



Version 1.1
4-dec-2002

xDoc

The documentation tool for PROGRESS...

```
&ANALYZE-SUSPEND _VERSION-NUMBER AB_v9r12 GUI ADM2
&ANALYZE-RESUME
&Scoped-define WINDOW-NAME wWin
&ANALYZE-SUSPEND _UIB-CODE-BLOCK _CUSTOM _DEFINITIONS wWin
/*-----
File:
Description: from cn
Input Parameters:
<none>
Output Parameters:
<none>
History: New V9 Vers

-----
/* This .W file
/*-----
/* Create an unnamed po
by this procedure
that this procedu
will execute in t
cleanup will occu

CREATE WIDGET-POOL.

/* *****
/* Parameters Definiti
/* Local Variable Defi
DEF VAR LASTHDL AS HAND
DEF VAR LASTIMAGE AS CH

/* _UIB-CODE-BLOCK-END
&ANALYZE-RESUME

&ANALYZE-SUSPEND _UIB-
/* *****

&Scoped-define PROCEDU
&Scoped-define DB-AWAR
&Scoped-define ADM-CONV

&Scoped-define ADM-SUP
Target,Update-Source,Up

/* Name of first Frame
&Scoped-define FRAME-N

/* Standard List Defini
&Scoped-Define ENABLED
```

```
~
BUTTON-6 BUTTON-7 BUTTON-4 BUTTON-1 BUTTON-2 IMAGE-4

/* Custom List Definitions
*/
/* List-1,List-2,List-3,List-4,List-5,List-6
*/

/* _UIB-PREPROCESSOR-BLOCK-END */
&ANALYZE-RESUME

/* ***** Control Defi:
***** */

/* Define the widget handle for the wind
*/
DEFINE VAR wWin AS WIDGET-HANDLE NO-UNDO

/* Definitions of handles for SmartObject
*/
DEFINE VARIABLE h_dcust AS HANDLE NO-UNDO
DEFINE VARIABLE h_dorder AS HANDLE NO-UNDO

/* Definitions of the field level widget
*/
DEFINE BUTTON btnexit
  LABEL "E&xit"
  SIZE 42 BY 1.43.

DEFINE BUTTON BUTTON-1
  IMAGE-UP FILE "images/mnudmargu.gif"
  LABEL "Button 1"
  SIZE 42 BY 1.43.

DEFINE BUTTON BUTTON-2
  IMAGE-UP FILE "images/mnudynbufu.gif"
  LABEL "Button 2"
  SIZE 42 BY 1.43.

DEFINE BUTTON BUTTON-3
  IMAGE-UP FILE "images/mnudradrou.gif"
  LABEL "Button 3"
  SIZE 42 BY 1.43.

DEFINE BUTTON BUTTON-4
  IMAGE-UP FILE "images/mnuresbrwu.gif"
  LABEL "Button 4"
  SIZE 42 BY 1.43.

DEFINE BUTTON BUTTON-5
  IMAGE-UP FILE "images/mnusedou.gif"
  LABEL "Button 5"
  SIZE 42 BY 1.43.

DEFINE BUTTON BUTTON-6
```

4GL
65, rue Belle Cordière
01600 – PARCIEUX (France)

+334 78 98 34 08
www.4gl.fr



Does your PROGRESS application have any real documentation?	<input type="radio"/> Yes	<input type="radio"/> No
Do you have ever printed UIB or AppBuilder programs and kept the result as valuable?	<input type="radio"/> Yes	<input type="radio"/> No
Do you think that you do not need a technical documentation?	<input type="radio"/> Yes	<input type="radio"/> No
Are you happy without a program documentation?	<input type="radio"/> Yes	<input type="radio"/> No

... if you answer **NO** to one or more of the previous questions, then you need **xDoc** !.

xDoc is the solution to automatically document a PROGRESS appBuilder or UIB) application. It produces smooth and readable reports of application programs either in **xPrint** or **Adobe Acrobat PDF** formats. The very first version use an OLE interface with WORD... all interface has been rewritten to use **xPrint** because of the time needed to produce a document...

You can use xDoc with all 32 bits PROGRESS version (starting at 8.3).
The version 1.1 is delivered with version 9.1 compiled programs. Please ask us for other versions.

Please feel free to send your comments, remarks or reports at xdoc@4gl.fr.

I hope you enjoy with **xDoc**.

[Marcel FONDACCI](#)

xPrint must be available in order to run xDoc. No additional license is required but xPrint.dll and wPDF0101a.dll are mandatory to produce the documentation. A version of these two DLL's is provided in the setup and put into the installation directory.

HOW TO REGISTER?

You'll find all information at the end of document.

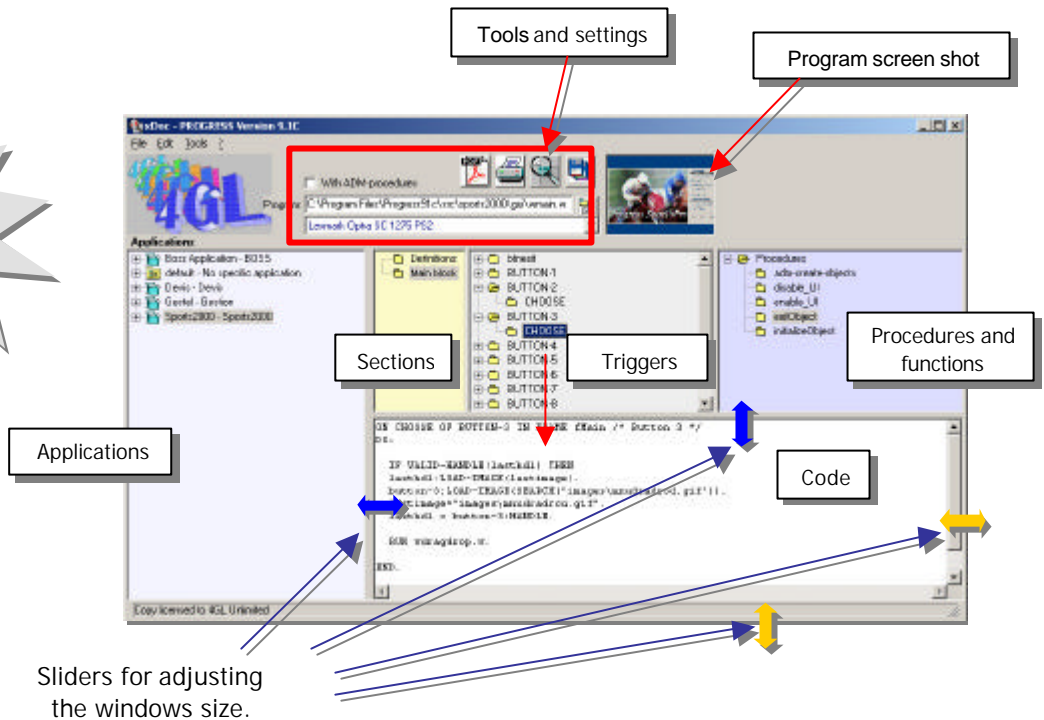
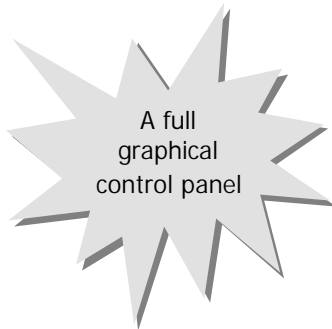
Evaluation version

The evaluation version allows you to test any function.

**But without license, the documentation is limited to
3 triggers and procedures.**

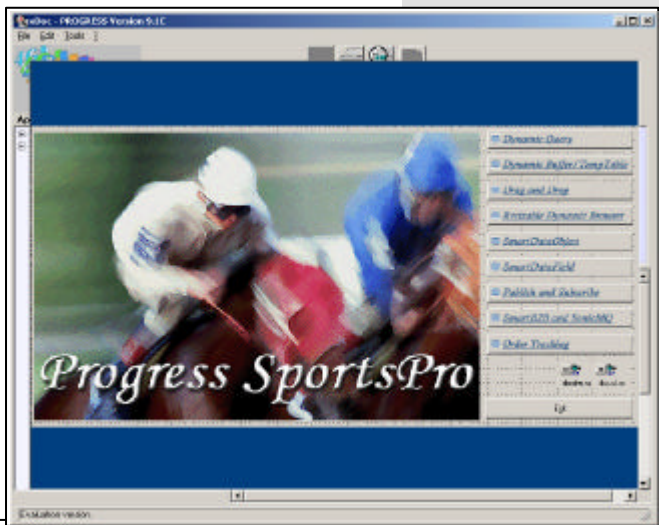


xDoc: main control panel



xDoc offers a graphical control panel. You can register different applications with default folder, bases connections, ini file and initialisation procedures.
xDoc makes at run-time all needed settings to run from the application default folder, connect the application databases, run the init procedure, load the *.ini file etc., to present to the program the expected running context.

This panel is used to control the 'paper' documentation process. But as all triggers, procedures, functions are displayed in the code window, it can be used for on-line documentation, as an help for the application developer...



When clicking on the screen shot in the main panel, xDoc zooms the image.



Starting with xDoc

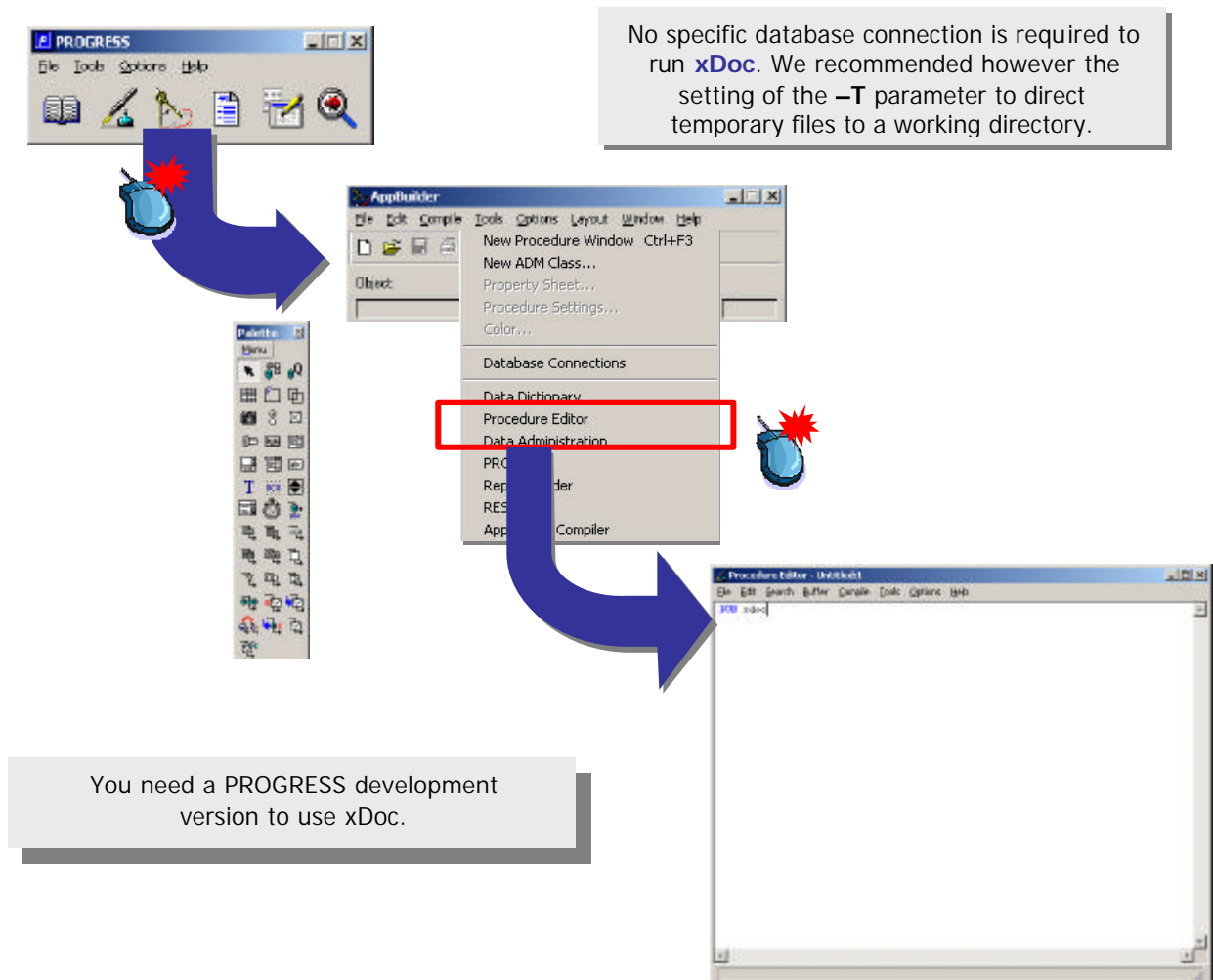
xDoc reads all source lines from a PROGRESS program.

It interprets the normal tags of appBuilder programs to display the different windows: definitions, main-block, triggers and procedures code.

When clicking on a documentation button, the second phase begins:

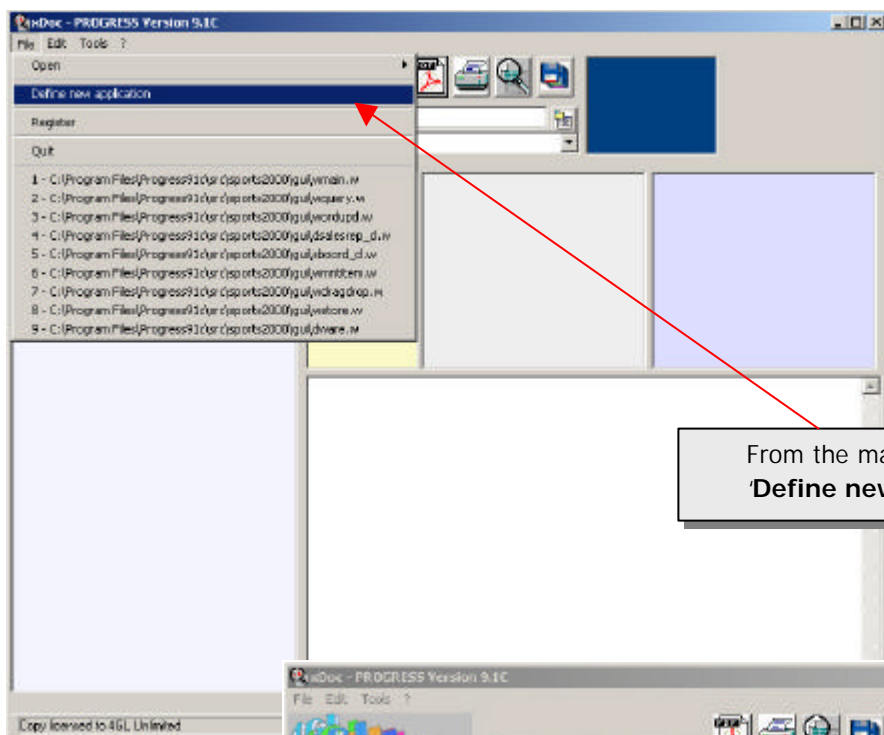
- **xDoc** tries to capture the application window : it scans the original code to compile and run a modified version of the programs to get a 'run-time' window copy. Due to the complexity of this process, the differences between the programmers coding and/or syntax errors that may exist, this process can be unsuccessful. In this case, **xDoc** calls the **appBuilder** PROGRESS program to load the program and make the screen capture from the appBuilder design window.
- **xDoc** calls xPrint to produce the final documentation in xPrint or PDF formats.

It is reason why the **xDoc** main program (xdoc.w) should be run from the procedure editor **AFTER** the appBuilder window :

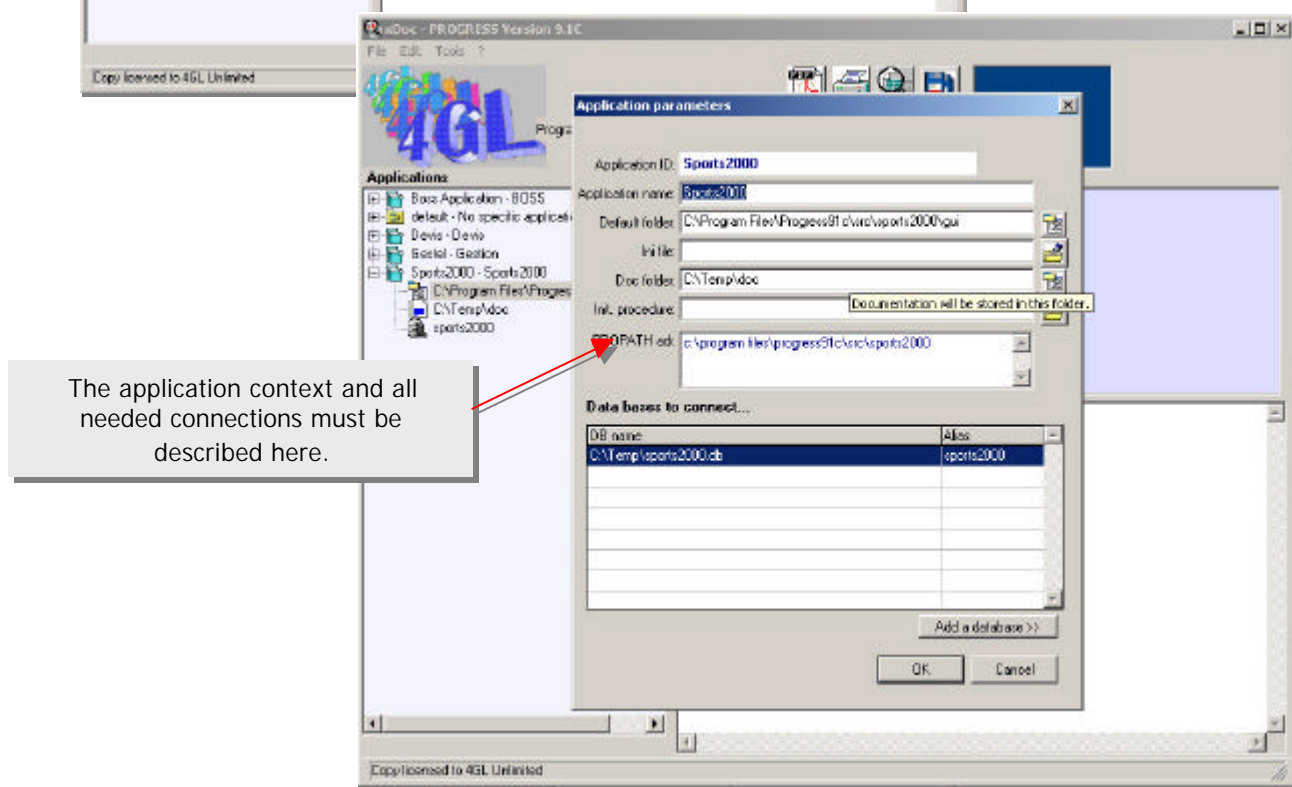




xDoc: defining application context



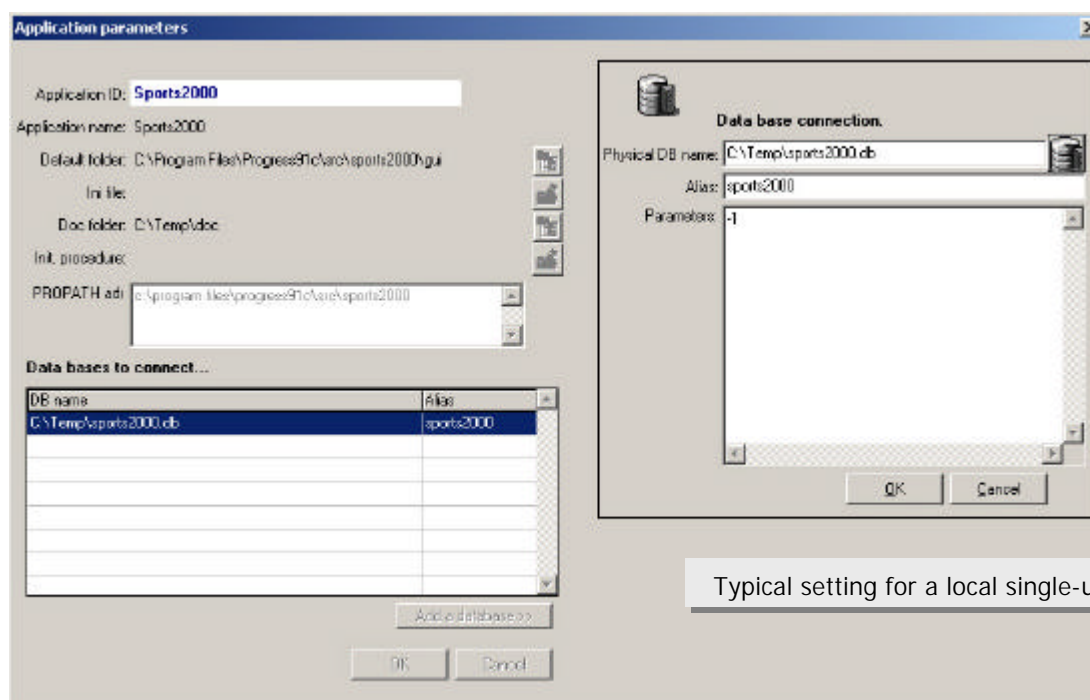
From the main menu, choose the 'Define new application' item.



The application context and all needed connections must be described here.



When double-clicking on the database name, or on the **add database** button, the following window appears:



Specify all database connections that your application needs, then click on the '**OK**' button.



Technical notes :

The more difficult problem is the screen capture. As previously said, xDoc uses a specific technique to get a 'run-time' capture.

Some different things are done like replacing 'PARAMETERS' by local variables, suppress shared declarations when possible etc. When this process fails, xDoc automatically calls **appBuilder** to capture a design screen. You may experiment that some programs have no screen capture due to this process or that some programs generate warning messages like 'No customer record available'.

Some rules should be respected to get an automatic documentation:

- When the first step of screen capture is done, xDoc takes care of the UIB IS RUNNING setting to compile code. The program will compile as if called by the appBuilder program. This setting allows some extra processing to provide a convenient context. It can be used for example, to read a needed record. You can experiment that when running your program directly from the appBuilder.

Example :

```
DEF INPUT PARAMETER R AS ROWID.  
FIND customer WHERE ROWID(customer) = R NO-LOCK.  
will be translated by xDoc to :  
DEF VAR R AS ROWID.  
FIND customer WHERE ROWID(customer) = R NO-LOCK.
```

When this code is executed, a run-time message is displayed 'No customer record available'.

As a documentation step, modify your code like this :

```
&IF DEFINED(UIB_IS_RUNNING) > 0 &THEN  
    DEF VAR R AS ROWID.  
    FIND FIRST customer NO-LOCK.  
    R = ROWID(customer).  
&ELSE  
    DEF INPUT PARAMETER R AS ROWID.  
    FIND customer WHERE ROWID(customer) = R NO-LOCK.  
&ENDIF
```

- When the program expects **shared** variables or buffers : xDoc eliminates these definitions for the main program. But when this program includes other smartObjects, the value of the shared variables can be a problem. To avoid that, you can define an initialisation procedure in the application definition to run a piece of code that defines all necessary values.
- Main menus :
Main menus can have a 'QUIT' statement to return to the O/S after processing. When executing this QUIT statement, the PROGRESS session is closed !
As with the previous point, protect the QUIT statement with :

```
&IF DEFINED(UIB_IS_RUNNING) = 0 &THEN  
    QUIT.  
&ENDIF
```

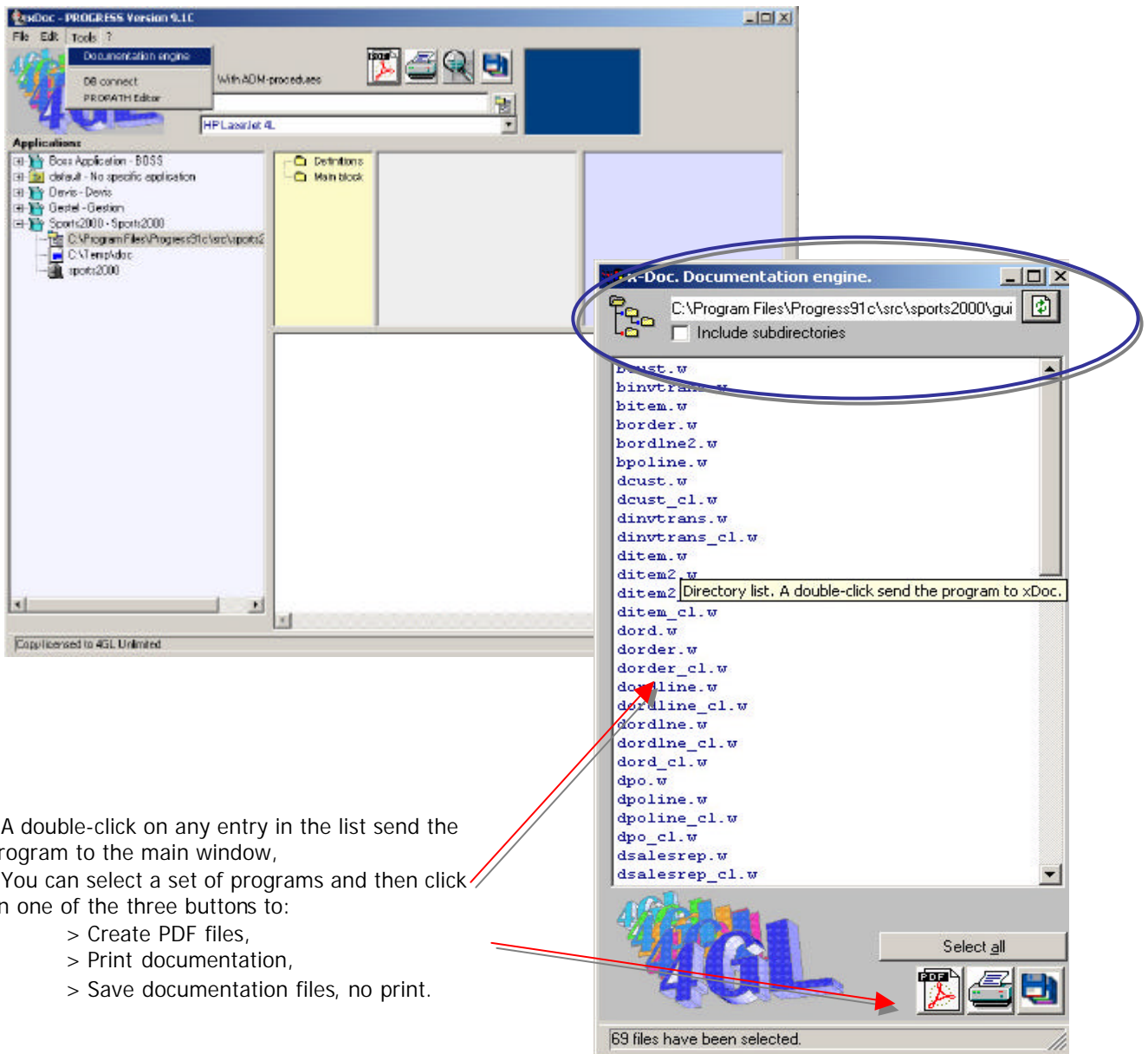


- **xPrint** is used to produce the documentation. No additional license is required but xPrint.dll and wPDF0101a.dll are mandatory. A version of these two DLL's is installed by the setup program.
- **Treeview4GL.ocx** and **xlImage.ocx** are used by xDoc. These **free Active-X from 4GL** are integrated to the install procedure. Fell free to use it in any other development. However, technical reference manuals are only available when downloading these Active-x from www.4GL.fr and are not included in the xDoc product.
- **During the capture** stage of application screens, we recommend to **not interfere with the screen display**. If a window overlaps the PROGRESS window just before the screen shot, this may result in incorrect image.
- **xDoc** is not designed to work with non-UIB/appBuilder programs.



xDoc documentation engine:

As application contains several application programs. You can document a full set of application programs through the xDoc documentation engine.



- A double-click on any entry in the list send the program to the main window,
- You can select a set of programs and then click on one of the three buttons to:
 - > Create PDF files,
 - > Print documentation,
 - > Save documentation files, no print.



HOW TO REGISTER ?

You'll find in the next page a form to fax to
Or to send to

+33 4 78 98 34 08
register@4gl.fr

PRICE LIST (04-dec-2002)

Base license (1 developer)	1.200 EUROS H.T.
Additional licenses	450 EUROS H.T.
Site license (any number of developers)	3.000 EUROS H.T.

A license allows you to install xDoc on **1 PC only.**

Special offer until 31 dec 2002

Base license (1 developer)	1.000 EUROS H.T.
Additional licenses	300 EUROS H.T.
Site license (any number of developers)	2.000 EUROS H.T.

Each license includes one year of free updates.

Evaluation version

The evaluation version allows you to test any function.

But without license, the documentation is limited to
3 triggers and procedures.

WARNING : This price list is subject to changes. Always refer to the last
xDoc.doc at www.4gl.fr to get the current values.



xDoc 1.1 Registration FORM

4GL 65, rue Belle Cordière 01600 – PARCIEUX France	Fax # +33 478 98 34 08 register@4gl.fr
--	---

Register: **see the price list above.**

DATE :

License Name:				
	<i>(mandatory : this name appears as a key for your license number)</i>			
User Name:				
Company:				
Address:				
:				
:				
City:		State:		
Zip:		Country:		
Tel.:		Fax:		
e-Mail:				
	<i>(mandatory : for return of your license key)</i>			
	License	Quantity	Unit Price	Total price
CEE TAX Identification: <i>For CEE users, please specify your TVA/VAT identification</i>				
Payment method	<input type="checkbox"/> VISA	<input type="checkbox"/> MasterCard		
Card number:				
Expiration date:				
Name of card owner:				
Signature:				



Disclaimer

Commercial usage of this software is at your own risk.

The author makes no warranties about the suitability of this software, either express or implied.

The author and/or owner shall not be liable for any damages suffered by licensee as a result of using, modifying or distributing this software and application that uses it.