


WIN-
PROLOG

4.900

WebFlex Server User Guide

by Alan Westwood

WebFlex Server User Guide

The contents of this user guide describe the product WebFlex Server, and are believed correct at the time of going to press. They do not embody a commitment on the part of Logic Programming Associates (LPA), who may from time to time make changes to the specification of the product, in line with their policy of continual improvement. No part of this manual may be reproduced or transmitted in any form, electronic or mechanical, for any purpose without the prior written agreement of LPA.

Copyright (c) 2004 Logic Programming Associates Ltd. All Rights Reserved.

Logic Programming Associates Ltd.
Studio 30
The Royal Victoria Patriotic Building
Trinity Road
London
SW18 3SX
England

phone: +44 (0) 20 8871 2016
fax: +44 (0) 20 8874 0449
web site: <http://www.lpa.co.uk/>

WebFlex Server, ProWeb Server, **LPA-PROLOG** and **WIN-PROLOG** are trademarks of LPA Ltd., London England.

19 July, 2004

Contents

WebFlex Server User Guide	2
Contents	3
Chapter 1 – WebFlex	5
Introduction	5
Chapter 2 – Installing and Setting-Up WebFlex	6
Introduction	6
The WebFlex Directory	6
HTTP Server	7
Copying The WebFlex Directory	7
Copying The ProWeb Files	7
Copying The FLEX.PC File	9
Setting Up The Aliases	10
Viewing The WebFlex Launch Pages	12
Changing WEBFLEX.INI Settings	13
Creating a Temp Directory	14
Running WebFlex	14
Trouble Shooting	15
Chapter 3 – Using WebFlex	16
Introduction	16
INI File Settings	16
[general]	16
[name]	16
Applicable settings	16
General behaviour	17
Example HEAD.HTM, BODY.HTM and FOOT.HTM	17

Introduction	4
Questions	18
Styles.....	19
Group Styles.....	19
Question Styles.....	20
Because and Help.....	21
Problems.....	21
Chapter 4 – WebFlex Demos.....	22
Introduction	22
Robbie	23
Solvents	24
Forest Yield	25
Insurance.....	26
Fuzzy Car	27
Pesticide.....	28
Timetable.....	29
Holidays.....	30
Forms	31
Index	32

Chapter 1 – WebFlex

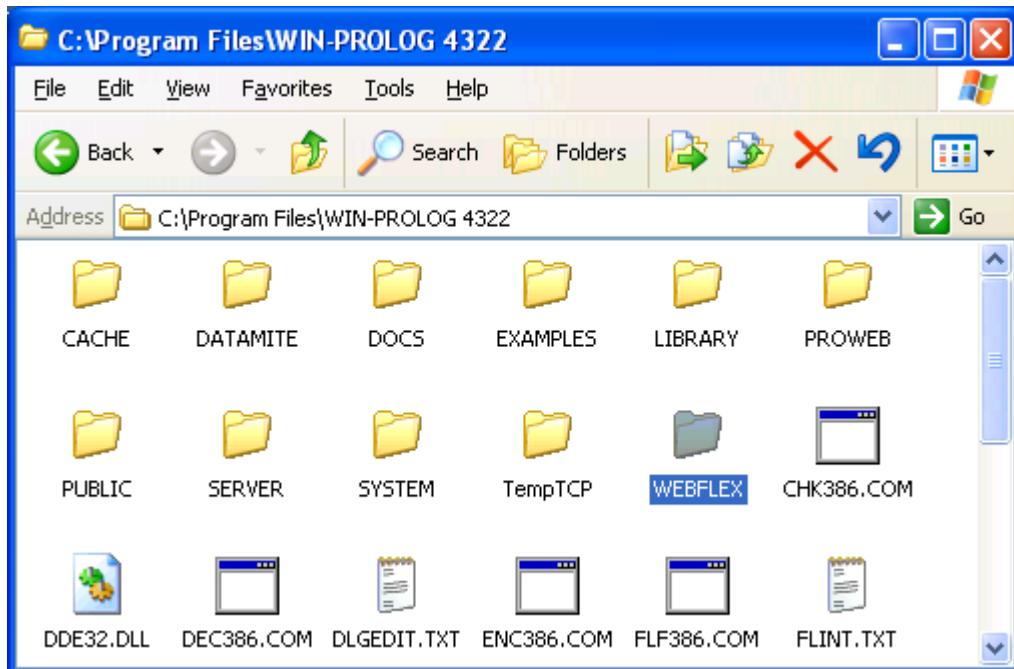
Introduction

WebFlex is an Internet enabled version of the flex expert system toolkit. The LPA web-based expert system, WebFlex, is basically an instance of a ProWeb application.

Chapter 2 – Installing and Setting-Up WebFlex

Introduction

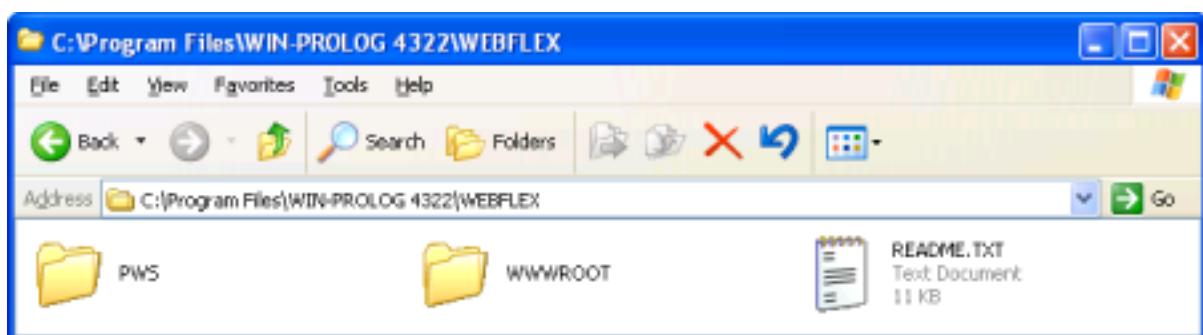
The LPA Setup program will install the WebFlex files. The WebFlex files have been installed if you have a WebFlex directory within your **WIN-PROLOG** directory.



The rest of this chapter will go through the manual setting-up process.

The WebFlex Directory

The WebFlex directory consists of two subdirectories - PWS and WWWROOT.

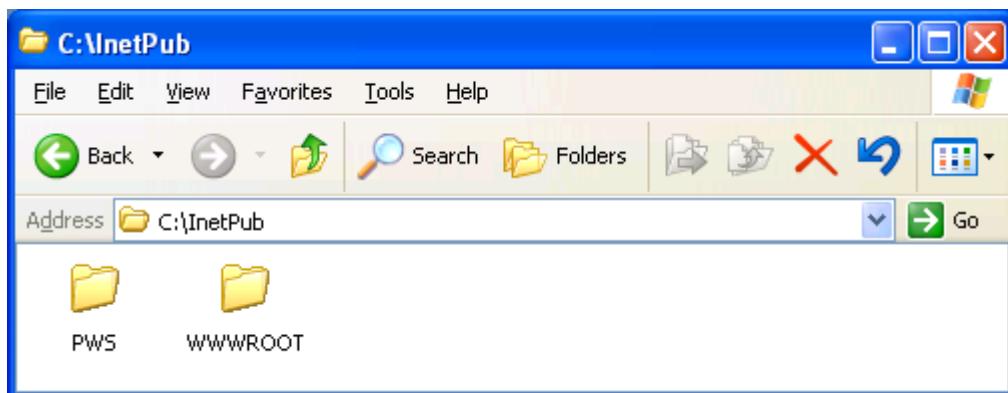


HTTP Server

You will need an HTTP server, such as Microsoft's IIS, installed. The rest of this chapter assumes that you are using IIS and that the main directory is C:\INETPUB and the home directory is C:\INETPUB\WWWROOT.

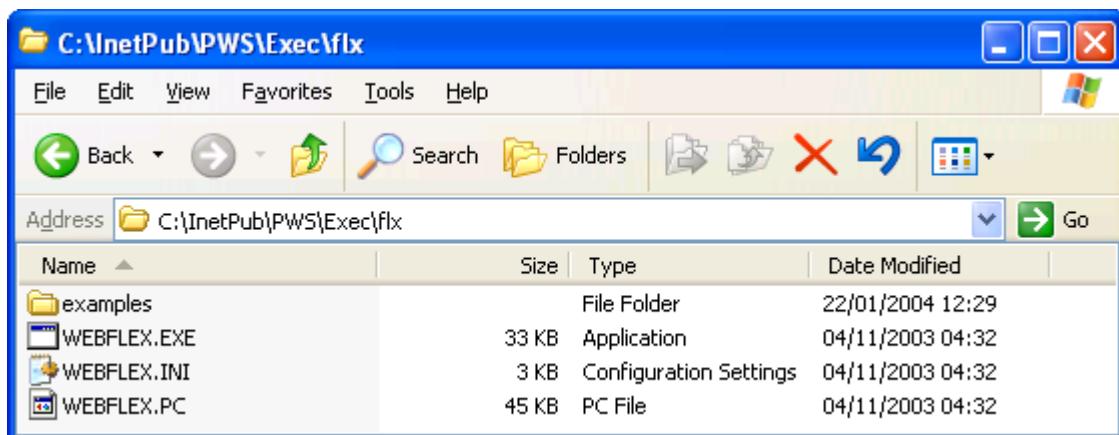
Copying The WebFlex Directory

You need to copy both the PWS and WWWROOT directories to your C:\INETPUB directory. You may already have a WWWROOT directory.

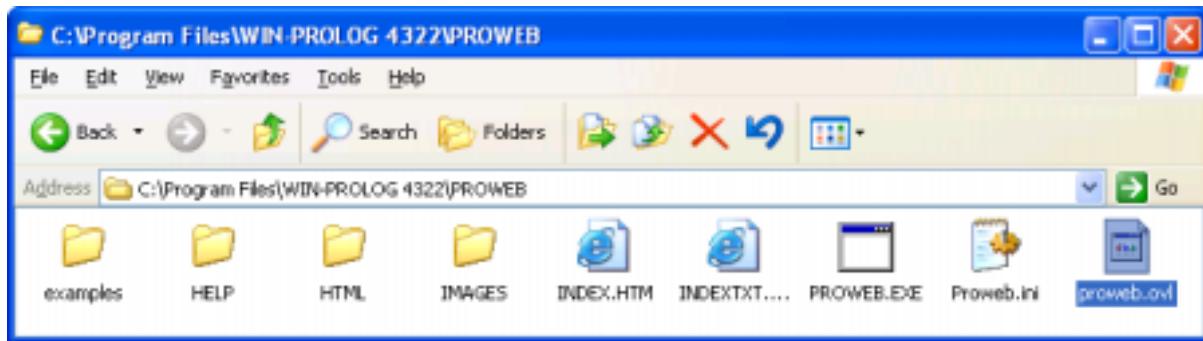


Copying The ProWeb Files

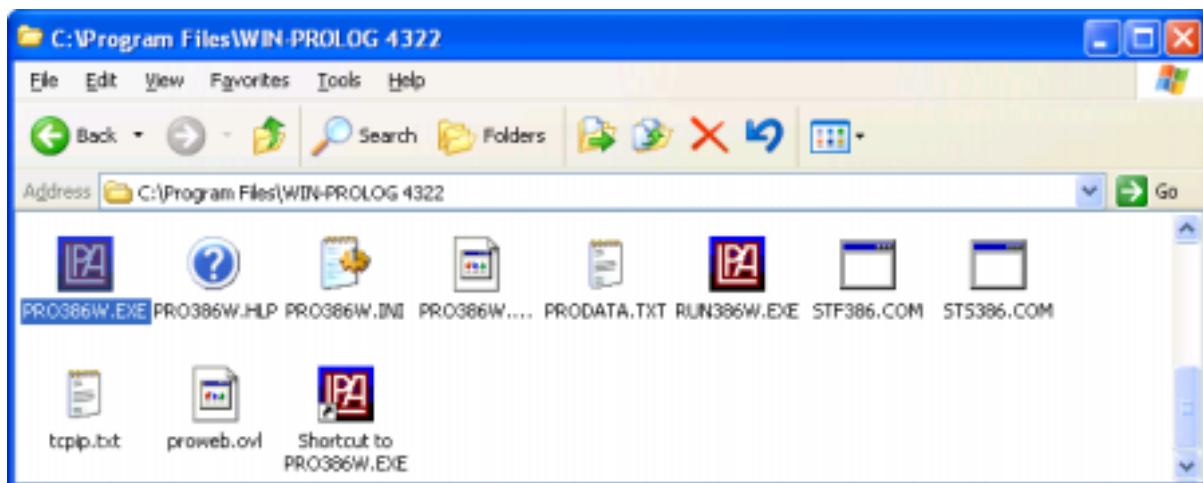
You now need to copy two files into the C:\INETPUB\PWS\EXEC\FLX directory.



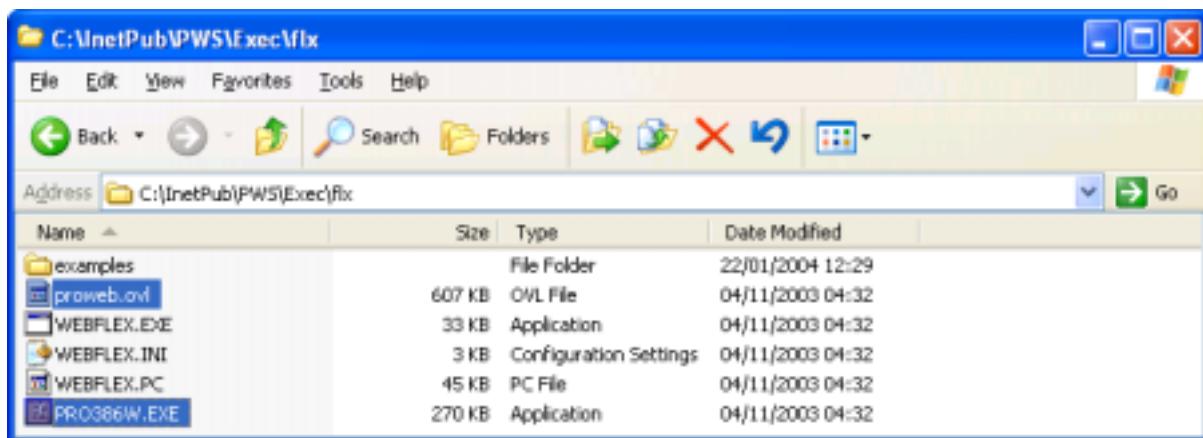
Firstly, you need to copy PROWEB.OVL from your C:\PROGRAM FILES\WIN-PROLOG 4420\PROWEB directory and place it into the C:\INETPUB\PWS\EXEC\FLX directory.



Secondly, you need to copy PRO386W.EXE from your C:\PROGRAM FILES\WIN-PROLOG 4420 directory and place this too into the C:\INETPUB\PWS\EXEC\FLX directory.



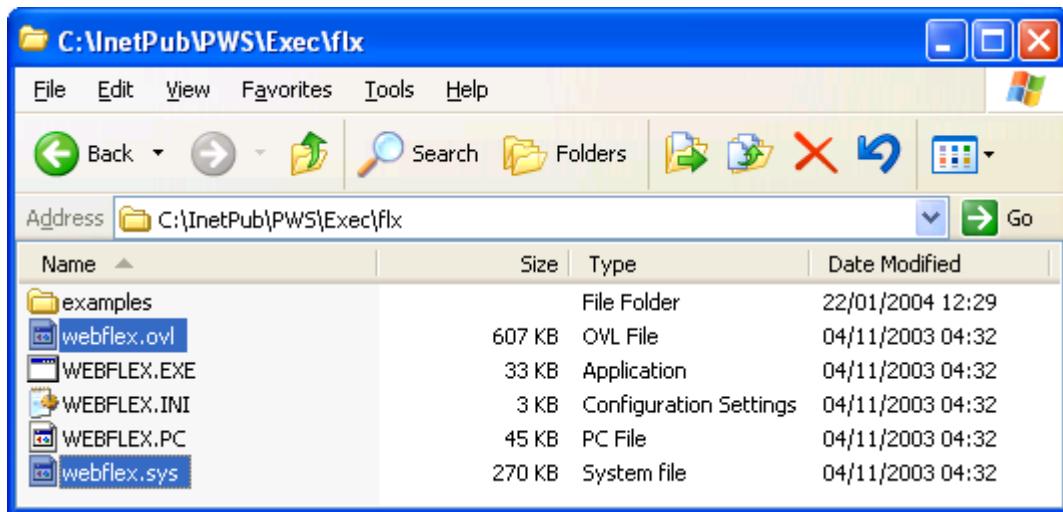
Your C:\INETPUB\PWS\EXEC\FLX directory should now look like this:



Thirdly, you need to rename PRO386W.EXE in the C:\INETPUB\PWS\EXEC\FLX directory to WEBFLEX.SYS.

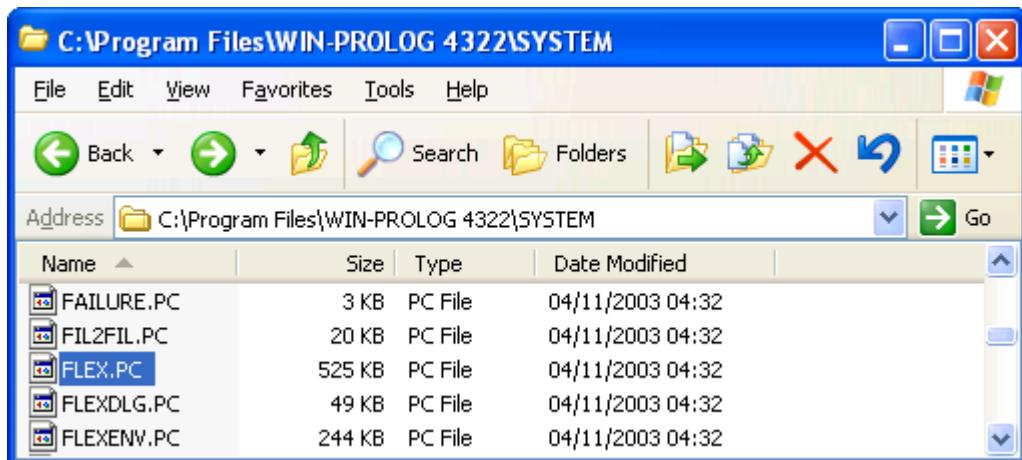
Fourthly, you need to rename PROWEB.OVL in the C:\INETPUB\PWS\EXEC\FLX directory to WEBFLEX.OVL.

Your C:\INETPUB\PWS\EXEC\FLX directory should now look like this:

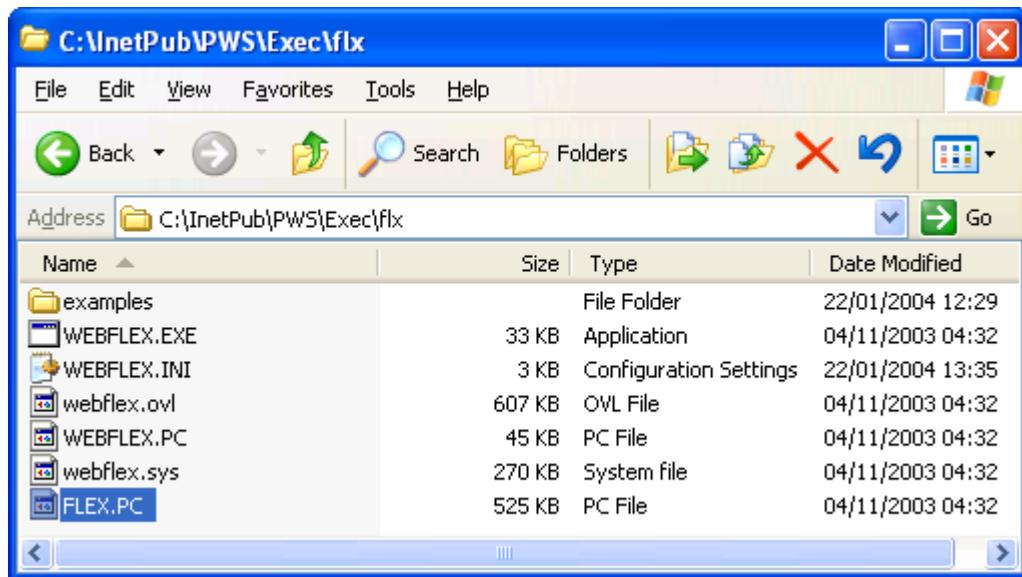


Copying The FLEX.PC File

You now need to copy FLEX.PC from the C:\PROGRAM FILES\WIN-PROLOG 4420\SYSTEM directory and place it into the C:\INETPUB\PWS\EXEC\FLX directory.



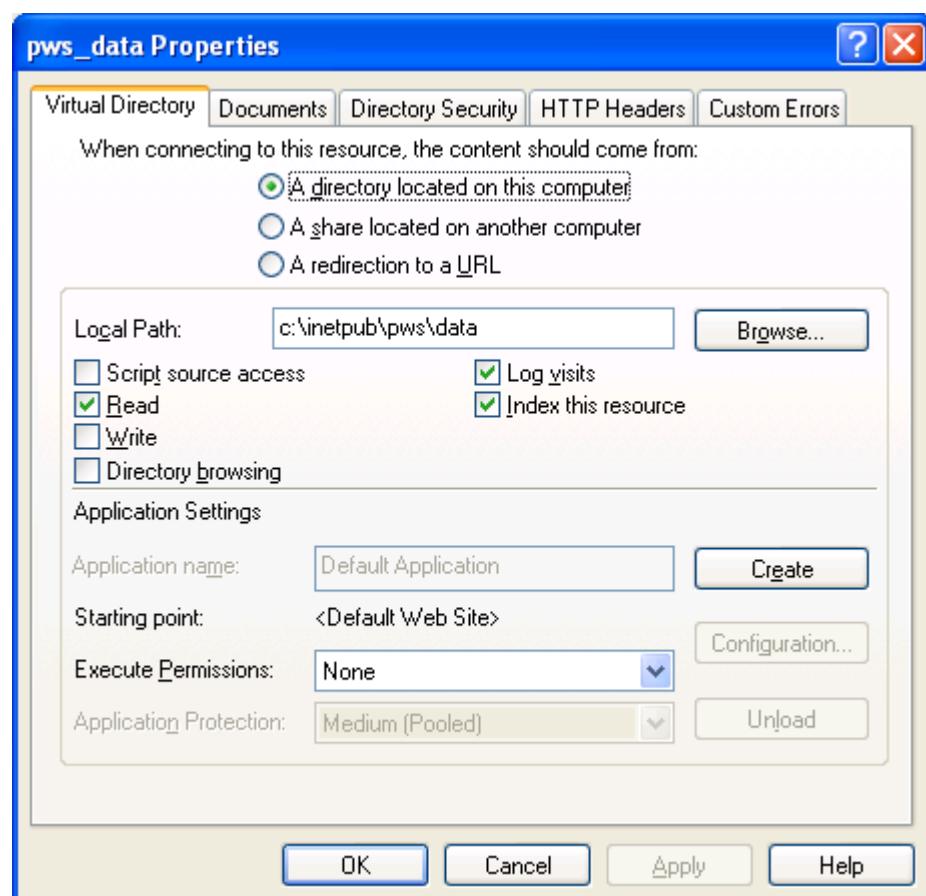
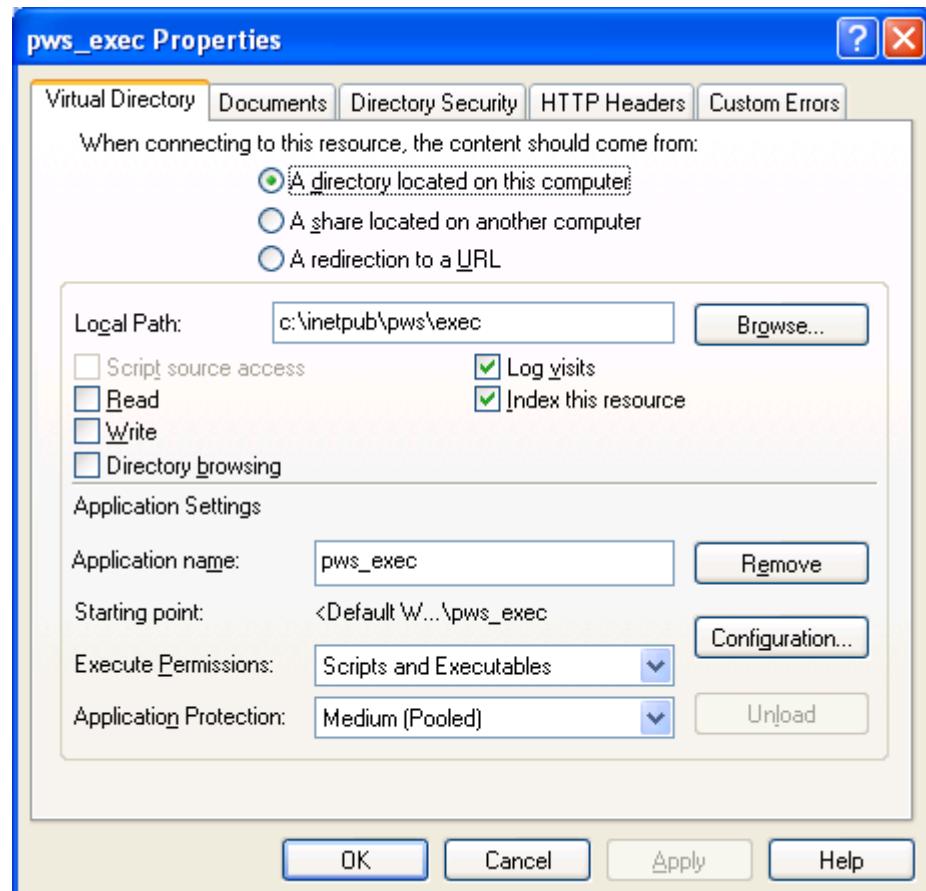
Your C:\INETPUB\PWS\EXEC\FLX directory should now look like this:



Setting Up The Aliases

The files making up WebFlex are split over three distinct areas, each requiring its own alias to be set up within Personal Web Server or IIS.

Directory	Alias Name	Permissions
C:\INETPUB\WWWROOT	This is IIS's 'home' directory	'read'
C:\INETPUB\PWS\EXEC	pws_exec	'execute'
C:\INETPUB\PWS\DATA	pws_data	'read'

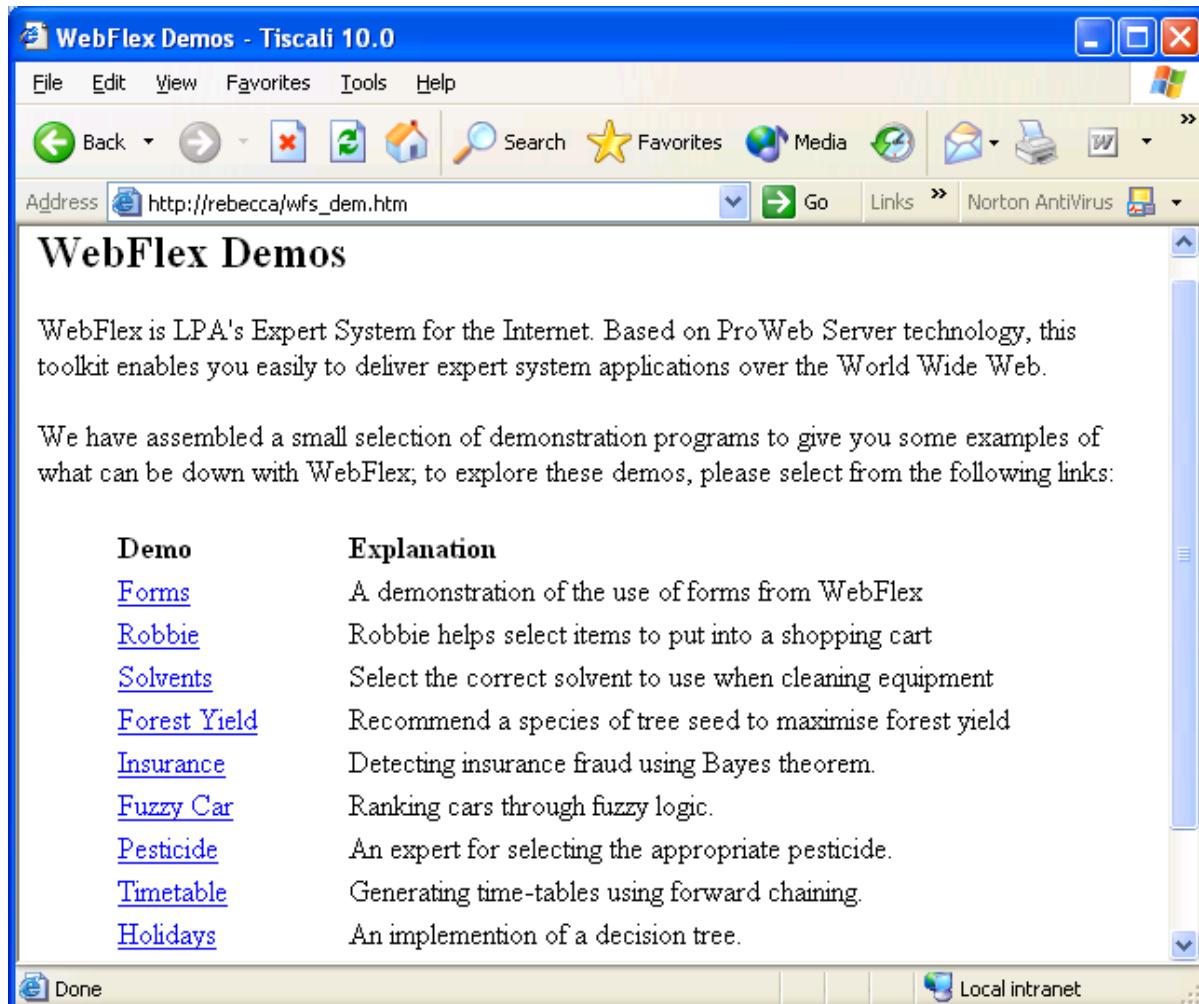


Viewing The WebFlex Launch Pages

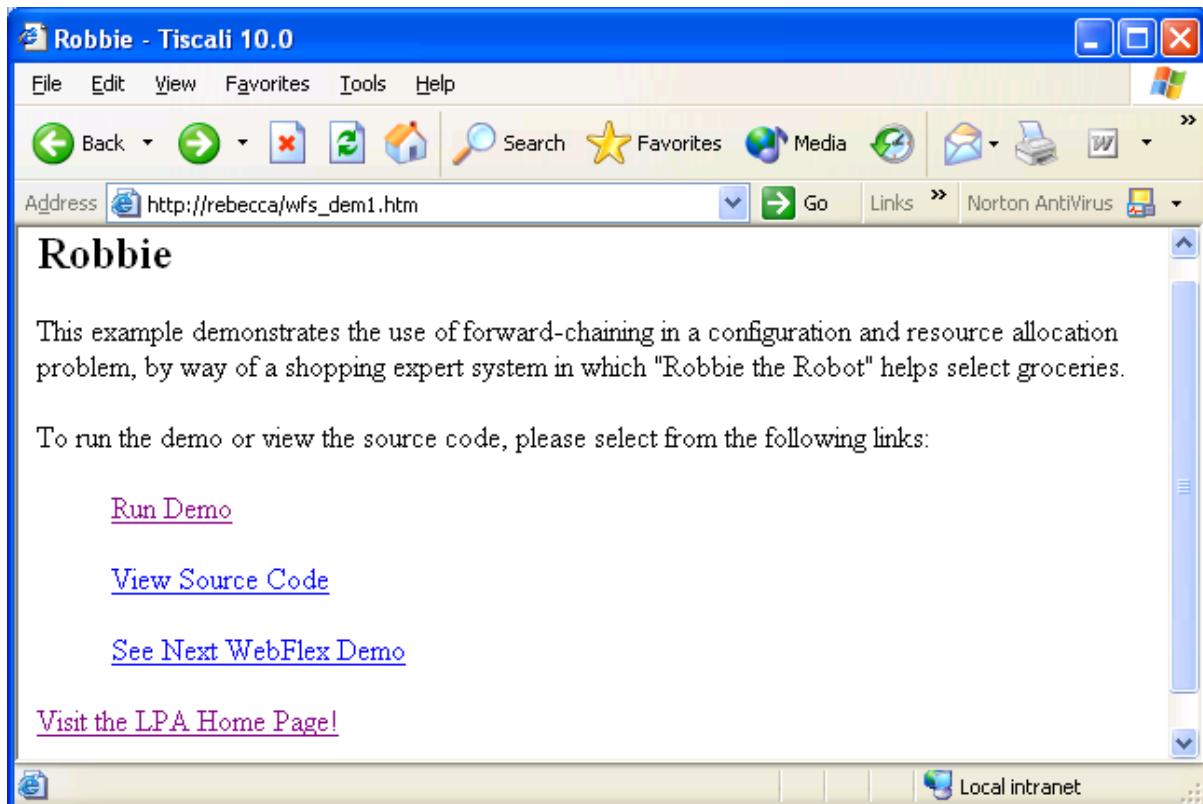
Go into your web browser and execute the following URL:

`http://<computer_name>/wfs_dem.htm`

replacing '<computer_name>' with the name of your computer.



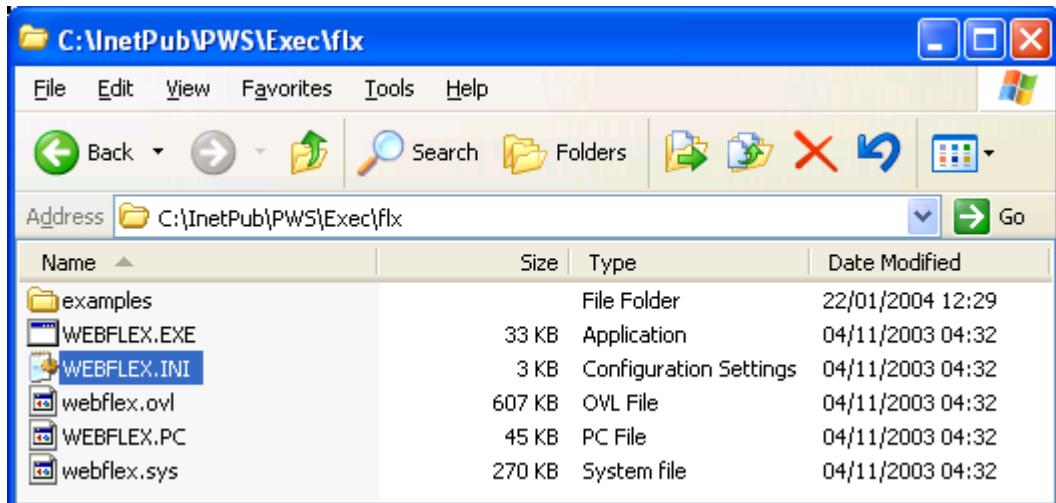
Click on the 'Robbie' hyperlink:



Before we can click on the 'Run Demo' hyperlink, we need to modify WEBFLEX.INI.

Changing WEBFLEX.INI Settings

You now need to modify some of WebFlex's settings within the WEBFLEX.INI file. This file can be found in the C:\INETPUB\PWS\EXEC\FLX directory:

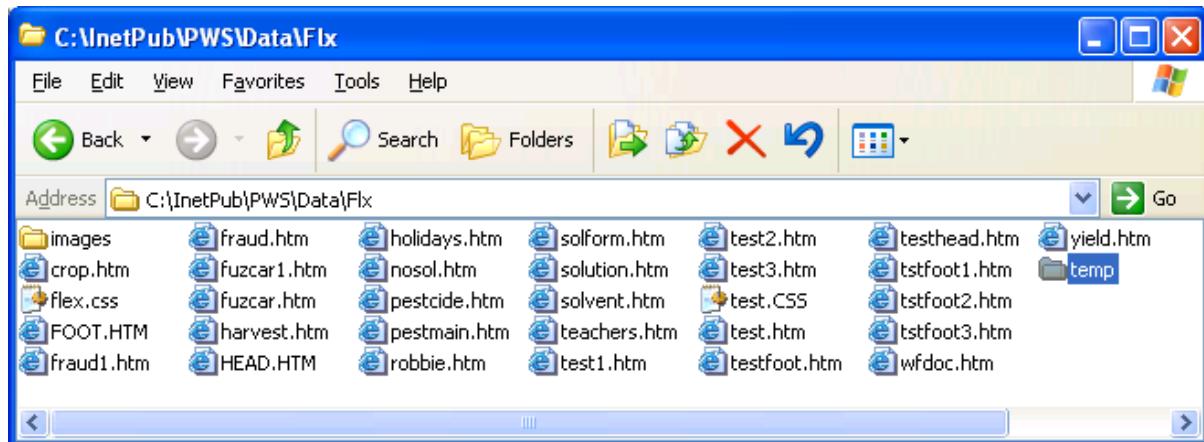


Load WEBFLEX.INI into a text editor (e.g. Notepad) and modify the following settings accordingly:

Original Setting	Change To	Example
TEMP_PATH=c:\inetpub\pws\data\flx\temp\	Leave as supplied but see next section about creating a TEMP directory.	-
BASE_URL=http://mysite/pws_exec/flx/	BASE_URL=http://<computer_name>/pws_exec/flx/	BASE_URL=http://rebecca/pws_exec/flx/
TEMP_URL=http://mysite/pws_data/flx/temp/	TEMP_URL=http://<computer_name>/pws_data/flx/temp/ See next section about creating a TEMP directory.	TEMP_URL=http://rebecca/pws_data/flx/temp/

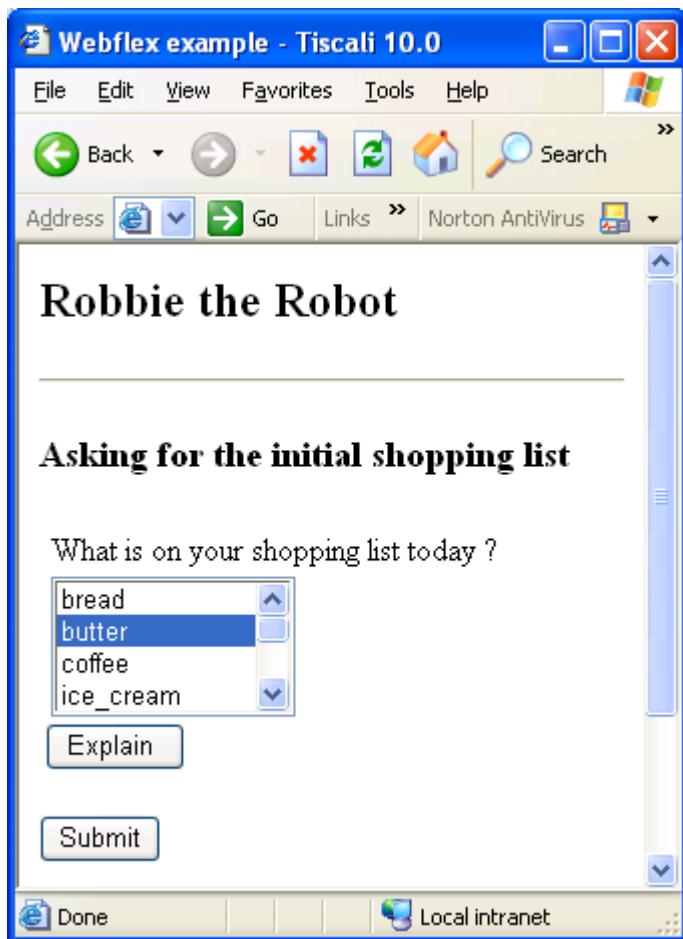
Creating a Temp Directory

Two of the WEBFLEX.INI settings above require a TEMP directory to be present in the C:\INETPUB\PWS\DATA\FLX directory. Create a new directory called TEMP within the C:\INETPUB\PWS\DATA\FLX directory:



Running WebFlex

You are now in a position to return to your web browser window and click on the 'Run Demo' hyperlink and run WebFlex's Robbie The Robot example:



Trouble Shooting

HTML Page Displayed	Solution
ProWeb Server Copyright (c) 1996-2003 Logic Programming Associates Ltd 4.31 - 03 Mar 03	You have forgotted to copy the FLEX.PC file.
Main program terminated with an error! Error #31 File Not Found ensure_loaded / 1	
System Error (OSE:001:0) Please try again...	You need to create a TEMP directory and point the WEBFLEX.INI settings correctly to it.

Chapter 3 – Using WebFlex

Introduction

WebFlex allows flex programs to run through a browser, to do this it generates HTML forms for the questions it needs to ask. Indeed most flex programs will run unchanged immediately through WebFlex, though they will look rather plain. Unlike previous versions of flex several questions can be asked at once, obviously this is important for a system that is generating HTML pages to send to a browser.

INI File Settings

There are a number of settings contained in the WEBFLEX.INI file to control webflex.

[general]

This section contains settings that are generally applicable.

The path to the flex KSL files can be specified here (it defaults to examples) the path should be named as:

```
scripts_directory=path
```

where path is a relative directory to where WEBFLEX.SYS (PRO386W.EXE) lives.

For example scripts_directory could be set to the directory KSL which is a sub-directory of where WEBFLEX.SYS lives, by the entry:

```
scripts_directory=ksl
```

The KSL files for your application will be picked up from this path.

[name]

A NAME section corresponds to a NAME.KSL file in the examples directory and applies to that. Settings here override settings in the general section.

Applicable settings

main_goal=my_main_goal	This specifies the main goal to run for a given example; this is not applicable to the general section . All other settings can be in the general section.
------------------------	--

explain_text=Explain	The text for an explain button.
----------------------	---------------------------------

help_text=Help	The text for an help button.
----------------	------------------------------

submit_text=OK	The text for an OK button.
question_seperator=br	The HTML code used to separate questions (could be hr)
explain_image=	An image URL to display instead of the explain button.
help_image=	An image URL to display instead of the help button
submit_image=	An image URL to display instead of the submit button

The following three settings can also be overridden in the style definition of a question group to allow individual pages for each group of questions.

header=testhead.htm	The head part of the HTML page
body=test.htm	The body part of the HTML page
footer=testfoot.htm	The foot part of the HTML page

General behaviour

For each example a header page, a body page and a foot page can be defined together with the output from the flex program up to the question(s) being asked and the form containing the question(s) constitute one page to be sent to the browser.

Example HEAD.HTM, BODY.HTM and FOOT.HTM

HEAD.HTM:

```
<HTML>
<HEAD>
<TITLE>Webflex example</TITLE>
<LINK REL="stylesheet" TYPE="text/css" HREF="/white.css">
</HEAD>
```

BODY.HTM:

```
<BODY>
<H2>Robbie the Robot</H2>
<HR>
```

WebFlex output will be inserted here.

FOOT.HTM

```
<P>
<A HREF="/wfs_dem.htm">Back to WebFlex demos</A><P>
<A HREF="/ind_top.htm" TARGET="_top">Visit the LPA Home
Page!</A><P>
</BODY>
</HTML>
```

Questions

The basic user input to flex is through questions, webflex allows flex questions to be grouped together, and for individual questions to have style information associated, and for question groups to have style information associated. The way the styles are associated to questions and groups is through the frame hierarchy by a naming convention. Say you have the question foo, then by defining a frame `foo_style` you can add slots to specify various things.

The ask mechanism allows for the following use:

ask groupname,

will ask the questions defined in the group `groupname` :

```
group group_test1
  multi_choice, single_choice_radio, single_choice_listbox,
  single_choice_dropdown.
```

Ask can also ask several questions (or one!) as:

ask q1,q2,q3

The difference is that by using a group you can change the appearance of the containing table, through the group style frame:

```
frame group_test1_style
  default caption is 'Form 1' and
  default body is 'test1.htm' and
  default footer is 'tstfoot1.htm' and
  default tablestyle is { bgcolor-'#FFccFF', border-1 } .
```

or for a question:

```
question single_choice_radio
  A single-choice radio-button question ;
  choose one of choices
  because I need to test a single-choice radio-button question .
```

```
frame single_choice_radio_style ;
default method is radio and
default prefill is apples and
default infix is br .
```

All output from flex up to the ask will be prepended to the form in the page, so for example:

```
...
and write( '<h1>hi there</h1>' )
and ask group_test1
...
```

will write out the '<h1>hi there</h1>' before the table containing the group_test1 form, notice that it is written out as HTML so if you want to use characters such as > then you will have to write the HTML equivalent, in this case > as:

and write('<h1>hi there 3 > 2</h1>').

Also notice that / is doubled up this is because flex uses / as an escape character.

Any output after the last question in a program is executed is sent to the final results page.

Styles

The following is a list of styles for various questions and groups. All of the styles are optional and default to various settings.

Group Styles

caption	The caption for the containing table.
columns	The number of columns in the table.
header	A header HTM file for use with this form.
body	A body HTM file for use with this form.
footer	A footer HTM file for use with this form.
tablestyle	A set that defines the style of the containing table of the form { attribute-Value, ... } see your HTML documentation for attributes applicable to a table, (cellpadding etc.).

Question Styles

multiple choice styles

method	Set the method, options are: {checkboxes,listbox(default) }
rows	The number of rows in the listbox.
prefill	A set of items on the list to preselect.

single choice styles

method	Set the method options are: { radio, listbox(default) }
rows	The number of rows in the listbox if omitted the single choice listbox becomes a drop-down list, only applicable if the method is not radio.
prefill	An item on the list to preselect
infix	Infix HTML between each item br or hr say, only applicable if the method is radio.

input styles

prefill	Text to prefill the text box.
maximum_length	Maximum number of characters to allow.
cols	The number of columns for the text box.
rows	The number of rows for the text box.
lower_bound	The least value for a number input field.
upper_bound	The greatest value for a number input field.

Styles can also be passed down by inheritance, for example if you want to apply the style 'method is radio' to several fields you can use the following:

```
frame radio_style ;
  default method is radio .

frame q1_style is a radio_style .
frame q2_style is a radio_style ;
  prefill is fred .
```

Because and Help

The because clause of a question will be associated with a button (or graphic) on the form that will when clicked show a popup dialog with the contained because text.

```
question single_choice_radio
  A single-choice radio-button question ;
  choose one of choices
  because I need to test a single-choice radio-button question .

question input_set
  A set input field ;
  input set ;
  browse file 'http://www.lpa.co.uk/fln.htm' .
```

Problems

There is a clash of operators between ProWeb and Flex. The nested use of the @ operator in ProWeb (e.g. li @ b @ `this is some text`) can not be used with WebFlex.

Chapter 4 – WebFlex Demos

This chapter gives a brief introduction to the supplied WebFlex demos.

Introduction

The screenshot shows a Windows Internet Explorer window with the title bar "WebFlex Demos - Tiscali 10.0". The menu bar includes File, Edit, View, Favorites, Tools, and Help. The toolbar contains standard icons for Back, Forward, Stop, Refresh, Home, Search, Favorites, Media, and Mail. The address bar shows the URL "http://www.lpa.co.uk/wfs_demo.htm". A status bar at the bottom displays "start C:\LPA\... WebFlex ... C:\LPA\... Wfs_Ref... EN 12:56". The main content area displays the "WebFlex Demos" page with the following text:

WebFlex Demos

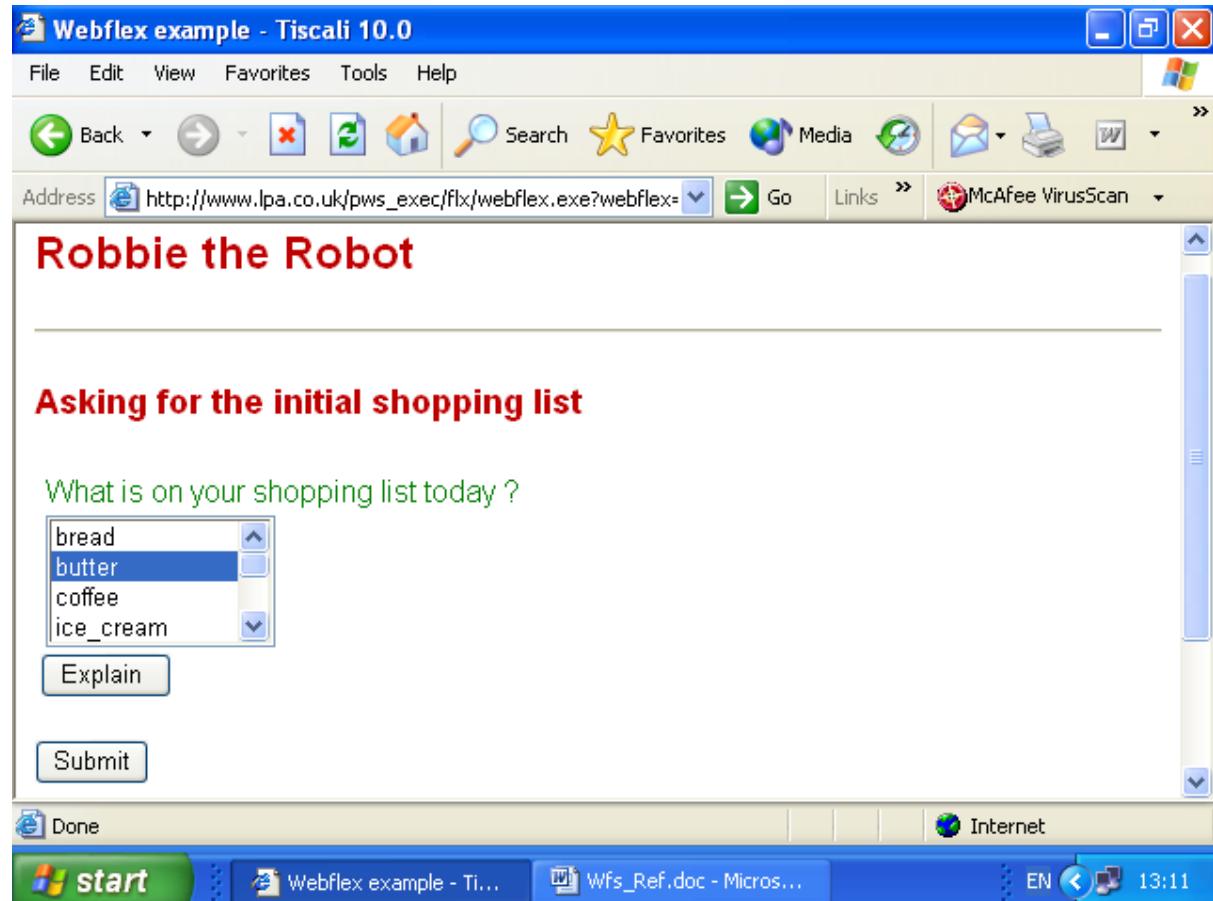
WebFlex is LPA's Expert System for the Internet. Based on ProWeb Server technology, this toolkit enables you easily to deliver expert system applications over the World Wide Web.

We have assembled a small selection of demonstration programs to give you some examples of what can be done with WebFlex; to explore these demos, please select from the following links:

Demo	Explanation
Robbie	Robbie helps select items to put into a shopping cart
Solvents	Select the correct solvent to use when cleaning equipment
Forest Yield	Recommend a species of tree seed to maximise forest yield
Insurance	Detecting insurance fraud using Bayes theorem.

Robbie

This example demonstrates the use of forward-chaining in a configuration and resource allocation problem, by way of a shopping expert system in which "Robbie the Robot" helps select groceries.



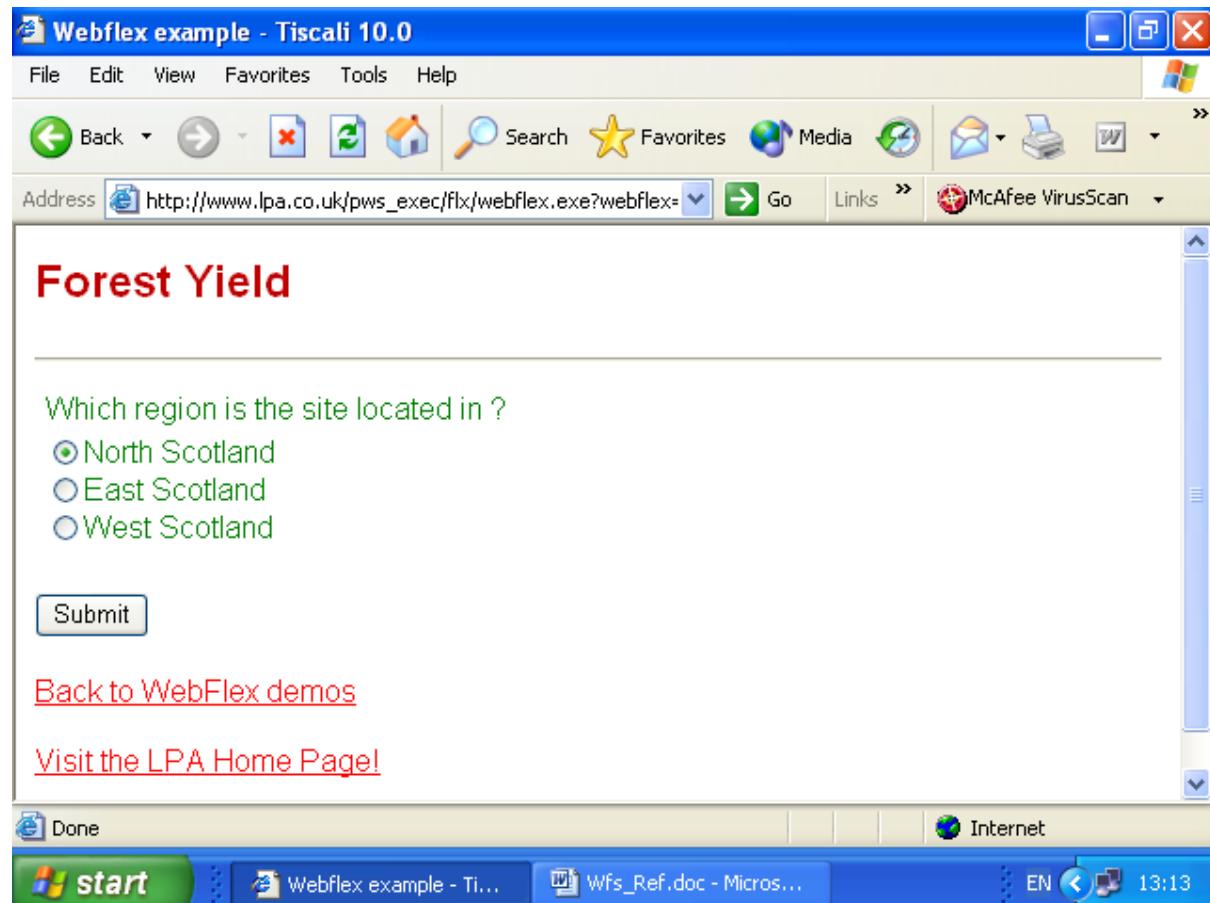
Solvents

This example recommends a solvent to be used when cleaning equipment. The recommendation depends on the equipment class, the ventilation of the site, the main material of the equipment and whether the equipment contains rubber compounds.

The screenshot shows a Windows desktop environment. At the top is the Windows taskbar with icons for Start, Internet, and system status. Below it is the Windows system tray. The main window is a web browser titled "Webflex example - Tiscali 10.0". The address bar shows the URL "http://www.lpa.co.uk/pws_exec/flx/webflex.EXE". The page content is titled "Solvents" and asks "How good is the ventilation at the cleaning site ?" with three radio button options: "poor" (selected), "fair", and "good". There are "Explain" and "Submit" buttons. Below the form are links to "Back to WebFlex demos" and "Visit the LPA Home Page!". The browser window has standard toolbar buttons like Back, Forward, Stop, Refresh, and Search.

Forest Yield

This example implements a simple expert system that recommends a species of tree seed, where to get the seed from, the normal yield of the seed and how that yield should be varied.



Insurance

This example uses forward-chaining uncertainty rules to implement a probabilistic fraud detection expert system based on Bayesian Networks.

The example also does some hypothetical reasoning and the user is able to go backwards and forwards using the browser, changing answers to questions and getting alternative suggestions.

Enter the details of the claim

Was the car recovered burnt out, damaged? <input type="radio"/> yes <input checked="" type="radio"/> not_known <input type="radio"/> no <input type="button" value="Explain"/>	Was the car S registration or later? <input type="radio"/> yes <input checked="" type="radio"/> not_known <input type="radio"/> no <input type="button" value="Explain"/>
Was the client too cut and dried pushing third party claim? (0-1) <input type="text" value="0.5"/> <input type="button" value="Explain"/>	Were you dissatisfied with the response of the client? (0-1) <input type="text"/> <input type="button" value="Explain"/>

Fuzzy Car

This example uses Fuzzy Logic to compare car attributes.

Using the age, engine capacity, cost, fuel efficiency, top speed and capacity of a car, this program calculates an overall rating of how suitable the car is for purchasing in comparison to a sample of pre-defined cars.

The screenshot shows a Microsoft Internet Explorer window with the title bar "Webflex example - Tiscali 10.0". The address bar contains the URL "http://www.ipa.co.uk/pws_exec/flx/webflex.exe?webflex=". The main content area displays a form titled "Simple Fuzzy Car Evaluation Program". The form asks for six pieces of car information: type, age, cc, cost, mpg, and top speed. Each question has a text input field next to it. Below the form, the status bar shows "Done" and "Internet".

Simple Fuzzy Car Evaluation Program

The system will calculate suitability based on the values you give.
Remember: you can always return back to this page and change your input.

Enter Your Car Details

What type of car is it ? <input type="text"/>	How old is the car in years (0-10) ? <input type="text"/>
What is the cc of the car (0-3000) ? <input type="text"/>	How much is the car in Euros (0-2000) ? <input type="text"/>
What is the mpg of the car (0-60) ? <input type="text"/>	What is the top speed of the car (0-140) ? <input type="text"/>

Done | Internet | start | Webflex example - Ti... | Wfs_Ref.doc - Micros... | EN | 13:15

Pesticide

This example demonstrates the use of user-defined questions within WebFlex.

This enables the presentation of questions to be owned by the developer and supplied, in this case, using ProWeb and Prolog.

The screenshot shows a Windows Internet Explorer window titled "Pesticide Expert System Example - Tiscali 10.0". The address bar displays the URL http://www.lpa.co.uk/pws_exec/flx/webflex.exe?webflex-. The main content area is a table listing four pests:

	<input type="checkbox"/> red_spider_mite	The red spider mite is an arachnid of minuscule proportions.
	<input type="checkbox"/> caterpillar	Caterpillars are the larval form of butterflies and moths.
	<input type="checkbox"/> aphid	Aphids are sap-sucking insects cultivated by ants.
	<input type="checkbox"/> slug	The common garden slug is a mollusc that likes beer.

The browser interface includes standard menu bars (File, Edit, View, Favorites, Tools, Help), toolbar icons (Back, Forward, Stop, Refresh, Home, Search, Favorites, Media, Links, etc.), and status bar information (Done, Internet, Start button, taskbar items for Pesticide Expert Syst..., WfS_Ref.doc - Microsoft Word, EN, 13:15).

Timetable

This example shows a forward-chaining method for creating a timetable.

Webflex example - Tiscali 10.0

File Edit View Favorites Tools Help

Back → Forward × Refresh Home Search Favorites Media Links McAfee VirusScan

Address: http://www.lpa.co.uk/pws_exec/flx/webflex.exe?webflex=1 Go

Please Enter a New Teacher

Please enter the name of a new teacher

Which classes can be taught ?

class1
class2
class3
class4

Please enter the subject for the new teacher

maths
physics
chemistry
english
geography
history

Which period is free ?

p1
p2
p3
p4
p5

Done Internet

start Webflex example - Ti... Wfs_Ref.doc - Micros... EN 13:16

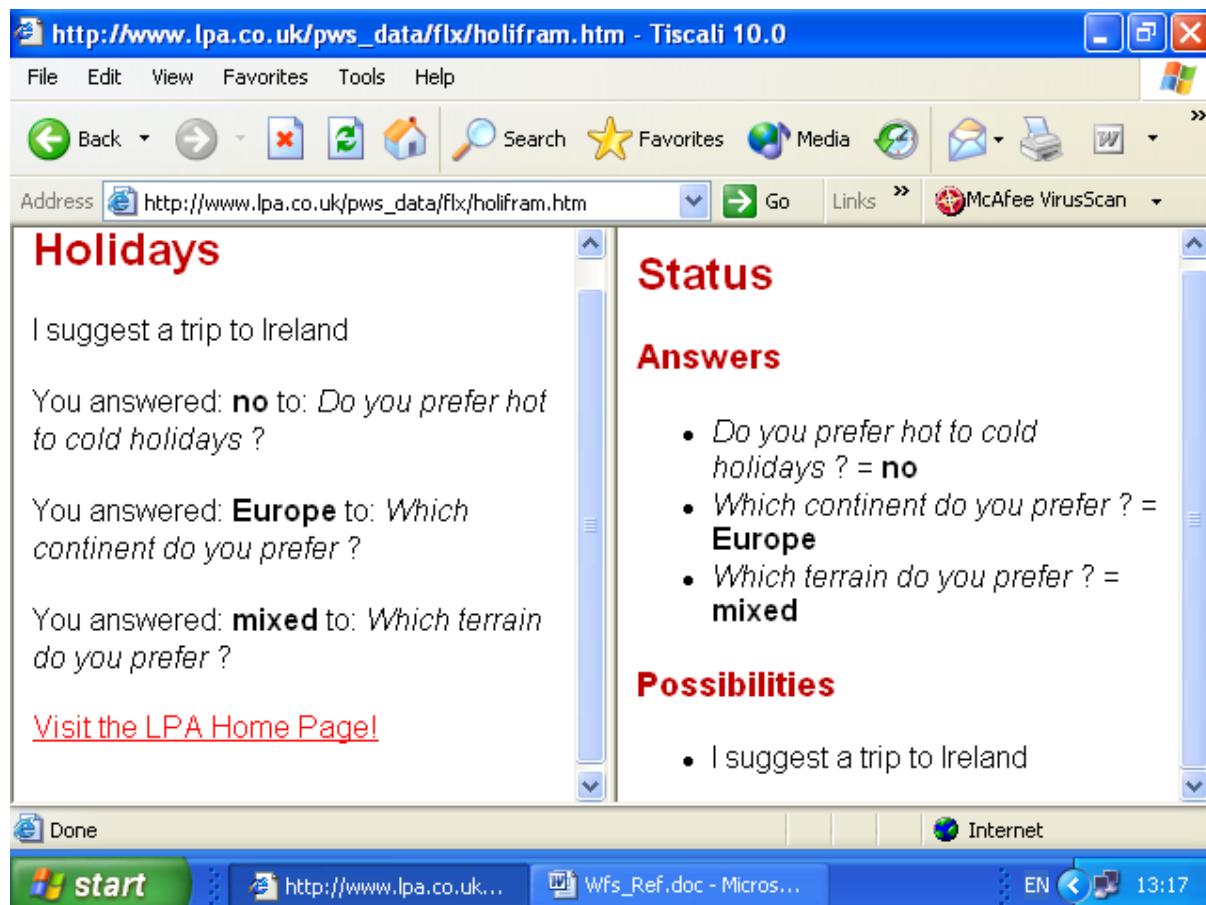
Holidays

This program is an example of a backward-chaining decision tree in Flex.

The example uses two frames:

- one to facilitate the expert system's questions and answers
- the other to show the current status of the execution

This second frame is automatically updated whenever a new answer is given. The Flex code has been written in a generic way, and can be easily adapted to other areas.

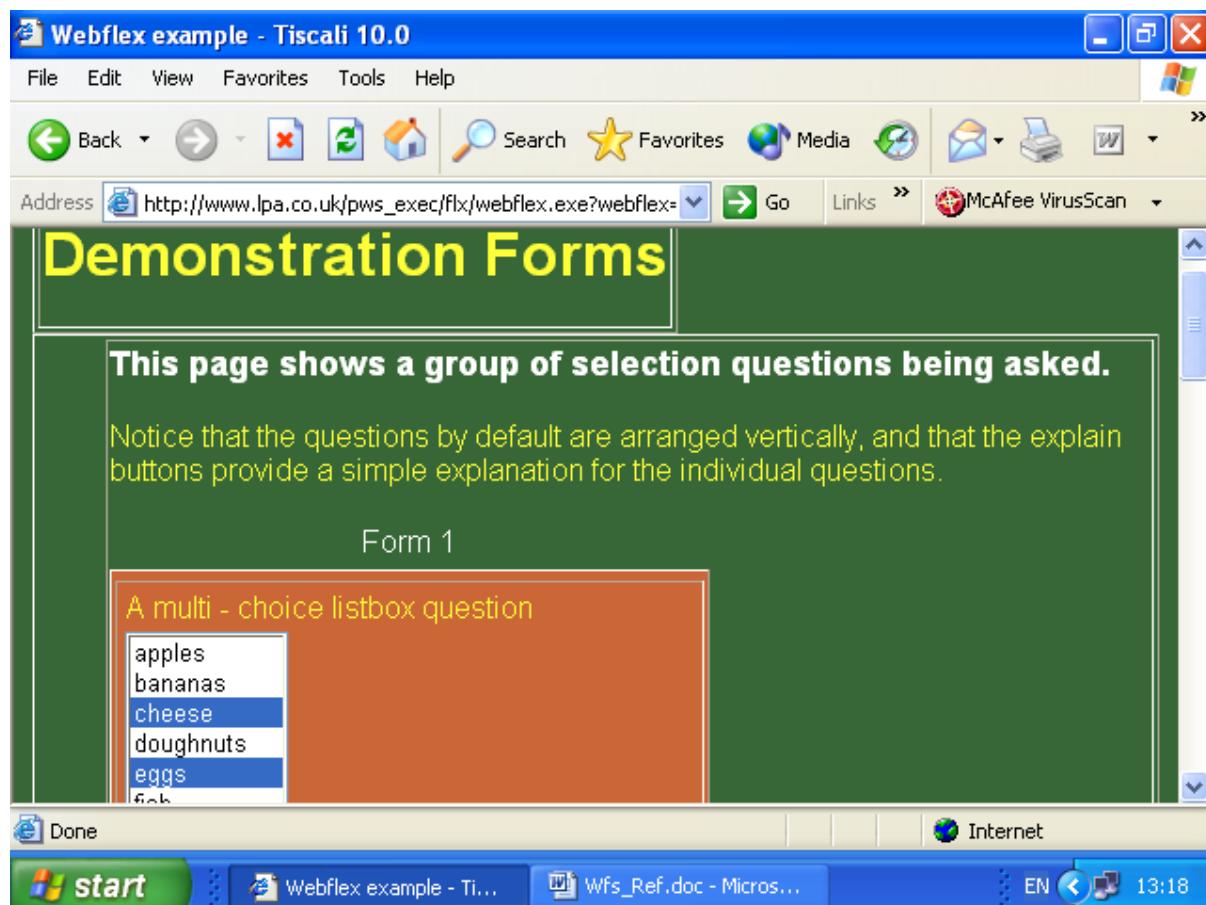


Forms

This example demonstrates the control WebFlex has over the display of forms using KSL.

The following are shown in this example:

- The asking of several questions on the same page through the use of groups.
- The declaring of attributes for the containing table.
- The control of attributes for individual questions.
- The constraint of numeric input fields to given ranges .
- The control of the position of questions in the containing table.
- The declaration of the containing HTML pages.
- The use of explanations and browse files.
- The declaration of graphics to use with command buttons.



Index

- @ operator, 21
- Aliases, 10
- BASE_URL, 14
- Bayesian Networks, 26
- because, 21
- BODY.HTM, 17
- Demo
 - Forest Yield, 25
 - Forms, 31
 - Fuzzy Car, 27
 - Holidays, 30
 - Insurance, 26
 - Pesticide, 28
 - Robbie, 23
 - Solvents, 24
 - Timetable, 29
- FLEX.PC, 9
- FOOT.HTM, 18
- forward-chaining, 26, 29
- frames, 30
- Fuzzy Logic, 27
- Group Styles, 19
- HEAD.HTM, 17
- HTTP Server, 7
- hypothetical reasoning, 26
- IIS, 7
- INI File Settings, 16
- input styles, 20
- Launch Page, 12
- multiple choice styles, 20
- PRO386W.EXE, 8
- PROWEB.OVL, 7
- PWS Directory, 6
- pws_data, 10
- pws_exec, 10
- Question Styles, 20
- Questions, 18
- single choice styles, 20
- Styles, 19
- TEMP_PATH, 14
- TEMP_URL, 14
- Trouble Shooting, 15
- uncertainty rules, 26
- user-defined question, 28
- WEBFLEX.OVL, 8
- WEBFLEX.SYS, 8
- WWWROOT Directory, 6