

COBVIEW PRINT SUITE

COBVIEW

Version 3.20

Users Manual

INDEX

Introduction	4
Limits of Cobtools-View	6
Command line Parameters	7
Variables Predefined.....	10
First changes	11
Colors	13
Report Sections.....	14
Report Title <REPORT>	15
Preparing page <PAGE>	16
Margins <MARGIN>	16
Page Titles <HEADER>	17
Footer pages <FOOTER>	17
HEADER and FOOTER content:	18
Fonts 	19
Font Styles	20
Center <CENTER>	20
Bold 	20
Italic <I>	21
Underlined <U>	21
Columns <COL>	22
Skip blank line 	26
Skip to new page <NEWPAGE>	27
Printing Texts and Variables <PRINT>	28
Printing Multiline Texts <MEMO>	30
Images impression 	33
Print Shapes <DRAW>	34
Draw Charts <CHART>	36
Printing Barcodes <BARCODE>	39
Details with several lines <BAND>	40
Including external files. <INCLUDE>	41
Passing values to an external file <VAR>	45
Placing remarks in the external files.....	46
Global Config file COBVIEW.DEF.....	47
Set configuration values <SET>	48
Initializing vars when page change <INITIALIZE>	53
Events in report parse <EVENT>	54
Cleaning defined Events <CLEAR EVENT>	55
Checking free space in the page <CHECK>	56
Sending Email Data <EMAIL >	57
Watermark protection <WATERMARK >	58
Web Reports Data <WEBREPORTS >	59
Search text in the report.....	60
Exports.....	61
COBVIEW.EXE – The report viewer	62

Charts with data reports	64
COBVIEWPDF	65
COBVIEWCGI.....	66
COBSPool-WIN Print Server for Windows Lan.....	67
COBSPool-FTP Print Server over Internet	69
COBVIEW Print Suite Configuration	71
COBVIEW.EXE Configuration	72
Email/WebReports Configuration	73
Send reports by EMail	74
Web-Reports Service configuration	75
Configuracion de Spool-FTP.....	76
Spool-WIN Configuration	80
EscFonts	83
COBVIEW License register	86

Introduction

Cobtools-View is an application that helps with report presentation in graphic environment. Was developed thinking in the people that how us program in COBOL.

The adaptation is very easy, since all the report the process of elaboration of the report is equal exactly, the only thing that needs is to add some "Special trainings" to which we called TAGS, which identify functions of CobView.

In this manual we will take to you of the hand, to turn a report. This way you will be able to know all necessary one to be able to work.

Most of the TAGS they are very similar to language HTML, to somehow make the adaptation easier of reports. This way, if you already know something of HTML you will not have problem in using this program.

How interacting with cobtools-View from Cobol

At the moment most of the compilers of Cobol they have a function to execute external programs of the Operating system, for example, in Acucobol this is possible with:

```
CALL "SYSTEM" USING "program.exe parameters".
```

With this option you can make a call to a executable program. Now, the program is a file with the extension EXE, call COBVIEW.EXE, and will be necessary to pass like parameter the file to him that contains your report, ej:

```
CALL "SYSTEM " USING "COBVIEW.EXE MIREPORT.TXT".
```

In order to generate your file with the content of the report, it will be necessary to make a change in the declaration of your SELECT, so that it does not send it port of printer, but that lodges the report in a disk file:

```
SELECT IMPRESO  
  ASSIGN TO PRINT "MIREPORT.TXT".
```

All the rest of the report is equal exactly. Now we suppose that, your declaration of the FD of report is the following one:

```
FD IMPRESO  
  LABEL RECORD IS OMITTED.  
01 LINEA    PIC X(200).
```

In the rest of examples of this manual we will use these definitions of report.

A thing is very important, if your system works in multiuser way, several people could enter to see the same report, by this, the file that it generates in disc must be different every time, this can be done of the following way:

```
SELECT IMPRESO
      ASSIGN TO PRINT WF-IMPRESO.
...
...
WORKING-STORAGE SECTION.
77  WF-IMPRESO      PIC X(20) .
77  WK-TIME        PIC 9(08) .
....
....
PROCEDURE DIVISION.
....
....
      ACCEPT WK-TIME FROM TIME.
      STRING WK-TIME, ".LIS" DELIMITED BY SIZE INTO WF-IMPRESO.
```

This way you could generate a report file whenever you enter to print it. Since the name would be armed with the time of the computer, including milliseconds, this causes that it is "almost impossible" that two computers take the same number, even so, you could add a validation when the file is opened.

Limits of Cobtools-View

Maximum amount of lines in a Footer: 20

Maximum amount of columns in a report : 60

Command line Parameters

/PRINT

It does not visualize the report by screen, sends the report direct to the printer, show the progress bar and show the printer dialog selection.

Inside your report, you can indicate this function with the tag:

<SET PRINT="ON">

/PRINTDIRECT

It does not visualize the report by screen, sends the report direct to the printer, show the progress bar, but the selection dialog printer is not showed. The report is send to default printer or the printer selected with the parameter /PRINTER.

Inside your report, you can too indicate this function with the tag :

<SET PRINTDIRECT="ON">.

/PRINTER "Printer name"

It allow select a printer, diferent from the default printer.

Inside your report, you can too indicate this function with the tag:

<SET PRINTER="printer-name">

/COPIES n

Allow indicate how copies are printed our report, if our report is previewed before print, the printer dialog contain set this value.

Also, you can indicate this parameter inside your report file with the tag:

<SET COPIES="n">

/NOPROGRESS

This parameter is for hide the progress bar while your report is printed.

Inside your report not exist any tag equal to this parameter.

The preview is showed, only hide the progress-bar.

/FROMPAGE #page

It initiates the impression or export from the page indicated in # page

/TOPAGE #page

Print or you export until the page indicated in # page

/EMAIL

With this parameter, we can send our report directly by Email, to any account email in the internet. Need the TAG <EMAIL>

/WEBREPORTS

With this parameter, we can send our report to Reports Server indicated in the configuration. Need the TAG <WEBREPORTS>

/NOBUTTON 123456789A

Allow hide any button from the Cobview Toolbar.

With this option we can do that our application show or hide any function like Exports, email sends, or searches.

Every number represents a button that we can hide.

- | | |
|-------------------------|---------------------|
| 1.- Search button | 6.- Export to EXCEL |
| 2.- Charts generation | 7.- Export to HTML |
| 3.- Send by Email | 8.- Export to PDF |
| 4.- Send to Web-Reports | 9.- Export to BMP |
| 5.- Export to ASCII | A.- Export to JPG |

Sample: For hide the PDF and EXCEL Exportation, we need:

/NOBUTTON 68

For hide Send by email, we need:

/NOBUTTON 3

/EXPORT-XXX file-name

It makes an export of the report, is necessary to specify the name of the file that we want to generate. This file was lodged in the exports directory which is defined in the Configuration window of cobview to less of than specifies a file name with drive and directory, in that case was lodged where is specified. Is necessary to replace part XXX with the type of following export:

The allowed types of export are:

- ASCII - Export to text file delimited
- EXCEL – Export to sylk file, excel compatible. it is not necessary to have installed Excel.
- HTML – Export the report to HTML file, this file contains only text, without images.
- PDF – It exports the report to file pdf, this type of file is completely transportable and can be seen in any computer that it has installed Acrobat Reader.
- BMP – It generates a series of BMP files with the name of the file and it is added the page number, generates a file by each page.
- JPG – It also generates a file of image by each page, just as bmp.

Samples:

Normal call

COBVIEW report-file

For print direct to printer, showing progress bar and selecting the printer:

COBVIEW report-file /PRINT

Hidding the progress bar

COBVIEW report-file /PRINT /NOPROGRESS

Direct to special printer in silent mode:

COBVIEW report-file /PRINTDIRECT /PRINTER "printer name" /NOPROGRESS

Generate our own PDF file:

COBVIEW report-file /EXPORT-PDF filename

Variables Predefined

Some predefined variables exist, that can be used only indicating in where it is desired to place the value of variable.

The variables must go preceded by sign \$ before the name.

\$DATE(format) – return the date of the computer, the format parameter must be::
dd/mm/yy, dd/mm/yyyy, dd-mm-yyy

\$TIME(format) – return the time of the computer, the format must be:
hh:nn:ss, where hh is the hour, nn the minutes and ss the seconds.

Obsolete:

\$PAGE(title) - print the page number. It can go in *HEADER* section or *FOOTER* section.
The title parameter specify the label that alongside left with of the page number.

Sample: \$PAGE(page)

\$PAGE – Return the value for the current page.

Now you can use any variable inside your text, sample:

```
<print "Page: $PAGE" top="1" left="7">
```

or

```
<set x="7">Page $PAGE
```

See also:

```
<set page top="n" left="n" title="page">
```

First changes

The headers ones are understood, are those that go to the beginning of each page. These headers is necessary to define them only once in all the report, and must be defined at the beginning of the report.

Normally all we used an line counter to know how when to make a jump of page. Then, it will not be necessary here to make these validations, since the CobView was in charge of this. For example, we suppose that we have an impression like the following one:

```
PROCEDURE DIVISION.  
....  
010-GENERA-REPORTE.  
    OPEN OUTPUT IMPRESO.  
    MOVE 67 TO CONLIN.  
    PERFORM 100-LEE THRU 200-EXIT.  
    CLOSE IMPRESO.  
    STOP RUN.  
100-LEE.  
    READ ARCHIVO NEXT RECORD AT END GO 200-EXIT.  
    MOVE CAMPO-1 TO DET-CAMPO1.  
    MOVE CAMPO-2 TO DET-CAMPO2.  
  
    IF CONLIN > 50  
        PERFORM 500-TITULOS.  
  
    WRITE LINEA FROM DETALLE-1.  
    ADD 1 TO CONLIN.  
    GO TO 100-LEE.  
200-EXIT.  
    EXIT.  
500-TITULOS.  
    PERFORM 600-SALTA-HOJA.  
    WRITE LINEA FROM TITULO-01.  
    WRITE LINEA FROM TITULO-02.  
    WRITE LINEA FROM TITULO-03.  
    WRITE LINEA FROM TITULO-04.  
    MOVE 04 TO CONLIN.
```

This is a classic and normal impression. If this observed, in the paragraph 010-GENERA-REPORTE, never is called the paragraph 500-TITULOS, since this it was called before printing forward edge of the report. And each that prints 50 lines, the titles will be printed.

This same impression, is of the following way: (already they added first tags to him to indicate titles)

```
PROCEDURE DIVISION.  
....  
010-GENERA-REPORTE.  
    OPEN OUTPUT IMPRESO.  
    PERFORM 500-TITULOS.  
    PERFORM 100-LEE THRU 200-EXIT.  
    CLOSE IMPRESO.  
    STOP RUN.  
100-LEE.  
    READ ARCHIVO NEXT RECORD AT END GO 200-EXIT.  
    MOVE CAMPO-1 TO DET-CAMPO1.  
    MOVE CAMPO-2 TO DET-CAMPO2.  
    WRITE LINEA FROM DETALLE-1.  
    ADD 1 TO CONLIN.  
    GO TO 100-LEE.  
200-EXIT.  
    EXIT.  
500-TITULOS.  
    WRITE LINEA FROM "<head border=2 bgcolor='rgb(227,230,196) ' ">".  
    WRITE LINEA FROM "<font size=10 color='red' ">".  
    WRITE LINEA FROM "<b ">".  
    WRITE LINEA FROM TITULO-01.  
    WRITE LINEA FROM TITULO-02.  
    WRITE LINEA FROM TITULO-03.  
    WRITE LINEA FROM TITULO-04.  
    WRITE LINEA FROM TITULO-05.  
    WRITE LINEA FROM TITULO-06.  
    WRITE LINEA FROM "</b ">".  
    WRITE LINEA FROM "</head ">".
```

First change, no longer is used an line counter, this is not necessary for us. Second change, the titles we sent them to call only one time. The CobView was in charge to place them in each page.

This does that in your impression file, it is printed:

```
<head border=2 bgcolor='rgb(227,230,196) ' >  
<font size=10 color='red' >  
<b >
```

The TAGS must bee into <...>, when a parameter needs a value but a word, it will be necessary that this locking up in quotes ("), or apostrophes ('). Recognizes all the TAGS in upper case or lower case.

Colors

Cobtools-View handles the colors of a simple way, you have option to handle the fixed names or, to do combinacion special that you wish using internal function RGB.

The fixed colors that CobView recognizes are the following:

Black
Maroon
Green
Olive
Purple
Teal
Gray
Silver
Red
Lime
Blue
Fuchsia
Aqua
White

The way to do one combination is using the RGB function:

RGB(red-value, green-value, blue-value)

How you know, RGB makes onecombination of these colors to form the color that you wish, these values go from 0 to 255 for perfect combination. Once executing COBVIEW counts with a toolbar, and this has a button to call the color dialog where you can see the values.

The colors you can use them either in font, or like bottom of the headers (HEADER) or the feet of page (FOOTER), to continuation I go to expuse some examples to you:

```
<head border=2 bgcolor='rgb(227,230,196)'>  
<font size=10 color='red'>
```

You also have the option to use the combinations of colors just as in HTML It is to say defining colors of the type: “#0F0F0F”. The result of the equal color that in a web page. Example, the following one defines font black color:

```
<font size=10 color="#000000">
```

Report Sections

Your report file contain the following main sections:

Section name	Common Tags in section
[page definition]	<page>, <margin>
<HEADER> [header declarations] </HEADER>	Open text, %vars% <print>, , <draw>, <barcode>, , <set>, , , <I>, <U>, <center>, <right>,
<FOOTER> [footer declarations] </FOOTER>	Open text, %vars% <print>, , <draw>, <barcode>, , <set>, , , <I>, <U>, <center>, <right>,
[special definitions]	, <defcol>, <band>, <var>, <event>
<body>	The body report start here
[report content]	All tags valids for print some content in the report.

Notes:

The tags defined in **<header>** or **<footer>** will be executed in every page of all report.

The tags defined in the Special definitions section can be only for declaration, not execute any action here.

All instructions defined in the body section will be executed only one time, when the parser find the tag.

Report Title <REPORT>

We can assign the title report with the tag:

```
<REPORT TITLE="Title for the Report">
```

With this, we indicate our title or report name, for when the viewer show the report in the screen, allocating this title in the COBVIEW Title Bar.

In the previous sample, the COBVIEW title bar will show:

COBVIEW – Title for the Report

This way, if we have two COBVIEW running, we can know that report we are seen in the screen in this moment.

This tag only affect the report title in the viewer title bar.

We can even put the Application name, sample:

```
<REPORT TITLE="Accounting –Sales report">
```

Preparing page <PAGE>

Before initiating the data impression, we must indicate to Cobview the page configuration. Size, direction, as well as the size of the margins that we wished.

<PAGE SIZE="size" ORIENTATION="orientación">

The first word must be PAGE, this indicates that the size of page.

SIZE it accepts the following values: LETTER, LEGAL, A3, A4 and CUSTOM

WIDTH, only for SIZE="CUSTOM", indicate the page width (in inches)

HEIGHT, only for SIZE="CUSTOM", indicate the page height (in inches)

ORIENTATION: accepts PORTRAIT and LANDSCAPE

Margins <MARGIN>

Another very important configuration is the margins, indicates the size of the area that will be nonprintible by the Cobview. Something that there is to stress is that some printers have defined of make margins, which cannot be changed, and if for some reason smaller margins to these are created, will be a portion of the report that will not be printed.

All values from this tag are in inches.

<MARGIN LEFT="0.1" RIGHT="0.1" TOP="0.1" BOTTOM="0.1">

The word MARGIN is obligatory, this indicate that will define the page margins.

LEFT, it specifies the left margin.

RIGHT, it specifies the right margin.

TOP, it specifies the superior margin of the page

BOTTOM, it specifies the inferior margin of the page.

Page Titles <HEADER>

```
<HEADER BORDER="n" BGCOLOR="color" ROUNDED="y" HEIGHT="n">  
... header content  
</HEADER>
```

It defines page heading. HEADER can be changed by HEAD, both are recognized.

BORDER, it indicates the wide one of the edge, which draws a frame around all the headed, if this is omitted does not have to border, as well as if also BORDER="0" is indicated to this.

BGCOLOR, it indicate the background color that have the header section.

ROUNDED, indicate if we want borders rounded in the box.

HEIGHT, indicate the minimum height for the header, in inches. If while print the header content, the height printed is greather than the value here specified, then use the height printed as header height.

Footer pages <FOOTER>

```
<FOOTER BORDER="n" BGCOLOR="color" ROUNDED="y">  
... footer content  
</FOOTER>
```

A foot page is the one that is printed in the inferior part of all the pages. Also is necessary indicate it at the beginning of the report, can be followed of headers. The parameters are identical to the Header section.

BORDER, it indicates the wide one of the edge, which draws a frame around all the headed, if this omitted does not have to border, as well as if also BORDER="0" is indicated to this.

BGCOLOR, it indicate the background color that have the footer section.

ROUNDED , indicate if we want borders rounded in the box.

HEADER and FOOTER content:

In these sections, you can put the following elements, and this are printed in every page:

Text formatted:

```
<header>
<font name="Arial" color="blue" size="12">
<center><b>%COMPANY%</b></center>
</header>
```

Print tag:

```
<header>
<font name="Arial" color="blue" size="12">
<center><b>%COMPANY%</b></center>
<print "%tit-report% $time(hh:mm)" top="0.50" left="7.40">
</header>
```

Images:

```
<header>
<font name="Arial" color="blue" size="12">
<center><b>%COMPANY%</b></center>

</header>
```

Shapes:

```
<header>
<font name="Arial" color="blue" size="12">
<center><b>%COMPANY%</b></center>
<draw line top="1.50" left="+0.00" height="0" width="8">
</header>
```

Barcode:

```
<header>
<font name="Arial" color="blue" size="12">
<center><b>%COMPANY%</b></center>
<barcode value="121214" top="0.50" left="1.40" width="1" left="1">
</header>
```

Notes:

In the header and footer sections, you can skip a line with a blank line and the tag `
`.

Fonts

```
<FONT FACE="name" SIZE="n" COLOR="color">  
... text content
```

The tag FONT changes the font present. All the following impression will be with this new font, until finding another font definition.

FACE can be replaced with **NAME**, and corresponds to the name of the font desired, example: Arial, Times New Roman. Must be closed by quotes or single quote.

SIZE, is the font size, sample: SIZE="10"

COLOR, is the color desired to use with font.

Also is possible save the font before modify this to make an special impression, and to return font to the previous font, this can make with:

```
<FONT SAVE>  
<FONT FACE="Times New Roman" SIZE="10" COLOR="TEAL">  
... impresión content  
<FONT RESTORE>
```

FONT SAVE – Save the actual font, specially when we need return to the same font.
FONT RESTORE – Restore the last font saved.

Font Styles

The next TAGS or marks are for change the font style, and they can be one preceded of another one, to make combinations of styles , example:

```
<B><I><U>  
...text content  
</U></I></B>
```

Center <CENTER>

```
<CENTER>  
... text content  
</CENTER>
```

```
<CENTER>Text to print</CENTER>
```

<CENTER> this serves to center a text in the page of automatic way. It can be used of the two ways as it is indicated above of these lines, if does not find the TAG **</CENTER>** all impression will be centered, until found the tag of end center.

<RIGHT> this serve to align the text or image to right margin in the page.

Bold

```
<B>  
...text content  
</B>
```

```
<B>Text to print </B>
```

**** this causes that the following impression is done with BOLD style, which causes that the letter has a Wide appearance. In order to finish this type of impression the mark must be : ****

Italic <I>

<I>
...Text content
</I>

<I>Text to print</I>

<I> This causes that the following Impression to the mark is done in Italic style, until finding the end mark: </I>

Underlined <U>

<U>
... text content
</U>

<U>Text to print</U>

<U> This causes that the impression is done in emphasized style, also until finding the mark: </U>

Columns <COL>

The main problem to print in Windows, is that we cannot send a text in sequential form without giving format him. What would cause that all our impression is distorted completely, for example, the following text:

00001	LUCIA MENDIETA RDZ	LUCIA MENDIETA
00002	HOSTELES TROPICAL SA DE CV	HOSTELES TROPICAL SA DE CV
00003	ISABEL VELASCO GALO	ISABEL VELAZCO GALO
00004	BAR LA GALERIA	ARACELY CARRASCO
00006	OSTIONERIA EL DELFIN	ESTHER RAMOS
00007	ULTRAMARINOS TAMPICO	MARTHA RODRIGUEZ
00008	BAR PLAZA 65	JOSE MARIA MENDOZA
00010	JUAN JOSE LOPEZ CHAV	JUAN JSOE LOPEZ
00012	RINCON DEL GRUTA	GUSTAVO CUELLAR
00015	VICENTE NOGUERIA	VICENTE NOGUERIA

In an Impression in text mode, it would leave exactly just as as it is seen above. But in an impression in Windows, the impression would leave the following way:

00001	LUCIA MENDIETA RDZ	LUCIA MENDIETA
00002	HOSTELES TROPICAL SA DE CV	HOSTELES TROPICAL SA DE CV
00003	ISABEL VELASCO GALO	ISABEL VELAZCO GALO
00004	BAR LA GALERIA	ARACELY CARRASCO
00006	OSTIONERIA EL DELFIN	ESTHER RAMOS
00007	ULTRAMARINOS TAMPICO	MARTHA RODRIGUEZ
00008	BAR PLAZA 65	JOSE MARIA MENDOZA
00010	JUAN JOSE LOPEZ CHAV	JUAN JSOE LOPEZ
00012	RINCON DEL GRUTA	GUSTAVO CUELLAR
00015	VICENTE NOGUERIA	VICENTE NOGUERIA

In order to solve this problem, we must define columns, which indicate where initiate and where finish in our file of report. Also they indicate to Cobview in that position of the paper we want that they appear.

We are going to suppose that in our file, the three columns are defined of the following way:

WORKING-STORAGE SECTION.

...

```
01 DETALLE-01.  
   02 DET-KEY      PIC X(05)B.  
   02 DET-NAME     PIC X(40)B.  
   02 DET-OWNER    PIC X(30)B.
```

Therefore, after of headers of our report, we will need to define the columns that we are going to print. This can do with the TAG **<DEFCOL>**.

```
<defcol #1 left=0 width=1 bgcolor=yellow border=1 title="KEY" align=left
visible=yes>
<defcol #2 left="+0.05" width=2 bgcolor=yellow border=1 title="NAME">
<defcol #3 left="+0.05" line="+1" bgcolor=yellow border=1 itle="OWNER">
```

The parameters for this tag are::

#n, indicate the number of column, normally is sequential. You can have until 60 columns by report.

LEFT="n", This indicate the position where we want that it appears the column in the page. This value is in inches. It can also be represented in relation the final position of the Previous Column, sample: "+0.05", that is 0.05 inches to the right of where the previous column finishes.

LINE="+1", This is used to create a column, that will be print a line down, from the present position. Single one specifies the first time, and the columns that follow to him in the declaration,.

WIDTH="n", This indicates the size width for the column in inches, without taking the account number of characters that content. If a too small column for the content is created that can have, the impression can alongside straight, invading its zone of next column.

BGCOLOR="color", indicate the background color for the column title.

BORDER ="n", this parameter is for indicate that we wished a box for the column title.

TITLE="name", is the title for the column in the page. When the page eject, the titles will print again, under the headers section **<HEADER>**.

ALIGN="align", indicate the position for the colum, can be: CENTER, RIGHT, LEFT

AUTOLINE – This parameter indicates that the line in that the columns will be printed need be calculated about the wide one of the page. And if a column does not fit in the same line, automatically will be placed in the following line, and left position LEFT indicated by **<SET COLUMN LEFT="nn">**.

VISIBLE="yes/no", this serves to indicate if the column is visible in the report or will not be printed. This is very useful when some customer asks for us that we hide a column, or needs a new column, we can hide them or show, without touching nothing of the logic of the program. In addition, if the Headers and the columns definitions have them in a external file, only we need modify this external file to show or hide columns, without need of recompile our program.

TOTAL="variable", with this parameter is possible specify that we want a total with the sum of all data in this column. This variable need be declared with the tag <VAR>

Once that has been defined the columns, every detail need have the division of columns, so that the Cobview can identify where this initiates and finishes a column, this will be done modifying our original detail, and inserting the TAG.

```
WORKING-STORAGE SECTION.
...
01  DETALLE-01.
    02  FILLER          PIC X(08) VALUE "<COL #1>".
    02  DET-KEY         PIC X(05) .
    02  FILLER          PIC X(08) VALUE "<COL #2>".
    02  DET-NAME        PIC X(40) .
    02  FILLER          PIC X(08) VALUE "<COL #3>".
    02  DET-OWNER       PIC X(30) .
```

By example, let us suppose that we have a report that prints 3 lines by each detail:

```
01  LINEA-1.
    02  DATO1          PIC X(30) .
    02  DATO2          PIC X(25) .
    02  DATO3          PIC X(25) .

01  LINEA-2.
    02  DATO21         PIC X(30) .
    02  DATO22         PIC X(25) .
    02  DATO23         PIC X(25) .

01  LINEA-3.
    02  DATO31         PIC X(30) .
    02  DATO32         PIC X(25) .
    02  DATO33         PIC X(25) .
```

Our COBOL program already prepared to conduct all the operation of the report with 3 TEXT lines.

Then, we do not change that, better we define 9 columns, the 9 columns will be in different lines, but that does not matter to cobview.

```
01  LINEA-1.
    02  FILLER          PIC X(08) VALUE "<COL #1>".
    02  DATO1           PIC X(30) .
    02  FILLER          PIC X(08) VALUE "<COL #2>".
    02  DATO2           PIC X(25) .
    02  FILLER          PIC X(08) VALUE "<COL #3>".
    02  DATO3           PIC X(25) .
```

```
01 LINEA-2.
  02 FILLER      PIC X(08) VALUE "<COL #4>".
  02 DATO21      PIC X(30) .
  02 FILLER      PIC X(08) VALUE "<COL #5>".
  02 DATO22      PIC X(25) .
  02 FILLER      PIC X(08) VALUE "<COL #6>".
  02 DATO23      PIC X(25) .

01 LINEA-3.
  02 FILLER      PIC X(08) VALUE "<COL #7>".
  02 DATO31      PIC X(30) .
  02 FILLER      PIC X(08) VALUE "<COL #8>".
  02 DATO32      PIC X(25) .
  02 FILLER      PIC X(08) VALUE "<COL #9>".
  02 DATO33      PIC X(25) .
```

Notes:

The max number of columns is defined in the section "Limits of cobtools-view".

If you need do some print in sequential form, without using the columns, this will need delete this, otherwise the Cobview will continue dividing the data line in Columns. In order to eliminate the Columns you must use: **<CLEARCOL>**

With this tag, the columns definition is canceled. Obvious if later you need to again make use of the columns, is necessary to return to print the columns definition, before initiating the impression of the details in columns.

You can redefine your columns in any time inside your report, even in your <body> content.

Automatically, when the last column is printed, cobview skip a line. If the last column defined is never print, when the column 1 is printed again cobview skip a line.

If you need to control your line skip, you can cancel the automatic line skip with the tag: **<set column autoskip="N">**, and you have the control for skip a line when you want, with the tag **
**. Then need add a tag **
** after print your last wished column.

Skip blank line

This tag allows us to skip to next line.

Format:

This skip size is using the current size font.

Skip to new page <NEWPAGE>

With this tag it's possible to make that our page jumps, with no need have to print the rest of the page with spaces.

Format:

<NEWPAGE>

Immediately it initiates the impression in the following page.

It is necessary to clarify that this instruction finishes the present page and initiates the new one in blank, in such a way that the new page already will contain the Headers and Footers that are had declared by default.

Printing Texts and Variables <PRINT>

Once a variable has been declared with <VAR namevar="valorinicial">, and that is had defined a column with a variable like totalizer, it is possible to print the content of this variable totalizer in each change of page, to print a total in the page foot.

For this, we use the next tag: <PRINT "%namevar%" LINE="n" POSITION="n">

With this mark it is possible the impression of the content of a variable, or text and variable in the same content. The content need be inside quotes "content".

the necessary parameters are:

LINE – it indicates the line where we want to place the data. It is represented in inches. Also it can be replaced by **TOP**.

* If is placed within the **HEADER** area:

If a value is indicated "+N" it indicates that will be printed after of the superior margin of the page.

If a value is specified "-N" it indicates that will be printed to start off the inferior part of header "N" inches above.

If a value is specified "N" it is an absolute position within the page.

* If it's placed within the **FOOTER** area:

If a value is specified "+N" it indicates that after of where initiates footer, was placed "N" inches down.

If a value is specified "-N" it indicates that will be N inches before where it initiates footer.

If a value is specified "N", it's an absolute position within the page.

* If is placed within the **BODY** area:

" +n " value indicate the last vertical position plus n inches

" n " indicate a fixed position in the page

POSITION – it indicates the horizontal position where we want to place the data, it is possible also to be used **POS** and **LEFT**.

PICTURE – it indicates picture mask it to use to print the variable, for example:
PICTURE="ZZZ,ZZZ,ZZ9.99".

WIDTH – this indicate the width define an impression area

ALIGN – Indicate the align mode for the text content inside the impression area. Can be: LEFT, RIGHT, CENTER

XLSCOL – Allow us indicate in that cell we wished that appear the data that we are printing, special for Excel, Html or text exportation.

ANGLE – it's for indicate the inclination angle for the data.

This angle will be always in numeric value positive. Sample:

ANGLE="45" – show the text with 45 of inclination

ANGLE="90" – show the text in vertical position, starting from the position to up

ANGLE="270" – show the text in vertical position, starting from the position to down

In order to obtain the impression when the page jump, we must place the instruction

<PRINT> inside the area <FOOTER> ... </FOOTER>.

Print also can print sections BAND complete. Like the sections BAND they already have included the positions, is not necessary specify in the PRINT the positions, only the section BAND name, sample:

<PRINT "BAND-NAME">

If you want a show total in the footers, you need:

```
<footer>
Footer Content
<print tot-variable top="top" left="5">
<initialize tot-variable value="0.00">
</footer>
```

First print the variable value and later initialize the variable for the next page.

The <PRINT> command not advance the line position. Because the print command have inside the position for print content.

If you want advance the line position in the page, you must add
 commands for every line that you want skip, another method is using the <SET Y="n"> command for move directly the line position for your needs after the print command.

Samples:

<print "Name: %NAME%" top="1" left="3">

<print "Page: \$PAGE" top="0.50" left="7.50">

Printing Multiline Texts <MEMO>

Since the version 2.95, is possible insert Memo sections.

A memo section can contain several lines of information, and print all with Wordwrap and Carriage returns include

Format:

<MEMO TOP="n" LEFT="n" WIDTH="n" DELCR="Y/N">

Text contained in the memo

</MEMO>

Parameters:

- **TOP** – it can be a constant value or of the type “+n” that it means that be placed “n” inches down from the present position. And another method is absolute position in the page.
 - If is placed within the HEADER area:
If a value is indicated “+N” it indicates that will be printed after of the superior margin of the page.
If a value is specified “-N” it indicates that will be printed to start off the inferior part of header “N” inches above.
If a value is specified “N” it is an absolute position within the page.
 - If it's placed within the FOOTER area:
If a value is specified “+N” it indicates that after of where initiates footer, was placed “N” inches down.
If a value is specified “-N” it indicates that will be N inches before where it initiates footer.
If a value is specified “N”, it's an absolute position within the page.
 - If is placed within the BODY area:
“+n” value indicate the last vertical position plus n inches
“n” indicate a fixed position in the page
- **LEFT** – it can be a constant value for indicate the left position.
- **WIDTH** – This is the wide for the memo area.
- **DELCR** – If indicate this parameter with “Y” value, then if the memo text contains Carriage Returns, this character is erased, and the text is printed in only one line, and wordwrap according to width size.

Is necessary close the Memo Tag with a </MEMO> for indicate the end of the memo text.

The memos can be allocated in the BODY report and inside a band.

If you need put a MEMO in a Headers section, then put this memo in a special band, and call the band with a PRINT tag in the headers section.

Since 3.20 version, you can put MEMO directly in Headers or Footers section.

If you need put the memo inside a band, and pass a value from the COBOL program to fill the memo, you need declare the memo in the band with the following format.

```
<MEMO NAME="memo5" TOP="n" LEFT="n" WIDTH="n" DELCR="Y/N">
</MEMO>
```

Look the parameter NAME, is for declare MEMOS with content variable, memo5 is the name for this memo declaration.

From this way, the memos is declared as empty, but have a name, and like a band is not printed until you execute the PRINT BAND tag, before call this tag need indicate the value for the memo with name "memo5" for this sample.

And since your COBOL program need build the memo content with the following format:

```
<MEMO VAR="memo5" >
```

Text contained in the memo

```
</MEMO>
```

With this format, we tell to COBVIEW that the value for the MEMO called "memo5" is the text inside the tags <MEMO> and </MEMO>. Look the VAR parameter, here use VAR. Next need print the band that contain the MEMO "memo5".

Sample:

MEMOS.DEF file

```
. . .
<header>
</header>
<footer>
</footer>

<band BAND_MEMO>
<MEMO NAME="MEMO1" top="1" left="2" width=5">
</MEMO>
</band>

<body>
. . .
```

REPORT.TXT file

```
<include "memos.def">
. . .
<memo var="MEMO1">
Text for the MEMO1 Control
With several lines. Later print the BAND
</memo>
<print "BAND_MEMO">

<memo var="MEMO1">
Another sample with a different text
Using the same BAND and MEMO control
</memo>
<print "BAND_MEMO">
. . .
```

Your COBOL program must generate a REPORT.TXT file with this format.

Images impression

The formats that you can use are:

- BMP files
- GIF files
- JPG files

Is possible insert images in Header and Footer, so that when jumping page these images are also printed along with the text.

For print image we can do with the following tag:

Format:

```
<IMG SRC='d:\file' TOP='n1' LEFT='n2' WIDTH='n3' HEIGHT='n4'>
```

Parameters:

SRC – This indicates the name of the file that contains the image to print

TOP – This is for specify the superior position where we want the image, like all the positions, this is in inches, this is possible to be specified a position fixed or absolute to the page, or can be from the present position adding sign + before the position, for example: **TOP="+0.05"**, this means that from where this at this moment advances 0.05 inches and prints the image.

LEFT – In order to specify the left position in where we want the image. In this parameter we can only specify absolute positions in the page. We can't use the sign of (+). And also it is in inches. Now, if you not specify the left position, the image is positioned with the tags <center> or <right>.

WIDTH – It is for indicating the wide one of the image that we wished. He is optional, if the image is not specified then print of the size that the image have. A size different from the physical can be specified and the image will expand or contract according to is the case, also is possible to specify a size in percentage with the sign of % at the end of the value, example: **"50%"**. This value is in **pixels** or **percentage** only.

HEIGHT – This parameter is for indicating the height of the image. Also it is possible to specify it in **pixels** or **percentage**. And like width, if the image is not specified print according to the physical size that it has.

Print Shapes <DRAW>

We can print the following shapes type in ur reports:

- Lines (LINE)
- Boxes (BOX)
- Boxes Rounded (ROUNDRECT)

For this, we can use the following command:

<DRAW type TOP="n" LEFT="n" WIDTH="n" HEIGHT="n" BORDER="n" COLOR="color" BGCOLOR="color">

Where the type can be the following words:

- LINE – It creates a line from TOP and LEFT, with a wide of WIDTH
- BOX – It creates a picture, from TOP and LEFT, with wide of WIDTH and a height of HEIGHT.
- TOP – it can be a constant value or of the type "+n" that it means that be placed "n" inches down from the present position. And "-n" with this print the draw "n" inches up from the current y position.
- LEFT – it can be a constant value for indicate the left position.
- WIDTH – This is the wide for the shape. It can be of wide a fixed one or of the type "-n" that want to say "n" inches before the right margin of the page.
- HEIGHT – This it is the height of the figure. Single it can have a fixed height.
- RIGHT – Indicate the right position to draw the shape, special for draw lines. RIGHT is the absolute position in the page.
- BOTTOM – Indicate the bottom position to draw the shape. BOTTOM is the absolute position in the page.
- ROUND – Indicate if the box is rounded. Need specify the ratio valud for the curve, example: ROUND="0.30" it's 0.30 inches of ratio in the curve.
- BORDER – Indicate the width for the Border in the box. Now accept decimal numbers.
- BGCOLOR – if is specified, the box will filled with the indicated color.
- COLOR – Is the color for the border in the box.

If a DRAW in header is specified or to footer, this figure is created in each page. Even if the coordinates are outside the areas of header and footer.

If the print shape is in the BODY area, this shape is printed only this time.

Samples:

<draw line top="0.50" left="2.00" width="4.30" bottom="0.50" border="1" color="Blue">

Draw a horizontal line from left to right side.

<draw line top="1.25" left="+0.00" width="-0.00" bottom="1.25" border="2">

Another way to draw a horizontal line

Left="+0.00" is since the left margin, Width="-0.00" until the right margin

Top="1.25" from the 1.25 inches in the top page

<draw line top="1.50" left="2.40" height="8.50" right="2.40" border="1">

Draw a vertical line, since the 1.50 top inches with 8.50 inches of large
Left and right positions is 2.40

<draw line top="+0.05" left="0.10" width="0.65" height="0.00">

Draw a line size the current position + 0.05 inches, and a height of 0.00

With this draw too a horizontal line, because height of 0.00 not advance.

Draw Charts <CHART>

This feature is only in cobview 1.80 and up.

With the tag <chart> can draw charts from the following types:

- Line
- Bar
- Pie
- StackedBar (Stacked only too work)

To draw the chart need indicate the Series and the values that will used for draw the Graphic.

Chart format:

```
<chart type="type" top="n" left="n" width="n" height="n">  
.... Chart elements ...  
</chart>
```

Parameters:

Type – type can be any from the following: Line, Bar, Pie, Stacked

The following parameters need be in inches, or the unit definied for the report (CMS or MM).

Top – Indicate the top position in the page for draw the chart.

Left – Indicate the left position in the page for draw the chart.

Width – It's the width for the chart.

Height – It's the height for the chart.

Elements for the chart

Registering the chart title:

```
<chart title="Chart Title">
```

Register the title for the chart, this title is printted in the top of the chart.

Registering the Series Name

`<chart serie="Serie Name">`

Set a new serie for values. Need first define the serie, and later put the values.

Registering Values for the chart

`<chart value="numeric value" legend="legend for value">`

Set a new value for the serie before defined.

Hiding the Legend box

`<chart ShowLegend="off">`

With this tag, the Legend box is not showed, and chart will occupy all area specified for it.
By default this value is set in ON.

Placing the Legend box outside the area chart

`<chart topLegend="n" leftLegend="n">`

With this tag, we can put the Legend-Box in another place in our page.
Remember that, the Legend-Box always is allocated in the right of the chart.
If we allocate the Legend-Box in another place (by sample: in the bottom chart), the chart is automatically resize to all area defined for it.

Notes:

- The PIE chart type not need legend for the values.
- Since the 1.82 version, you can define the next parameters with a Variable value: TYPE, TITLE, SERIE, VALUE y LEGEND.

Sample:

```
<chart type="bar" top="1" left="3" width="4" height="4">
<chart serie="2003">
<chart value="10000.00" legend="amount 1">
<chart value="15000.00" legend="amount 2">
<chart value="8000.00" legend="amount 3">
<chart serie="2004">
<chart value="9000.00" legend="amount 1">
<chart value="20000.00" legend="amount 2">
<chart value="6000.00" legend="amount 3">
</chart>
```

Printing Barcodes <BARCODE>

Now is possible print bar codes in our report.

Format:

<BARCODE TOP="n" LEFT="n" WIDTH="n" HEIGHT="n" BORDER="n" TYPE="type" VALUE="number">

Parameters:

- TOP – it can be a constant value or of the type "+nn" that it means that "nn" was placed inches after the present position.
- LEFT – it can only be constant with an absolute position.
- WIDTH – This he is the wide one which it will have our bar code, must of being of wide fixed.
- HEIGHT – This it is the height of the bar code. Single it can have a fixed height.
- TYPE – it indicates the name of the type of bar code to use. The valued types are detailed down.
- VALUE – Contains the numeric value to convert in bar code.

Bar code types:

CODE25INTERLEAVED	CODEMSI	CODEEAN128A
CODE25INDUSTRIAL	CODEPOSTNET	CODEEAN128B
CODE25MATRIX	CODECODABAR	CODEEAN128C
CODE39	CODEEAN8	
CODE39EXTENDED	CODEEAN13	
CODE128A	CODEUPCA	
CODE128B	CODEUPCE0	
CODE128C	CODEUPCE1	
CODE93	CODEUPCSUPP2	
CODE93EXTENDED	CODEUPCSUPP5	

Notes:

* Now the value in the parameter value can be in a variable format: **"%variable%"**

Details with several lines <BAND>

A band can contain several lines of impression, within this block can also be defined variables, even can contain instructions PRINT within this BAND.

Format:

```
<BAND band--name >  
BAND content  
</BAND>
```

Sample:

```
<BAND band-name >  
band content to print, can have several lines  
can have variables: %VAR%  
<PRINT "Text placed fixed" TOP="1.5" LEFT="2">  
</BAND>
```

This band definition can be printed with a tag <PRINT>, sample:

```
<PRINT band-name >
```

This way, we can have a BAND defined in an external file of permanent way, and only add one line to our program to print all the band in our report.

Parameters:

AUTOLINE – If use this parameter with True value, jump a line for every line-inside the BAND section. If user False, then jump only when we use
 tag, or using parameter TOP="n" in print tags.

HEIGHT – Parameter allow us specify height size for the band. It's only for validate space free in the page, or jump to next new page and print in top.

TYPE – Used only by COBVIEW Developer, for build Title columns, for exportation in HTML, Excel and ASCII types. The only value valid is "TITLEDDETAIL".

Notes:

The band declaration need be before <body> tag.

Including external files. <INCLUDE>

We can now include external files with the tag : <INCLUDE>

This can be used to have a fixed part of the report that would not be necessary to be generated from inside of our COBOL program .

Format:

<INCLUDE "D:\FILE.EXT">

The file must be inside quotes or apostrofes.

An external file is a good tool, if we need change the report without recompile our cobol program. An external file will have to contain a fixed part of the report, like for example: the headers, footers.

Suppose that we wished to print a listing that consists of 3 columns.

```
01 LINEA-1.
  02 FILLER          PIC X(08) VALUE "<COL #1>".
  02 LIN-DATO1       PIC X(30).
  02 FILLER          PIC X(08) VALUE "<COL #2>".
  02 LIN-DATO2       PIC X(25).
  02 FILLER          PIC X(08) VALUE "<COL #3>".
  02 LIN-DATO3       PIC X(25).
:::
:::
PROCEDURE DIVISION.
:::
:::
LEE-ARCHIVO.
  READ ARCHIVO NEXT RECORD AT END
  GO TO TERMINA-REPORTE.
  MOVE DATO1 TO LIN-DATO1.
  MOVE DATO2 TO LIN-DATO2.
  MOVE DATO3 TO LIN-DATO3.
  WRITE LINEA-REPORTE FROM LINEA-1.
:::
:::
TITULOS.
  WRITE LINEA-REPORTE FROM "<INCLUDE 'TITREP.DEF'>".
:::
:::
```

As it can be observed, at the time of arming the headers for the report, only we make reference to an external file called TITREP.DEF, because this file will can contain the following thing:

```
<header border="1">
<font face="Arial" size="12" color="blue">
Report sample with external files
</header>

<footer>
Copyright - cobtools 2003
</footer>

<font face="Arial" size="10" color="black">

<defcol #1 left=0    width=1    border=1 title="DAT1" visible=yes>
<defcol #2 left=1.5  width=1.5  border=1 title="DAT2" visible=yes>
<defcol #3 left=4    width=2    border=1 title="DAT3" visible=yes>

<body>
```

Let us suppose that we wished to hide a column, we did not need to recompile our program, with only modifying file TITREP.DEF and putting the column wished with parameter visible=no will be sufficient.

Now, we suppose that we have a report with 20 defined columns, and that our program prints all the columns as if in fact they were needed, but that by default our single system showed 10. And a client arrives whom she asks for to us to place a column to him that already we have but not shown Navy, good then with this solution not problem. With this we could be adding columns to our reports and showing it conforms them for our clients to ask us.

Even will have our customized reports, nonsingle with the name of the company, but also with an image logo of the client. Since an external file can even make reference to another external file, so as it is continuation:

```

      COBOL PROGRAM
      |
EXTERNAL FILE (define report)
      |
EXTERNAL FILE 2 (define headers of client)
```

Here, our program makes reference to an external file, which can be ordered to define the columns of our report, and this to make reference to another external file as well to have the file of the image logo of the client. This way, if all our reports are in the same way, with single changing the external file of the headed one of the reports we will be able to change of logo of all clients.

They even will can to have the definition of the HEADER of all the reports, in case a client arrived and she said to us "I want your system, but that the reports do not take boxes in the titles".

This way will have the same program, but changing file TITREF.DEF, so that the content is the following:

```
<INCLUDE "HEADER.DEF">
<font face="Arial" size="12" color="blue">
Sample with external files
</header>

<INCLUDE "FOOTER.DEF">
</footer>

<font face="Arial" size="10" color="black">

<defcol #1 left=0    width=1    border=1 title="DAT1" visible=yes>
<defcol #2 left=1.5  width=1.5  border=1 title="DAT2" visible=yes>
<defcol #3 left=4    width=2    border=1 title="DAT3" visible=yes>

<body>
```

The file HEADER.DEF is that to order to define tag < HEADER > for all the reports of our application, will can contain the following thing:

```
<header border="1" bgcolor="RGB(125,125,125)">

```

This file define our header, using a box and background color. In addition it defines to use an image in coordinates 0.70, 0.60.

This way, if all our reports include an external file to define ours headers, and this external file uses as well a global external file that the global part of all the reports defines of general way, we can modify the appearance of all the headers only modifying a file. Either to change the color of bottom of all the reports, to clear the position frame or changing the image to them of logotipo

(Version 1.62 up)

Now you can have only a file for all your reports, and define the fixed parts of the reports of all reports in the same file, separating the declarations with the new tag <report>.

Format:

```
<REPORT report-name>  
  ... special report definitions  
</REPORT>
```

Now, for include only this section in our cobol program, we need use the following include tag:

```
<INCLUDE 'filename.ext' REPORT='report-name'>
```

Now all samples use this method, all reports use the same file.

Passing values to an external file <VAR>

It is very possible that you got yourself to find with the problem to need an external file, but with a part of the variable content, for example: an external file that define the header report, but that takes the name of the company (wich is variable).

```
<header border="1" bgcolor="RGB(125,125,125)">

<font face="Arial" size="14" color="#255255255">
<b><center>
Company Name, INC.
</center></b>
```

If this file defined your reports, when you got it to install with another client, you need to modify your external file, or if you have one for each report you need that change all your external files of each report.

However, we can define a variable part of an external file, whose value will be defined within our external program, or so that removes from some file or some another means

This same external file can be left:

```
<header border="1" bgcolor="RGB(125,125,125)">

<font face="Arial" size="14" color="#255255255">
<b><center>
%COMPANY%
</center></b>
```

This way, our reports used a variable, call %COMPANY%, which need be defined from our program of the following way:

```
<VAR COMPANY="COMPANY NAME, INC.">
```

Supposing that the company name we have in some data file of our application, then need that to arm the line of the following way:

```
STRING "<VAR COMPANY=' ", MEM-COMPANY, "'>"
      DELIMITED BY SIZE INTO WSS-LINE.
WRITE LINE-REPORT FROM WSS-LINE.
```

Placing remarks in the external files.

Sometimes it is necessary to put remarks in the content of an external file. This is possible using the character # in the first position of the line.

The line will be taking like commentary, example:

```
# This file contains the headers definition for the report
#
<header border="1" bgcolor="RGB(125,125,125)">

<font face="Arial" size="14" color="#255255255">
<b><center>
%COMPANY%
</center></b>
```

Global Config file COBVIEW.DEF

Now we can do special configuration for our cobview installation that affects all reports. Only put the special tags in the file:

COBVIEW.DEF

This file need exist in the cobview.exe directory installation.

This file will procesed before our report file.

You can put all that you need for all reports in your application.

Sample:

```
<SET UNITS="CMS">  
<SET MARGIN-TXT="OFF">  
<SET MARGIN-OBJ="OFF">
```

Set configuration values <SET>

With this tag we can change values in runtime only for one session, in order to modify the aspect of the report, or the configuration of Cobview.

<SET AUTODELETE="ON">

It causes that automatically the file generated by our program COBOL will be deleted after view or printed. If we put value to him OFF it left our file generated existing. This configuration alters the default value in the configuration window only for this time.

<SET SHOWCONFIG="ON">

It caused that the configuration button for Cobview, with which we will be able to accede to a Configuration form of Cobview. If we put value OFF the button will be hide. The other way to enter the configuration form is executing the Cobview with /CONFIG parameter .

<SET HEADER BORDER="nn">

<SET FOOTER BORDER="nn">

It allows to alter the value us of the edge defined for our HEAER or FOOTER, he is very util specially if the header or footer they are defined in an external file, and we want that a report in special takes or it does not take edges in header or to footer.

<SET DETAIL TOP="x" BOTTOM="y">

It allows to specify a space in target for the area of details, or in superior part TOP or the inferior part of page BOTTOM. Example:

```
<SET DETAIL TOP="0.03" BOTTOM="0.03">
```

It defines a space in target of 0.03 inches after of header and 0.03 inches before footer.

<SET UNITS="value">

This tag define the unit of measurement for our report. The values can be INCHES, CMS for centimeters, MM for milimeters.
The default is INCHES.

<SET AUTOLINE="off">

This tag can do that our report can't skip to next line with the CR char in our line. Then we need skip the line manually with the tag
 where we want.

This tag afect only the DETAIL section.

Default is ON

<SET COLUMN COLOR="color" BGCOLOR="bgcolor">

It defines an impression area with different colors from those from the report, for when we worked with columns. This will do that the details of the report are printed with varied colors, making see the details clearly. Example:

```
<SET COLUMN COLOR="BLACK" BGCOLOR="WHITE">  
<SET COLUMN COLOR="BLUE" BGCOLOR="TEAL">
```

If we placed these two lines before beginning the impression of body, will do that the first line appears with white background and black text, and the second blue sky background and text navy blue. And they would be to repeat the colors. We can have one mix of colors up to 5 details.

<SET COLUMN CLEARCOLOR="YES">

Clear the colors definition for several details with <set column color/bgcolor>, and the next columns details are printed in normal color, black/white.

<SET COLUMN LEFT="nn">

This defines the left position that was used in the columns of type AUTOLINE. The columns of type AUTOLINE automatically calculate their line, in agreement to the wide of the page, and advances automatically to the next line, but also we can specify in which position initiated to be placed these columns with this value. The value of nn is in inches, just as all those that the Cobview uses

<SET COLUMN AUTOSKIP="N">

With this tag the auto skip line is cancelled when print columns details. When print the last column defined in the report the cobview skip 1 line, with this tag this way is cancelled and cobview not skip 1 line. Then you need skip the line manually.

<SET COLUMN TOP="nn">

This define the top position for our column titles in every page.

<SET MARGIN-TXT LEFT="ON/OFF">**<SET MARGIN-OBJ LEFT="ON/OFF">**

With this values, can turn off the margin left, specially if you want print in absolute positions in your report. Special for print invoices or another format preform. MARGIN-TXT is for the text in the report including columns, and MARGIN-OBJ is for the objects like IMG, DRAW and BARCODE. By default are ON.

<SET PRINTER="Printer name to use">

Print automatically in the specified printer. This will that Dialog print selection not showed.

<SET PRINT="yes">

It causes that a visualization of the report is not made in screen, and that prints the direct report to the printer. Only it showed a progress bar.

<SET PRINTDIRECT="yes">

It does that report will printed the direct to printer. It did not show the dialog selection printer, is used the default printer, or the specified by <SET PRINTER>

<SET COPIES="n">

Set the number of copies for our report, when or report is send direct to printer.

<SET OEMCHARS="yes">

Allow do the characters conversion where the chars like Ñ or another special char can be printed in Windows, specially when the data are created using MSDOS compilers. By default OEMCHARS is NO.

<SET OEMEXP="yes">

Allow do the characters conversion for exponent numbers only.

<SET PAGE TITLE="Page" TOP="n" LEFT="n">

Set the title and position for print the page number.

Example:

<SET PAGE TITLE="Page" TOP="0.80" LEFT="7.2">

Only accept absolute positions in the page.

<SET ZOOM="n%">

Set the initial value for the ZOOM preview for cobview. Only area valid the values in the list box in the preview mode. By default the preview is set in 100%.

Sample:

<SET ZOOM="60%">

<SET DECIMAL-POINT="char">

Set the decimal point for build the correct mask in the clause PICTURE, in the instruction PRINT.

Also is need for extract the numeric value in the column totals.

The default value is .

<SET THOUSAND-SEPARATOR="char">

Set the correct thousand separator for work with variables, PRINT with PICTURE

Sample:

```
<SET THOUSAND-SEPARATOR=".">
```

The default is ,

<SET SEPARATOR="char">

Set the column separator for ASCII exportation.

Specially used when the exportation is made since the command prompt.

Chars valids: , | ; . :

<SET Y="n">

Set the new value for the Y position in the page. Y is the control for print in any position in the page. The command
 skip 1 line, depending on the current font size.

The n value can be:

“+n” – skip n inches down from the current position

“-n” – skip n inches up from the current position

“n” – set Y to one position fixed in the page

Notes:

If this tag is placed in the last line of the header section, automatically in every page, after print your headers page the Y position is re-defined.

If this tag is placed inside the <band> tag, every time that is printed the band, the Y position is redefined.

<SET X="n">

Set the new position for the X position in the current line.

Every time that you print a text the X position is moved to end of text printed or in begin of line if a line skip happened.

If you wish do a special print, you can do with this tag and after print your text.

Example:

```
<SET Y="+0.25">  
<set x="2">Text one  
<set x="+0.30">Text two  
<set x="4">Text Tree<br>
```

In this sample, first position the Y from the current position plus 0.25 inches, After print **"Text one"** in the 2 inches left position. Advance 0.30 inches to right and print **"Text two"**, later position in the inche 4 and print **"Text tree"**, at the end skip of line, and the X is positioned in the 0 inches left.

Example 2:

```
<SET Y="+0.25" X="2">Text one<br>
```

In the same tag is defined the new position for Y, and the fixed position for X. Later print **"Text one"**, and skip to next line.

Remember that the position is only temprally, because after print text, the position X or Y is moved to the end of text or the next line.

You can define tempraly also, a width and alignment for temprally impresions.

For width definition use W="n", where n is the size in inches.

For alignment use A="a", where a can be L, C or R. (Left, Center, Right)

```
<SET Y="+0.25" X="1" W="1" A="R">15,670.00<br>
```

In the previous sample set the Y position in the next 0.25 inches, and print the value 15,670.00, in an area of 1 inche specified by W, and aligned to Right.

Initializing vars when page change <INITIALIZE>

Is possible to initialize a variable after this has been printed, and before beginning to totalize the columns in the next page.

Format:

<INITIALIZE variable VALUE="newvalue">

With this is possible to place the value to the variable.

It's necessary to have well-taken care of to place tags INITIALIZE after the PRINT.

Sample:

<INITIALIZE VAR-TOTAL VALUE="0.00">

This tag must be inside the tag <FOOTER> ... </FOOTER>. And one would be due to initialize the variables after printing them to begin in zeros in the new page.

Events in report parse <EVENT>

Event AfterNewPage

It happens when the page jump to create a new blank page.

```
<EVENT AFTERNEWPAGE FIRSTPAGE="Y/N" AUTOLINE="Y/N" PAGES="no. pages">
... print tags
</EVENT>
```

Parameters:

FirstPage – Indicate if you want that the event execute in the first page too.
Default: Y

AutoLine – Indicate if we want execute a
 tag in every line printed. If we want controller the break line with
 tag manually, we will put Autoline="N"
Default: N

Pages – With this parameter you can specify in wich pages you want active the event, the value can be separated with the symbol (,) to specify individual pages, or the symbol (:) to specify pages range, example: **pages="1,2,3,10:20"**
The event execute in the pages 1, 2, 3 and from 10 to page 20.
You can also specify only one value in the range, sample: **pages="3:"**
In this way, the event is executed from page 3 to the end of report.

Notes:

You can specify any valid tag into block <event> and </event>.
The tag <newpage> skip a page, but not generate a new event AfterNewPage.
You can specify two or more events AfterNewPage, and this events are executed in the order that they appear, maebe for they could be to execute in diferent pages. Sample:

```
<event AfterNewpage pages="1">
</event>

<event AfterNewPage pages="2:">
</event>
```

In the previous sample, the report have two events AfterNewPage, the first event is executed only in the page 1, and the second event is executed from page 2 to the end of the report.

With this event you can specify Report Titles, that appear only in the first page of report, or Report headers that appear in diferent pages from the report.

Cleaning defined Events <CLEAR EVENT>

With this tag we can indicate that disable events that are defined in our report until this moment, and that they are working.

We can use by sample, if we can defined an event for print titles in every page, but we need change our page title, printing another texts.

Sampleo:

```
<CLEAR EVENT=""AfterNewPage">
```

This sample, clear all events AfterNewPage defined until now.

If we wished start to use anoter event, only do the declaration with a new EVENT mark, for start to work.

Checking free space in the page <CHECK>

With this tag can validate the free space in the page for print several lines complete.

Format:

```
<CHECK FREESPACE LINES="n">  
<CHECK FREESPACE SIZE="n">
```

Parameters:

Lines – indicate the lines that you need for print the text. The height for every line is taken of the current font.

Inches – indicate the size in inches that you need for print the text.

Sample:

```
<check freespace lines="5">
```

```
<check freespace size="1">
```

in the first sample we need 5 lines for print our content.

In the second sample we need a size of 1 inch for print our content.

If not exists this size in the current page, automatically jump to next page for print our content.

Sending Email Data <EMAIL >

This TAG is for pass the Email data to COBVIEW for send the report by Email. Such a way that wen we do click in the button “**Send by Email**”, the data like TO or SUBJECT are typed.

Format:

```
<EMAIL TO="email@domain.com" >  
<EMAIL CC="email@domain.com">  
<EMAIL SUBJECT="EMAIL TITLE">  
<EMAIL MESSAGE="Email content message">
```

Parameters:

TO – Indicate the email account, for send our EMAIL Report.

CC – This parameter for indicate an email accout for send a Email copy.

SUBJECT – This is the Title for our email.

MESSAGE – Is for send the initial content message to Windows Send by email.

Watermark protection <WATERMARK >

With this TAG, we can pass the data for the Watermark protection.

The Watermark is a display protection in the page background.

The Watermark is for avoiding duplicity, copies and for indicates a confidential report.

Format:

```
<WATERMARK TEXT="TT" DIRECTION="n" TRANSPARENCY="n" PAGES="n">
```

Parameters:

TEXT – Is the Text for protect the pages.

DIRECTION – Is the directio for print the Legend in the background page:

- HORIZONTAL – Text is printted in horizontal form
- VERTICAL - Text is printted in vertical form
- BACKWARD –Text is printted in slash, up to down
- FORWARD – Text is printted in slash, down to up

TRANSPARENCY – It's a range sinze 0 to 10, 0 is a solid text, 10 is a transparency text.

PAGES – It's the number pages where apply the watermark. The "ALL" value indicate to every pages. Samples:

1. PAGES="1,2,5"
2. PAGES="10-20"

In the sample 1, especify separating with commas the values, where we want applies.

In the sample, we specify a rank, separating the initial page from the end with a script.

The "ODD" value, indicate that print only odd number pages.

The "EVEN" value, indicate that print only even number pages.

Web Reports Data <WEBREPORTS >

With this TAG, we can pass the data for the WebReports form.

Several Web Reports data can be configured with this tag.

Format:

```
<WEBREPORTS SECTION="Sect-Name" DESCRIPTION="Report Description">
```

Parameters:

SECTION – Indicate the Section name, where we can store our report. This section can only be configured in the Reports Server program.

DESCRIPTION – It's any description to store our report in the Reports Server. Can be any open description. Sample: "SALES FROM NOV/2005"

Notes:

For more about Web Reports see:

- Email/WebReports Configuration
- Command Line Parameters

Search text in the report

Now Cobview can do search for find any text desired.

Just do click in the icon “Search Text” or push the F3 key.

The dialog box for search need the following data:

Text to search

Ignore case, mark this check-box if you wish find text in lower case and upper case.

Mark text found- with this checkbox, you can paint the words found, the text is marked with the color indicated in Background y Foreground.

When push the button “Search” begin the search. When finish the search automatically go to first page with the text found.

In the following ocassion that you push F3, you can see the result list, and click any page with results.

Notes:

If you print the report after any sarch, and marked the text, all text founded will be printed with this characteristic, with the background and foreground as you see in the screen.

Exports

Cobview counts on different types to export our reports.

The exports can be done by means of the line of commands with the parameters for export or when we are visualizing our report with the button: Export.

The allowed types of export are:

ASCII delimited

It generates an ASCII file delimited, when doing an export of this type solicited to us that we indicate the character that we will use like delimiter of columns as well as the text delimiter.

When one becomes an export of this type by means of the line of commandos is used by default the comma (,) like delimiter of columns and quotes (") like text delimiter. In this type it is not possible to choose initial and final the pages that we wished to export, since it exports all the complete report.

Excel Sylk

It generates a compatible file XLS with excel, is not necessary to have the Excel program installed in the computer where the export is generated. This type either does not allow to select the pages to export, makes the export of all the complete report.

PDF

It generates a file pdf that can be visualized with the Acrobat Reader. This type of export generates an image from the original report as it visualizes with COBVIEW.

BMP Image

It generates a series of BMP files, generates a file of image by each page, to the name of the file that we choose is added to him the number of page to have a series of image files with the numbered page.

JPG Image

It generates a series of JPG files, one by each page just as in BMP

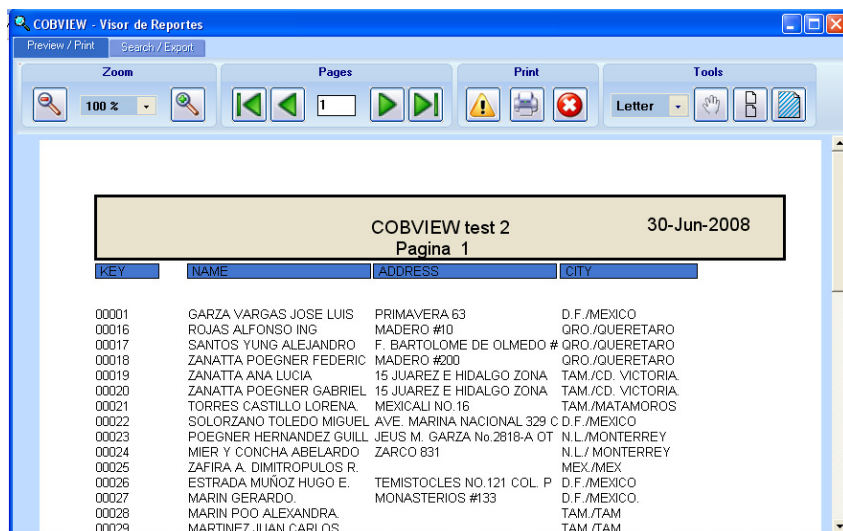
HTML

A file makes an export yet towards HTML generating the report, and can be visualized with any internet navigator. This file does not include the images that we have in our report. If we used images the best way to make an export it is with PDF, BMP or JPG

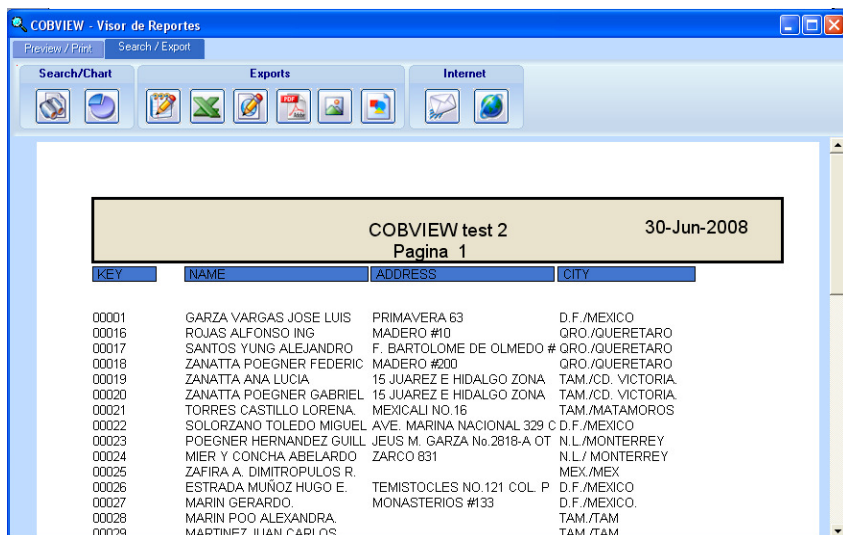
COBVIEW.EXE – The report viewer

Next we explain the several components of the COBVIEW reports preview.

Basically the viewer analyze the file content, with the special COBVIEW tags, build and render all report pages.



Have two tool bar, that can be accessed by a Tab components: **Preview/Print** and **Search/Export**.



In the **Preview/Print** tab are allocated the next buttons:



This button allow do Zoom in the report.



Allow do Zoom out.



Go to first report page



Go back a page in the report



Go to next page in the report



Go to last page in the report



Show an about form for the COBVIEW program.



Print the current report, allow select the printer



Close the COBVIEW program



Activate the hand tool, for move freely in the page



Activate the Multipage tool, for browse easy in the report



Show the form to configurate the Watermark

In the Search/Export tab, are allocated the next buttons:



Tool for seach text in the reporet



Open a form for draw charts with the data report



Export the current report in text format



Export the report for the Excel program



Export the report in HTML file



Export the report to PDF format



Export the report to BMP format, 1 file for page



Export the report to JPG format, 1 file for page



Open the form to send the current report by email



Open the form to send the current report to WebReports.



This icon is for acces to Reports stored in our Reports Server.

Charts with data reports

Now with Cobview it is possible to generate charts from the information contained in our reports.

The data that we needed to indicate are:

General:

Graph Title – I title that we want that appears in our chart.

Column for legend – Here indicate the column that we will use like legend of the data of our chart.

Data Series

Here we needed to select the columns that we will use like data serie. The column that already chose is the column for legend is not necessary to select it again.

Data Ranges

For complete we needed to indicate the ranks of data that we wished to use to arm our chart. This is mainly so that probably we have a too long report and we wished in the chart only a section of data.

Once the previous data are indicated our chart show already armed. Show automatically as bar chart, using toolbar of the Chart viewer we can change the type of chart, also like investing the legend by the series of data to see the grafica of another different way

.

COBVIEWPDF

This external program, was built for do conversions since COBVIEW reports to PDF files.

Although is possible build a PDF file since the command parameters using the COBVIEW.EXE program, this was created for do only this work.

The main use of this program is in the COBHTTPD webserver, another program too from Cobtools, but now CobviewPDF is possible use since any program, from this now is part from COBVIEW Print Suite.

Format:

Cobviewpdf report.txt /pdf filereport.pdf

Parameters:

/PDF filename

Indicate the file name for the PDF that we need generate, since our COBVIEW report.

/FROMPAGE page

Indicate the initial page since we need start the exportation.

/TOPAGE page

Indicate the final page until we need the exportation.

/NODEL

With this parameter, we can indicate that not want delete the original report file

COBVIEWCGI

This program too can be used for do conversions since a COBVIEW report to any PDF file.

This program can be called since any webserver, because this program not access any GDI resource, and the result PDF file can be showed in the web explorer after conversion.

This program was built like a Console Application, and can be used like a gateway between the Webserver – Cobol and build your own CGI programs for build PDF reports.

The main difference between COBVIEWCGI and COBVIEWPDF is that COBVIEWCGI is a Console Application and COBVIEWPDF is a Windows Application.

Format:

Cobviewcgi report.txt /pdf filereport.pdf

Only we need indicate the /PDF parameter, followed of the PDF filename to generate.

COBSPOOL-WIN Print Server for Windows Lan

This program is a Spooler for build Print Queues, and thus to be able to send our COBVIEW reports to any printer in the Lan.

This printers can be Dot matrix printers, Lassers, Inkjet, even USB printers.

This program in our Directory programs is called SpoolWIN.EXE

The main idea is have a several directories in the computer, that can be shared in the lan. Every directory it's a Spooler folder, everything that can be placed in this directories is printed in any LAN printer, depending on the Spooler configuration.

Only need one computer in all LAN that can woprk like Print-Server, sharing your Queues folders, and sending to print any file placed here.

Once that this program is executed, work without human intervention, only if any error is produced when working.

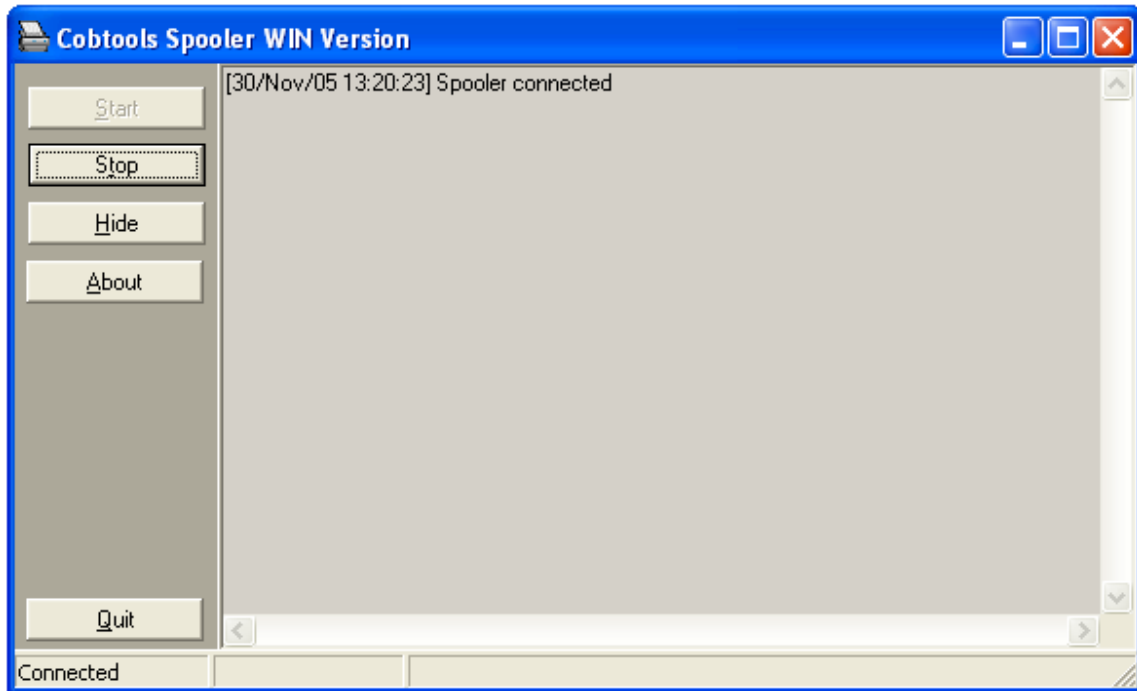
All configuration is founded in the Control Panel for the COBVIEW Print Suite.

Depending for the program configuration, will be executed automatically, and be allocated in the Windows Systray.



If the spooler is not configured for run from this way, the main windows is showed.

If the program was allocated in the systray, we can open the main windows doing dbl-click over the Icon.



This window is showed once that the program is runing, and opened.

Buttons function:

Start – Start the Spooler function.

Stop – Stop the Spooler function, although the program follow runing the spooler function is stoped.

Hide – Hide the main window, and the program run in the Systray.

About – Show the About window.

Quit – Stop the spooler and close the program.

COBSPOOL-FTP Print Server over Internet

This program is for build a Printer Spooler, but connecting to any FTP server, it can be any UNIX server, LINUX, or any other computer that have installed a FTP server.

Actually can be conected only a one FTP server, y can have several printers configured.

The printers can be locals to the computer that work like Print-Server, or connected in the LAN where will print the reports.

We can connect differents Print-Servers to our FTP server, even since diferents cities, with the purpose of send reports to several offices connected by TCP/IP or internet with any Central office.

This program in our Programs directory is called SpoolFTP.EXE

Once that the program is executed, work without human intervention, only if any error occurs can be need that the Administrator indicate something to resolve the problem.

All configuration can be made with the Control Panel for the COBVIEW Print Suite.

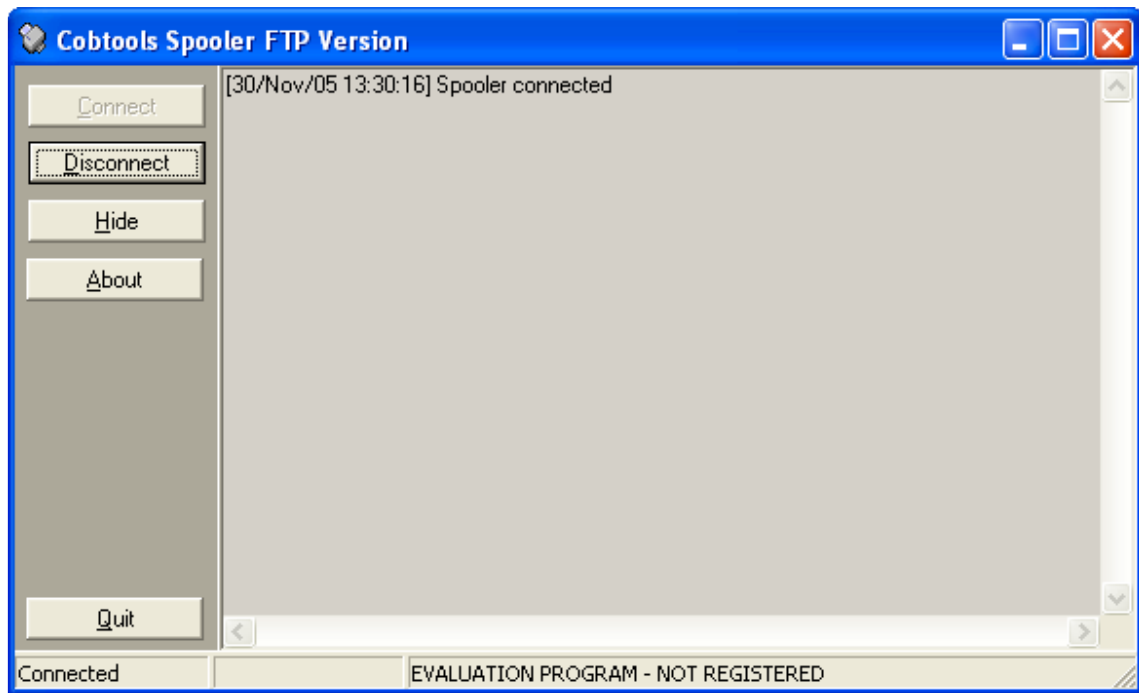
Very important is that our Print Server can be connected with the FTP server with a System User, for this reason the username for the connection need have rights over the directories that work like Print Queues.

Depending on the program configuration, can be executed automatically and be allocated in the Windows Systray.



If the program is not configured for allocate in the Systray, the main window is opened when the program start.

If was allocated in the Systray, we can open the windows doing dbl-click over the Icon.



This screen is when the program is running.

The buttons function are:

Start – Start the Spooler function, if is stoped.

Stop – Stop the spooler function, the program can be running, only the spooler function is stoped.

Hide – Hide the Window program, and the program run over the systray.

About – Show the About window.

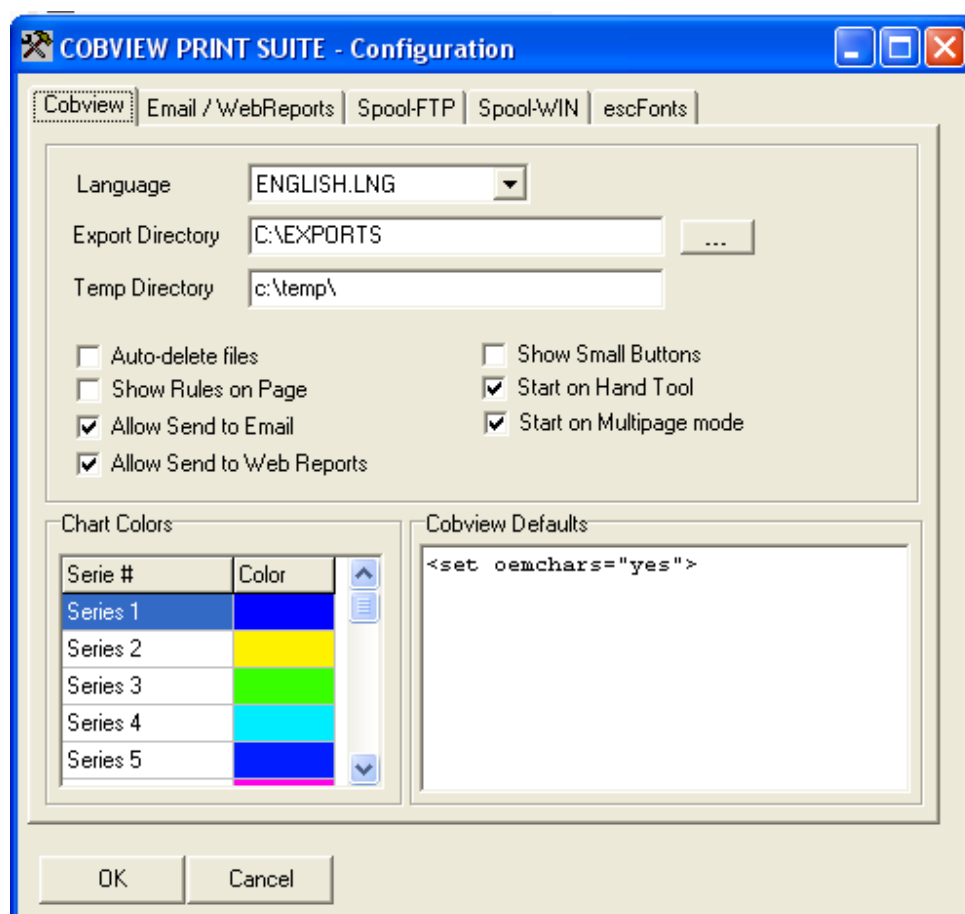
Quit – Stop the spooler and close the program.

COBVIEW Print Suite Configuration

Now in this new COBVIEW version, all configuration is made with a external program, and configure all tools for the suite.

This program is: ViewConfig.EXE

When this program start, the next window is showed, every folder is for any program.



COBVIEW.EXE Configuration

Language – It allows to form the language in which we want that it shows to the messages and titles to us of the different controls from the Cobview.

Export Directory – This is the directory where we want that it lodges the file export when the file is not specified drive and path to export.

Temp Directory – This directory where will be created the temprary files, that the program need for send by email or to Web-Repotrs.

Autodelete Files –Automatically erases the files that already were visualized

Show rules on page – It allows to show rules in the page, to be testing and to see as place the data in fixed positions.

Show rules on page – Permite mostrar unas reglas en la pagina, para estar haciendo pruebas y ver como colocar los datos en posiciones fijas.

Allow Send to email – This data is for indicate if we want activate the Send by email function. For this function is necessary that the SMTP server configuration in the folder *Email/Webreports*.

Allow Send to Web – With this data, you activate the function for send reports to Web-Reports service, is necessary too configure the service in the folder: *Email/Webreports*.

Show small buttons – This checkbox allow build small buttons on toolbar preview.

Start on hand tool – If check this option, when start preview, the hand tool is ON.

Start on multipage tool – If check this option, when start preview start on Multipage preview mode.

Chart Colors

Now is possible configure the colors that we want use inthe charts generation. Before only we have 16 colors diferent, now we can define the 200 colors, by default have already defined 50 colors, but is possible modify the this colors or add the next 150.

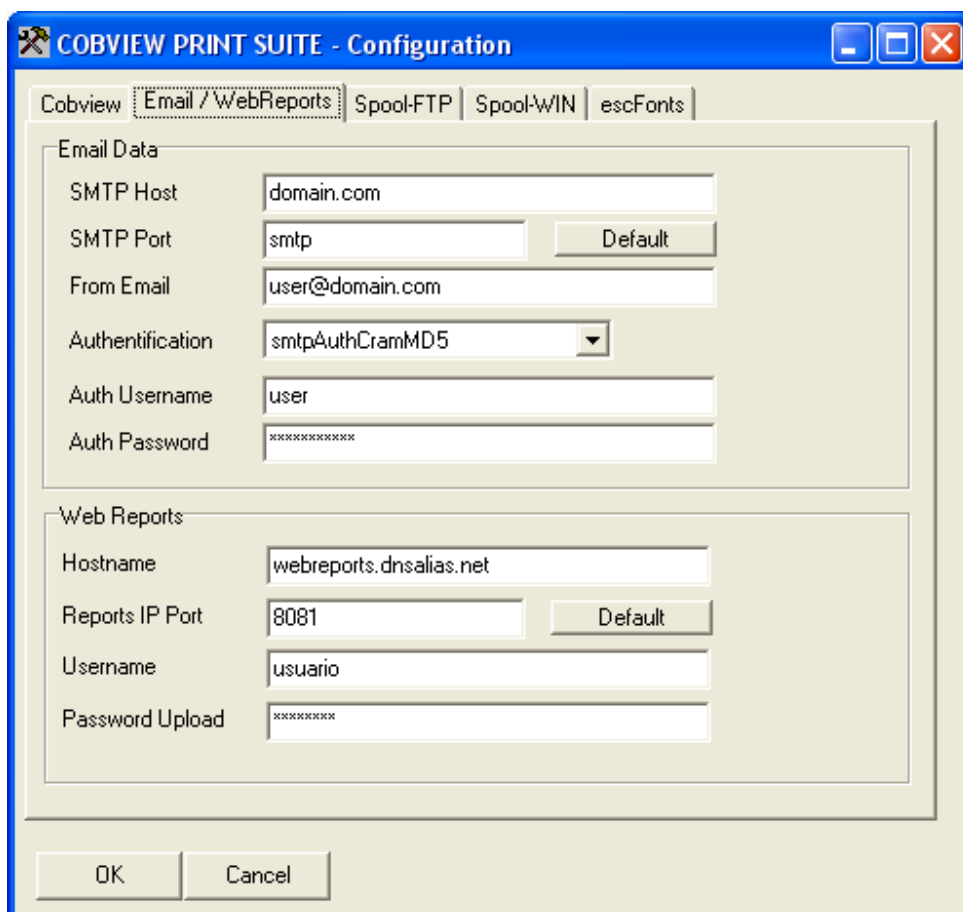
These colors are saved in the file COBVIEW.DAT.

Cobview Defaults

Here we can insert defaults TAGS for all reports.
These defaults are saved in the COBVIEW.DEF file.

Email/WebReports Configuration

In this folder can be configured the functions for Send emails and Send to Web-Reports service.



The screenshot shows the 'COBVIEW PRINT SUITE - Configuration' dialog box with the 'Email / WebReports' tab selected. The dialog has a blue title bar and standard Windows window controls. It contains two main sections: 'Email Data' and 'Web Reports'. The 'Email Data' section includes fields for SMTP Host (domain.com), SMTP Port (smtp), From Email (user@domain.com), Authentication (smtpAuthCramMD5), Auth Username (user), and Auth Password (masked with asterisks). The 'Web Reports' section includes fields for Hostname (webreports.dnsalias.net), Reports IP Port (8081), Username (usuario), and Password Upload (masked with asterisks). Both sections have 'Default' buttons next to the port fields. At the bottom are 'OK' and 'Cancel' buttons.

Section	Field	Value
Email Data	SMTP Host	domain.com
	SMTP Port	smtp
	From Email	user@domain.com
	Authentication	smtpAuthCramMD5
	Auth Username	user
	Auth Password	*****
Web Reports	Hostname	webreports.dnsalias.net
	Reports IP Port	8081
	Username	usuario
	Password Upload	*****

Send reports by EMail

Now can send any report that you are seeing with the preview to any Email account, the report is sended in PDF format, this function only require that the computer where the report will visualized have the PDF viewer program installed, not need have COBVIEW installed.

Even in this folder is made the configuration for the email connection, it depends on if is enabled or disabled the function for send emails in the COBVIEW folder.

The data that need configured for work sending emails are:

SMTP Host – The domain name for use in the SMTP funcionts.

SMTP Port – IP Port for do the connection with the SMTP host.

From Email – Indicate the account since the email is sended. Need be in the form:
[username@domain](#)

Authentication – If the Domain where will be sended the email need auth data, here need specify the method that use the Email Server. This data depend of the Server configuration, and have the more used.

Auth Username – Is the username wich be used for do the authentication with the Email server.

Auth Password – Password for the authentication.

Some data need be obtained with the Email administrator, or with the support for you ISP.

Web-Reports Service configuration

Abouts Web-Reports.

Web-Reports is a special service that is executed in a computer which we will call Server. This Server, would be in charge to store our reports for a later preview or print.

Our Reports Server (Web Reports) can be created installing the cobWebServer program, is an application also of COBTOOLS.COM

Actually COBVIEW can preview and print Reports. Now with this new function, also can preview and store reports in a Reports Server.

The reports can be previewed in any User terminal, as commonly it's made until now, and later can send this reports to your Reports Server (Web Reports).

Also we can do that, our reports automatically are sent to you Reports Server, thanks to new TAGS. And automatically are stored this reports in any special section, and with any Description sended by your COBOL program.

Your Reports Server can even in another city, and COBVIEW can send your reports by Internet to store it.

This can allow us that, any company with offices in severa cities can store your reports in the same Reports Server over internet, with this program installed.

For the later preview this reports we can use COBVIEW to connect itself in the Reports Server, and consult the old reports.

At the time of installing our application in any customer, we only have to configure the Server name, Username and Password for upload the reports.

The data that need configure are:

Hostname – Is the IP address or Internet name to allocate our Reports Server.

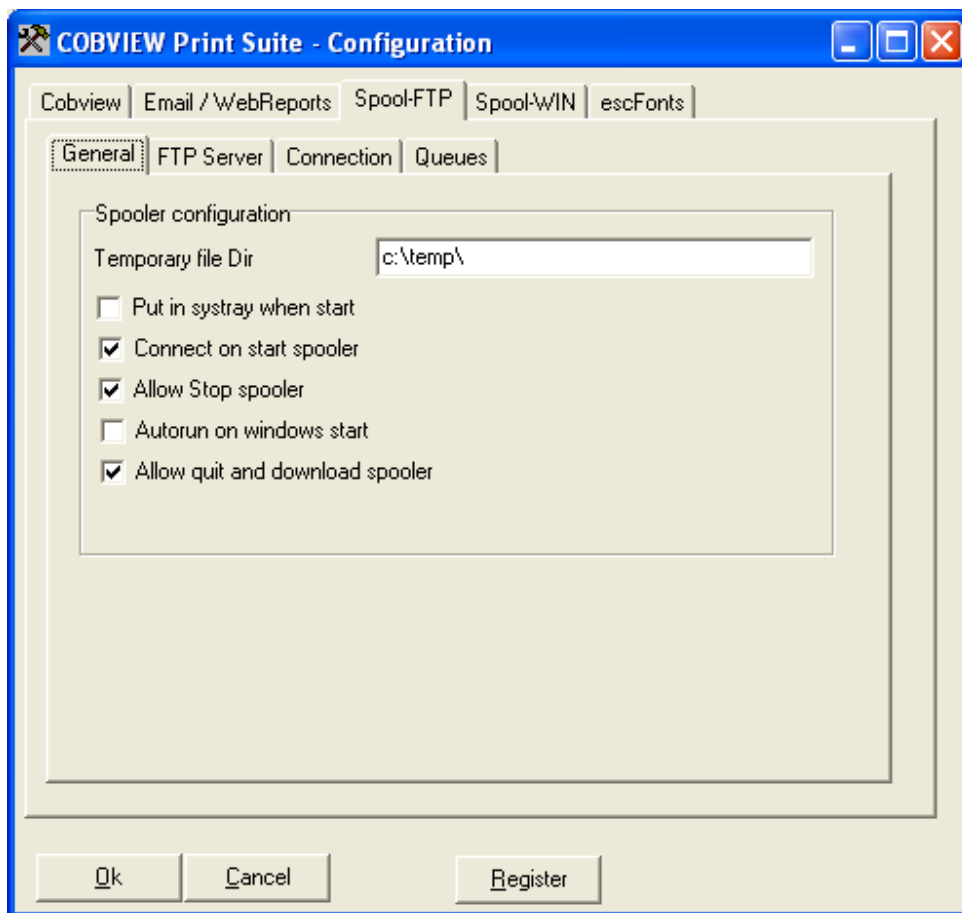
Reports IP Port – Is the TCP/IP port for connect with our Server. By default, the port is 8081, but we can configure any other in the Server and COBVIEW.

Username – Is the Username for connect with our Reports Server. This User and Password need be configure in the Report Server before.

Password Upload – This password is for allow upload reports to Web-Reports.

Configuracion de Spool-FTP

In this folder can configure the Spool-FTP program, that allow create a Print Server using any FTP server computer.



General:

Here we have the main configuration for run Print Server.

Temprary file Dir – Local directory for download remote files and print in the local printers.

Extension Filter – Is a filter, for print only this file types, sample *.txt

Put in Systray when start – When start the spooler program is allocated in the systray, in the task bar

Connect on Start Spooler – Auto connect when the program run

Allow Stop Spooler – Enable the button in the main screen for stop the spooler, when the user need work in the computer.

Autorun on Windows start – Active the spooler before the user logon in the computer

Allow quit and download Spooler – Enable the button for quit from the spooler program, if you disable this option the program never can be stoped

FTP Server

This internal folder contain the configuration data for connect to FTP Server.

FTP Server – Indicate the FTP server to connect, for extract the report files.

Port – The tcp port to connect with the FTP server. By default the ftp port is: 21

Username – Is the valid user in the ftp server for connect by ftp sessions.

Password – The password for connect the user

Timeout – Timeout connections, for error control in transfer operations.

Spooler timer – Is the amount seconds for check exist new report files

Autodelete files after printed – Delete the server files after printed

Only preview report – If check, then when download a report, is visualized, and not printed.

Connection

This internal folder configura the way for do the connection, is used when we have any proxy server in our lan.

Connection type – Indicate the method for do the connection with the host. The valid options are: FTP Direct / Proxy / Sockets 4 / Sockets 4a / Sockets 5.
When you change the option, automatically enable the options needs for the new selection.

Proxy Server – Is the proxy server name or address for the proxy connection.

Server – Is the server for do the connection by Sockets option. For the firewall connection.

Port – Port number for the proxy server.

Username – The username for the Socket server connection.

Password – The password for the username in the Socket server.

Queues

In this folder can be configured the remote directories that can be used like printer queues.

We have firstly tree main buttons, that allow us create, modify or delete any queue.

Add – Create a new printer queue.

Modify – Modify the curent selected queue in the queues list.

Delete – Delete the current selected queue.

Data need for create or modify any queue.

Queue Name – Is the name for the queue, is onyl for reference, and show in the queues list

Directory Queue – Is the remote directory, in the ftp server where find the files to print in the local printers.

Local Printer – Is a list with the printers, you can choose one, for indicate where print the jobs.

ESC file definition – Is the filename for the Escape conversions, this file es defined in the **EscFonts** tab.

This options is used for text reports conversion to windows fonts.

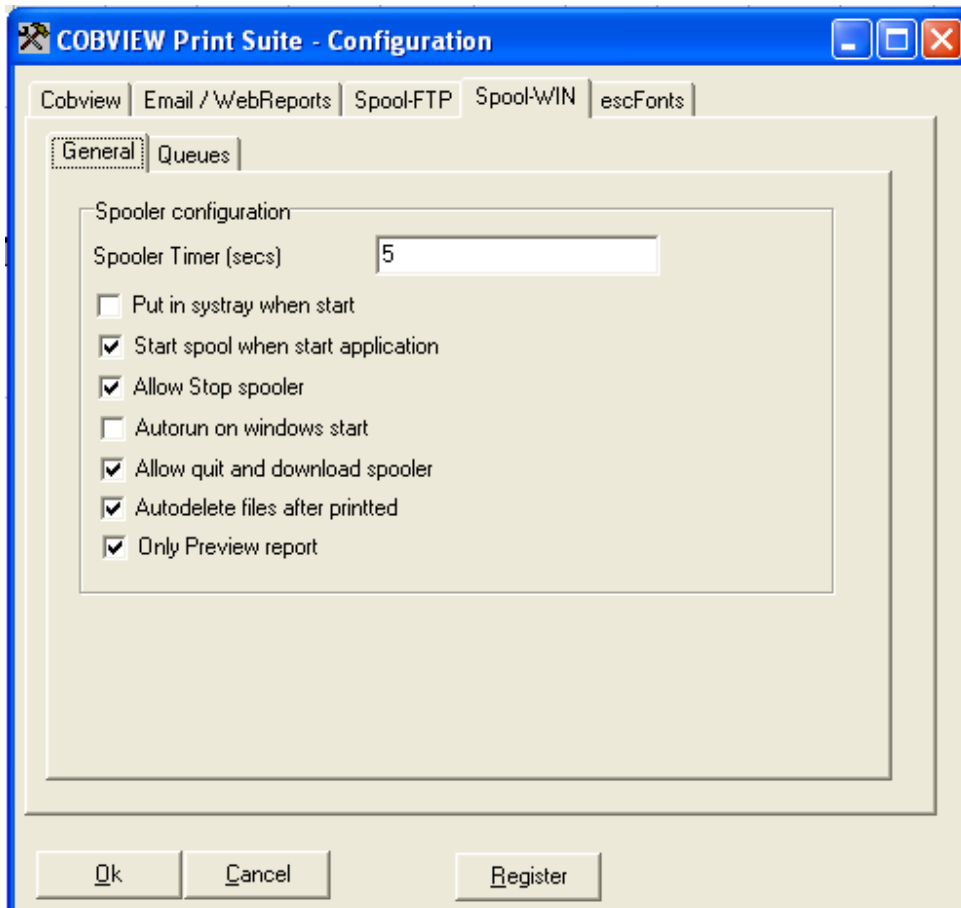
Once that all data are filled, we can save the changes clicking the **Save** button.

Or if we want cancel all changes, we can do click over the **Cancel** button.

Spool-WIN Configuration

This program allow us create a Print Server, using our own Windows lan.

Basically this program work using a Shared directory resource like Printer queue, and everything that we save in this shared folder is printted in the configured printer for do it.



General

In this tab is found all data need for indicate the way like the Print server work.

Spooler timer – Is the ammount seconds that the program check if any new file is found in the shared folder before print it.

Put in Systray when start – When start the spooler program is allocated in the systray, in the task bar

Start spool when start application – Autostart the spool process, when the program run

Allow stop spooler – Enable the button in the main screen for stop the spooler, when the user need work in the computer.

Autorun on windows start – Active the spooler before the user logon in the computer

Allow quit and download spooler – Enable the button for quit from the spooler program, if you disable this option the program never can be stoped.

Autodelete files after printed – Delete the files after printted

Only preview report – With this data we can do that COBVIEW preview the report, and not send direct to printer.

Queues

In this tab will configure the local directories that be used like printer queues. The data that need be filled are the same that the Spooler-FTP.

Contamos primeramente con tres botones principales, que nos permitiran crear o modificar alguna cola de impresión.

We have firstly tree main buttons, that allow us create, modify or delete any queue.

Add – Create a new printer queue.

Modify – Modify the curent selected queue in the queues list.

Delete – Delete the current selected queue.

Data need for create or modify any queue.

Queue Name – Is the name for the queue, is onyl for reference, and show in the queues list

Directory Queue – Is the local directory where the reports will be allocated, sample:
c:\queues\hplasser.

Local Printer – Is a list with the printers, you can choose one, for indicate where print the jobs.

ESC file definition – Is the filename for the Escape conversions, this file es defined in the **EscFonts** tab.

This options is used for text reports conversion to windows fonts.

Once that all data are filled, we can save the changes clicking the **Save** button.

Or if we want cancel all changes, we can do click over the **Cancel** button.

EscFonts

This tab allow us to create Definition files for Escape sequences.

Escape Font file it's a special conversion between traditional escape sequences and windows fonts, for the appropriate font size in the windows printer.

COBVIEW Print Suite have two Escape files, EPSON and HPLASSER like samples.

Every file have defined data that are fixed:

Report file lines – Is your report text size in lines. Traditional are 66 lines. This data must be exact for the correct operation. By sample in the EPSON file we have defined 66 lines.

Report data lines – Is the data size in lines, for your text reports. Normally, in text reports we print 60 lines by example and 6 lines are for the skip page.

Default Orientation – Default windows orientation for the reports. Portrait or Landscape

Default page size – Default windows paper size, the sizes are: Letter, Legal or A4

The escape sequences and the font are defined in separate mode, can have a name for distinguish them.

When the program run automatically load the files definition in the combo indicated in ESC file.

In a side is found tree buttons that have the next functions:

New – Allow create another new escape file sequences

Open – Select and open the current file in the combo box.

Close – Close the current open file

Sequences and fonts definition

Once that we have the file opened, with NEW or OPEN, the next buttons are enabled:

Add – Allow a new sequence

Modify – Modify the current sequence selected.

Delete – Delete the current sequence selected.

The data need for add a new sequence are:

Name – Name for identify the escape sequence Samples: DEFAULT, CONDENSED, LARGE

Sequence – Is the escape sequence that find in the report text. the format for special characters is: \xNNN
samples:

\x015 – secuencia tradicional de Epson para reducir la letra

\x027@ - secuencia de Epson para resetear la impresora

Font – This button allow select the font that we want use for this sequence.

In the data Font Sample show a sample like how can be viewed when use this sequence in the cobview report.

Notes:

Because a definition file is used in any special queue, for FTP or WIN, all reports that send to this queue will be printed using the same page configuration and escape conversion.

Wich means that all reports need be defined with the same specification, with the same lines amount in the report, lines for page footer and the same escape sequences.

We can't mixed reports in any same queue with reports definitions diferents, by sample, if we have a queue using a EPSON printer, we can't print reports using sequences for HP printers.

COBVIEW License register

Now for register your COBVIEW copy only need copy the license file that we send by email to you, after your registration operation.

COBVIEW Print Suite is a comercial version, because no have a several tools for doing printing works.

You can distribute this license file and COBVIEW programs with your applications.

If you want buy your own license for using COBVIEW Print Suite, visit the site: www.cobtools.com where you can fond more information.