htmldoc

Components from XPDF

http://www.foolabs.com/

iText by Paulo Soares

http://sourceforge.net/projects/itext http://itextpdf.sourceforge.net/

http://www.lowagie.com/iText/

GIMP-Print

http://gimp-print.sourceforge.net

ESP Ghostscript

http://sourceforge.net/projects/espgs and http://www.cups.org

All source code for HTMLDOC, XPDF and GhostScript are included with this distribution.

Updates to the whole PDF Suite are of course available at

http://www.execuchoice.net/

Disclaimer

This software is to the best of my knowledge bug free and has been thoroughly tested. There is NO warranty or guarantee of any kind implied or written. I the author/company will NOT be responsible for any loss of income or damages due to the usage of ANY of this software. Use at your own risk!

General Licensing Information

The software in the section *"Included Software and Source Locations"* falls under the GNU license agreement with the exception of HTMLDOC which has it's own license as seen on page 36. By using this software you are bound to all the licenses and agreements described in this manual.

Included within this package are the modules – EC_Helper.dylib, EC_PDFSuite.class and EC_PDF_Suite_V30_LP85.LassoApp which are CLOSED source and the source is NOT available. They are intended to be distributed with this package only. Reverse engineering of any of these tools is strictly prohibited and will be considered a violation of the simple license agreement. Use and distribute this package in its entirety as needed.

If you come across any problems I will be happy to look at it. Just email me and I will help as best I can.
Steffan@ExecuChoice.net

Special Thanks to :

Pier Kuipers for his work on the new PDF cover.
Chris Buxton for his EXTENSIVE assistance with the included scripts.
Marc Pope for the graphics, icons and images used in the installer.
Rich Fortnum for his assistance in writing the tech article showing usage of the tags.
Kyle Jessup for his C/C++ assistance in programming of the C backend applications.

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place - Suite 330, Boston, MA 02111-1307, USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

* a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

* b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

* c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

* a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

* b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

* c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.) The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies,

or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

HTMLDOC License Agreement

Copyright 1997-2008 by

Easy Software Products
44141 AIRPORT VIEW DR STE 204
HOLLYWOOD, MARYLAND 20636-3111 USA

Voice: +1.301.373.9600
Email: cups-info@cups.org
WWW: http://www.cups.org

End-User License Agreement

PLEASE READ THIS DOCUMENT CAREFULLY. THIS IS A LEGAL AGREEMENT BETWEEN YOU AND EASY SOFTWARE PRODUCTS ("ESP"). BY INSTALLING THIS PACKAGE, AND USING THE HTMLDOC SOFTWARE AND DOCUMENTATION ("SOFTWARE") CONTAINED IN THIS PACKAGE, YOU ARE AGREEING TO BECOME BOUND BY THE TERMS AND CONDITIONS OF THIS AGREEMENT. THIS AGREEMENT REPRESENTS THE ENTIRE AND SOLE AGREEMENT CONCERNING THE ESP SOFTWARE, AND IT SUPERSEDES ANY PRIOR PROPOSAL, REPRESENTATION OR UNDERSTANDING REGARDING THE SOFTWARE BETWEEN YOU AND ESP OR ANY OTHER PARTY. IF YOU (hereafter in this document referred to as "LICENSEE") ARE NOT WILLING TO BE BOUND BY THE TERMS OF THIS LICENSE AGREEMENT, PLEASE DO NOT INSTALL OR USE THIS SOFTWARE.

TERMS AND CONDITIONS OF SOFTWARE LICENSE

GRANT OF LICENSE; USE RESTRICTIONS. In consideration for the payment of a license fee, ESP grants to LICENSEE a personal, nontransferable (except as provided below) and nonexclusive right to use the SOFTWARE on one (1) computer system, solely for LICENSEE's internal business purposes. LICENSEE agrees that it shall not reverse compile or disassemble any portion of the SOFTWARE not already provided by ESP in source code form.

1. **OWNERSHIP OF SOFTWARE**. LICENSEE agrees that no title to the SOFTWARE, or the intellectual property in any of the SOFTWARE, or in any SOFTWARE copy, is transferred to LICENSEE, and that all rights not expressly granted to LICENSEE hereunder are reserved by ESP. This license is not a sale of the original SOFTWARE or any copy thereof.
2. **TRANSFER RESTRICTIONS**. If LICENSEE transfers ownership, or otherwise disposes, of a computer system for which a license for the SOFTWARE was purchased, provided that the transferee agrees to accept the terms and conditions of this AGREEMENT, LICENSEE may transfer the SOFTWARE and all licenses and rights in SOFTWARE granted under this AGREEMENT to such transferee, provided that all SOFTWARE copies are also transferred.
3. **DISCLOSURE RESTRICTIONS**. LICENSEE agrees to not disclose or otherwise disseminate software licensing information, including the so-called "license key" provided to LICENSEE by ESP, to third parties other than ESP or its official distributors. Should LICENSEE violate this restriction, LICENSEE shall comply with the termination clause of this license and pay a penalty fee of $1000 US per offense or $100,000 US, whichever is greater. This restriction does not apply if the license is

being transferred according to the rules in paragraph 3.
4. **TERMINATION**. If licensee fails to fulfill any of LICENSEE's material obligations under this AGREEMENT, ESP may, at any time thereafter, and in addition to any other available remedies, terminate this AGREEMENT and all licenses and rights granted to LICENSEE under this AGREEMENT. Upon termination of this AGREEMENT, LICENSEE shall, within thirty (30) days after termination, deliver to ESP all removable media and documentation containing the SOFTWARE, and shall render unusable all copies of the SOFTWARE placed in any storage apparatus.
5. **GOVERNING LAW**. This License is governed by the laws of the State of Maryland, U.S.A., excluding choice of law rules. If any part of this License is found to be in conflict with the law, that part shall be interpreted in its broadest meaning consistent with the law, and no other parts of the License shall be affected.
6. **U.S. GOVERNMENT USERS**. This SOFTWARE is provided with RESTRICTED RIGHTS. If you are a unit or agency of the United States Government or are acquiring the Software for any such unit or agency, the following applies:
If the unit or agency is the Department of Defense ("DOD"), the SOFTWARE and its documentation are classified as "commercial computer software" and "commercial computer software documentation" respectively and, pursuant to DFAR Section 227.7202, the Government is acquiring the SOFTWARE and its documentation in accordance with the terms of this License. If the unit or agency is other than DOD,
7. *End-User License Agreement A-3*
the SOFTWARE and its documentation are classified as "commercial computer software" and "commercial computer software documentation" respectively and, pursuant to FAR Section 12.212, the Government is acquiring the SOFTWARE and its documentation in accordance with the terms of this License.

LIMITED WARRANTY AND DISCLAIMER OF WARRANTY; LIMITATION OF LIABILITY

ESP warrants that it has the authority and right to license the SOFTWARE, and that the SOFTWARE will conform to the material printed specifications therefore which are in effect on the date of original delivery of such SOFTWARE. ESP's warranty and obligation shall extend for a period of ninety (90) days after the date of the original delivery of SOFTWARE to LICENSEE by ESP, and is solely for the benefit of LICENSEE, who has no authority to assign or pass through this warranty to any other person or entity. Except as provided in this Section, the SOFTWARE is provided to LICENSEE on an "AS IS" basis, and ESP makes no other warranty of any kind, express or implied, with regard to the SOFTWARE licensed hereunder. ESP warrants that the SOFTWARE properly processes date and time information between the years 1970 and 2038. ESP does not warrant or represent that the SOFTWARE will operate uninterrupted or error free or that all defects in the SOFTWARE are correctable or will be corrected. This warranty shall not apply if SOFTWARE is used other than in accordance with ESP's written instructions, or if any of LICENSEE's hardware equipment or other software malfunctions. ESP's entire liability and LICENSEE's exclusive remedy for any defects in the SOFTWARE shall be to obtain ESP's SOFTWARE updates via the Internet or pay a media update fee for each copy of the SOFTWARE. THE FORGOING WARRANTIES ARE IN LIEU OF, AND ESP DISCLAIMS, ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow the exclusion of implied warranties, so the exclusion above may not apply to

LICENSEE. This warranty gives LICENSEE specific legal rights, and LICENSEE may have other rights which vary from state to state. IN NO EVENT SHALL ESP OR ESP'S LICENSORS BE LIABLE TO LICENSEE FOR ANY SPECIAL, CONSEQUENTIAL, INCIDENTAL OR INDIRECT DAMAGES OF ANY KIND (INCLUDING WITHOUT LIMITATION LOSS OF PROFITS OR DATA AND PERSONAL INJURY), WHETHER OR NOT ESP HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH LOSS, HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, ARISING OUT OF THIS AGREEMENT. THESE LIMITATIONS SHALL APPLY NOTWITHSTANDING THE FAILURE OF THE ESSENTIAL PURPOSE OF ANY LIMITED REMEDY. In no event will ESP be liable for any claim against LICENSEE by a third party, and LICENSEE hereby agrees to indemnify and hold ESP harmless for any claims for cost, damage, expense or liability arising out of or in connection with the installation, use and performance of the SOFTWARE licensed hereunder, whether alone or in combination with any other product or service. Some states do not allow the limitation or exclusion of liability for incidental or consequential damages, so the limitation above may not
apply to LICENSEE.