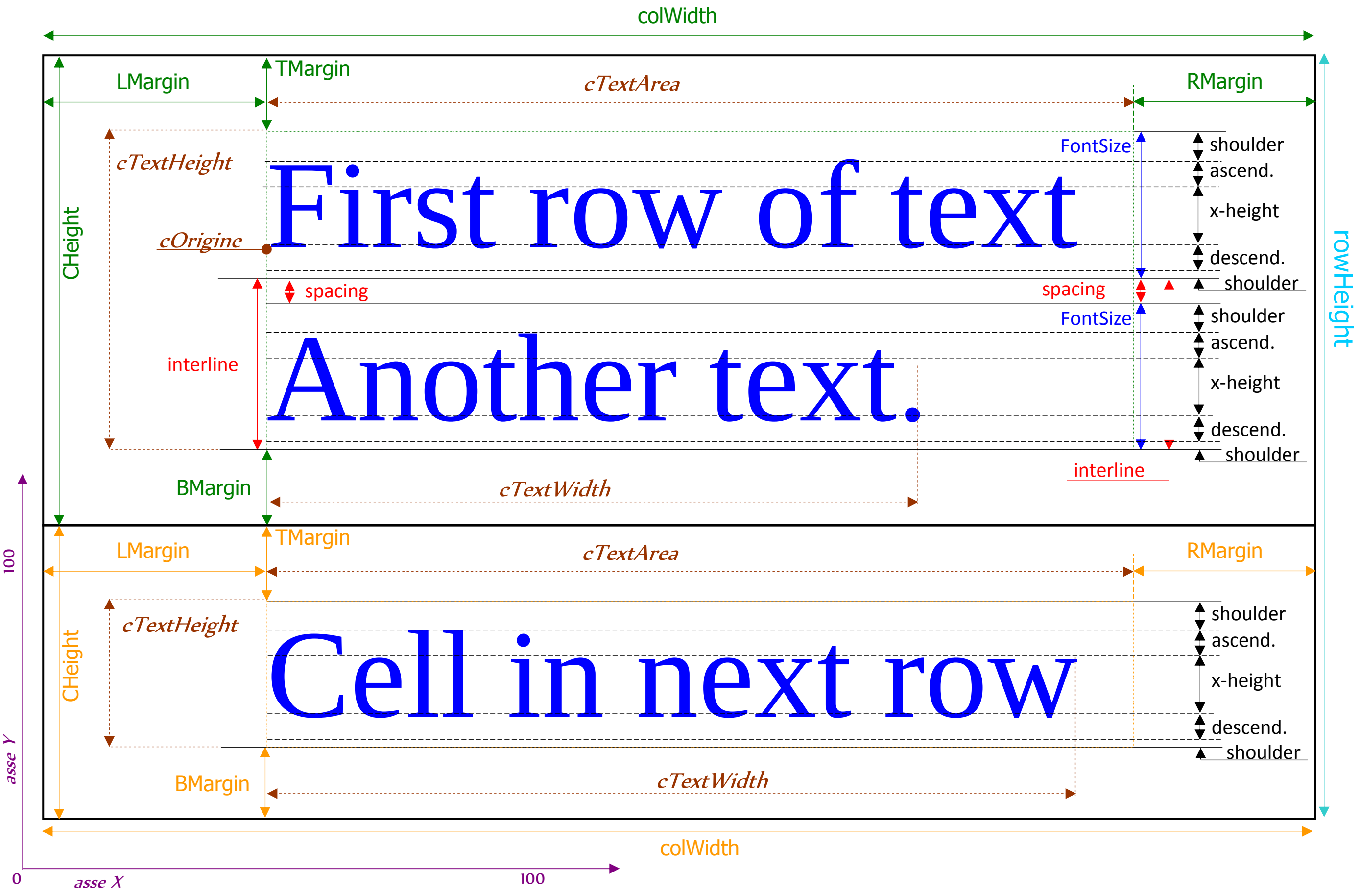
AS_PDF3_V5 Formatting parameters

V 3.5.02



The coordinate system of PDF document is a cartesian plane with origin in the lower left corner of the sheet.

The measures are indicated with differnt colours:

green for the first row cell, **orange** are for the second row of cell, **light blue** for the entire record, ***brown*** for the text, **red**, **blue** and **black** are used for font.

The package performs the calculations of dimensions in points (1 pt = 1/72 inch), but you can specify the parameters supplied to the package using the unit of measure defined with the parameter p_um..

Valid units of measure are mm, cm, pt (or point), in (or inch), em (or pica, pc, p)..

TYPE tp_Column (Record with columns property) **tp_Columns** (Table of tp_Column)

The prefix **col** is relative to entire column

colLabel	Column Label
colWidth	Column width, measure unit as specified by p_um parameter
cellRow	Cell Row in Multirow Record. No missing rows are allowed. (0,1,2,3,...) or (1,2,3,...) NOT (0,2,3,..) Usually at every change of line you set offsetX = 0
offsetX	Force X position of cell, default is calculated from previous cell
offsetY	Force Y position of cell, default is calculated from previous cell. Indicates how many um moves down from top border of record row .

The prefix **h** is relative to the Header.
The prefix **t** is relative to the Table of data

Header	**Table**	
hFontName	tFontName	Font Name
hFontStyle	tFontStyle	Font Style (Bold/Italic/Normal)
hFontSize	tFontSize	Font Size (in points)
hFontColor	tFontColor	Font Color (ink) in RGB mode
hBackColor	tBackColor	Back color (paper) in RGB mode
hLineColor	tLineColor	Line color (border) in RGB mode
hLineSize	tLineSize	Line size (Thickness) of border. 0=no border
hBorder	tBorder	Side of border (4 bit: 1=Top 2=Bottom 4=Left 8=Right)
hAlignment	tAligment	Horizontal Alignment (L/C/R) (Left/Center/Right)
hAlignVert	tAlignVert	Vertical Alignment (T/C/B) (Top/Center/Bottom)
hTMargin	tTMargin	Top Margin of text into cell
hBMargin	tBMargin	Bottom Margin of text into cell
hLMargin	tLMargin	Left Margin of text into cell
hRMargin	tRMargin	Right Margin of text into cell
hCHeight	tCHeight	Cell Height
	tNumFormat	Number format for Numeric and Date Field for CLOB field this string indicate image formating parametrs: (ex: ’W=-1,H=20mm,A=C,V=C’) W=width of picture valid values are: -1=same proportion, 0=unchanged measure, Number+um new width with um H=height (same as W) A=Horizontal aligment (L,C,R) V=Vertical aligment (T,M,B)

The prefix **V** is relative to variable parameters

vSpacing es 12pt 50% 3mm	The spacing can be indicated in: pt (1 point = 1/72 inch), % of FontSize or millimeters. Interline are calculated as vSpacing + FontSize. If vInterline is specified vSpacing is ignored !
vInterline	Interline can be specified as vSpacing (pt/%/mm)
hSpacing	Header Spacing (see vSpacing)
hInterline	Header Interline (see vInterline)

The prefix **C** is relative to calculated parameters and are allways in pt.

cSpacing	Is equal to vSpacing if specified, or cInterline-FontSize.
cInterline	Is column property equal to vInterline if specified or cSpacing (default=0) + FontSize.
cTextArea	Is column property that indicate maximum widht available for text and is eqaul to colWidth-tLeftMargin-tRightMargin.
ctSpacing	Table Spacing (see cSpacing)
ctInterline	Table Interline (see cInterline)
ctTextArea	Table TextArea (see cTextArea)
chSpacing	Header Spacing (see cSpacing)
chInterline	Header Interline (see cInterline)
chTextArea	Header TextArea (see cTextArea)

During elaboration, this array are populated with single cell data
TYPE tp_Cell (Record with cell property) **tp_Cells** (Table of tp_Cell with alla alla row cells)

cX	Left side (value is relative to Start_X)
cY	Top side (value is relative to current Y posiztion)
cYbase	Te basis of the row to which the cell belongs
cTextHeight	Is a cell property that indicate overall text heght and is equal to cInterline*cRowsCount – cSpacing
cRowsCount	Number of Text rows
cTy	Distance between top border and the first text row
cWidth	Cell Width (pt)
cHeight	Cell Height (pt)
cRowText	Array with Text of each row
cRowTextX	Array with start point relative to left border, of each text row
cRowTextY	Array with start point relative to top border, of each text row
cRowTextWidth	Array with the width of each text row

The content of each cella is analyzed, if it's numeric or date type, it's formatted using tNumFormat, otherwise use wrapping for divide too long text rows

SET_MARGINS

It defines margins for data table, or for body of document
The parameters are in order Top, Left, Bottom, Right, units.
Default values are 3, 1, 4, 1, cm

SET_PAGE_PROC

It defines the code that will be executed at every page change.
This procedure accept as parameter a varchar2 or a clob, that contain the code that will be executed defined as an execute immediate.
Each time the procedure is called, the code will is appended to an array, so I can define multiple blocks of code that will be executed in the same order that they were defined.
In the code I can use 3 substitution variables
#PAGE_NR# that at runtime contains current page number,
#PAGE_COUNT#, that at runtime contains the total numerb of pages,
§, usefull when will to call procedures and functions that are defined into this package, this substitution variable will reduce the changes when you change the name of the pakage).
This procedure will be executde from (finish_pdf), which in turn is executed by save_pdf or get_pdf which are used respectively to save the PDF file or to obtain the CLOB with the content of the PDF document.
in other words just before saving, when we already know the total number of pages, it execute a loop for each page that invokes all page_proc the order in the same order that they were defined..
There are no distinction between header and footer, it depends on te coordinate used into procedure.
The Margins defined with set_margins haven't effect for this procedure.

QUERY2TABLE

This procedure use query result for print a datagrid

The parameters are: query, formats, colors, hRowHeight, tRowHeight, udm, StartX, BreakColumn)

`query`	is the query to execute
`formats`	is an array of columns formats
`colors`	is an array of Hex RGB colors, for header and data table with odd/even evidentiation (new in V 3.5.2: can be a varchar2 list of Hex RGB colors comma separately)
`hRowHeight`	is forced header Height, if null it's calculated, if 0 the header isn't drawn.
`tRowHeight`	is forced record height, if null it's calculated depend on contained data
`udm`	is the measure unit used for specify heights and coordinates
`StartX`	is l'offset tra il margine definito e il lato sinistro della tabella. (insentazione)
`BreakColumn`	Check columns from 1 to BreakColumn for detect breakings, 0 test is disabled.

QUERY2LABEL

This procedure use query result for print recod into array of labels, same as mailing list address.

The parameters are: query, formats, colors, hRowHeight, tRowHeight, udm, StartX, labeldef)

`query` is the query to execute

`formats` is an array of columns formats

`colors` is an array of Hex RGB colors, for header and data table with odd/even evidentiation (new in V 3.5.2: can be a varchar2 list of Hex RGB colors comma separately)

`hRowHeight` is forced header Height, if null it's calculated, if 0 the header isn't drawn.

`tRowHeight` is forced record height, if null it's calculated depend on contained data

`udm` is the measure unit used for specify heights and coordinates

`StartX` is l'offset tra il margine definito e il lato sinistro della tabella. (insentazione)

`labeldef` is type defined record that contail label definition, each mesure unit are in pt

`(`

- `MaxColumns` is Number of label columns in a page
- `MaxRows` is Number of label rows in a page
- `Width` is Width of each label
- `Height` is Height of each label
- `hDistance` is horizontal distance between right border and left border of two labels
- `vDistance` is vertical distance between bottom border and top border of two labels

`)`

How the package works:

1. Initialization
   ```
   as_pdf3_v5.init;
   ```
2. Defines sheets formats
   ```
   as_pdf3_v5.set_page_format('A4');
   ```
3. Defines page orientation
   ```
   as_pdf3_v5.set_page_orientation('P');
   ```
4. Defines margins
   ```
   as_pdf3_v5.set_margins(90, 10, 10, 10, 'mm');
   ```
   margins are in order Top, Left, Bottom, Right, measure unit.
   This margns are valid only for data table or body of document..
5. Defines Header and Footer procedure
   ```
   as_pdf3_v5.set_page_proc(varchar2 o clob)
   ```
   (see previously procedure explanation)
6. Defines columns formats (array or type `as_pdf3_v5.tp_columns`)
7. Defines SQL code of query
8. Execute query2table or query2label (see above for parameter definition)
9. I can repaet from 6 with other query
10. Execute save_pdf o get_pdf for saving the file or get blob content.