CALCTAB package
version 0.6.1

Roberto Giacomelli

e-mail: `giaconet dot mailbox at gmail dot com`

2009/07/12

> *"The table computes the sum*
> *not because is useful,*
> *but because the result*
> *is not an user essential data"*

Contents

1   Introduction

The CALCTAB package helps the user to typeset a kind of economic table such as invoices, expence notes and liquidation, or other tabular material with numeric columns.

A standard tabular environment works well but forces the author *to think* in a generic LaTeX language to reference economic object and without a conceptual relation among them. In addition a lot of formatting commands takes the same relevance of the main data, values, text description, etc.

Economic items maybe are independent values or not, like sums and percentages on previous values. The language must be expressive concerning together data and its relations. So, the priority is to add a new language element.

2   User manual

This section will be explain the use of the two CALCTAB environments calctab and xcalctab, first through several examples and then showing its formal syntax.

2.1   calctab and xcalctab by examples

2.1.1   A simple table

Well, starting with a simple case of a list of three entries, a 10% tax apply on it, and the final sum as showed in the next table:

	Items description	€
A	First cost	12 500,90
B	Second cost	8 560,03
C	Third cost	9 200,45
D	Tax (10,00% on A+B+C)	3 026,14
	Total cost (A+B+C+D)	**33 287,52**

The user can typesetting the example table with the followed CALCTAB code (obviously, the directive \usepackage{calctab} must be present in the document preamble).

```
% in the document preamble
\usepackage{calctab}
```

```
1  % in the document body
2  \begin{calctab}
3      \amount{First cost}{12500,9}
4      \amount{Second cost}{8560,03}
5      \amount{Third cost}{9200,45}
6      \perc{Tax}{10}
7      \add{Total cost}
8  \end{calctab}
```

Probably, is better to show the sum on which the 10% tax is applied. Look this code:

```
1  \begin{calctab}
2      \amount{First cost}{12500,9}
3      \amount{Second cost}{8560,03}
4      \amount{Third cost}{9200,45}
5      \add{Partial total}   % <-- The 'Partial total' row is not
6      \perc{Tax}{10}         %     considered in the 'Total cost' sum...
7      \add{Total cost}
8  \end{calctab}
```

Since the \add command not adds new table element in the stack so is easy to insert a subtotal. This is one of the basic rules implicit in the calctab environment so you are free to add an \add row without you worry about following row's computation. In other words, the \add command is a *trasparent* math object.

The behaviour of the inner commands is based on these main rules:

- Each command typeset one table row;

- The sequence of commands produce the sequence of the table rows;

- The \perc and the \add commands computes all previous values (with exception to the \add rows);

- Optional command argument provides a relation rows language to select an item or a group of items.

An independent item is not only a numeric data but also it's defined by a description text. This concept is directly express by the \amount command which needs two mandatory arguments: the *description text* and the item *numeric consistency*.

No more different is a percentage object structured in a description text and in a numeric value this time not an absolute number but a *factor* multiply by one hundred applied on some previous data, still we can easy express this with the \perc command.

Finally, we encountered a sum that only needs a description text. The CALCTAB language provides the \add command to cover this concept.

These three commands are available in an environment called calctab that represents the comprehensive structure of a table.

The code implementation of the inner commands computes the numeric results with the floating point algorithm rounded it with a max of two decimal digit (currently the results are truncated at this fixed precision). The number of the commands mandatory argument can be negative.

The CALCTAB environments manages all the table format elements such as the rows color, the rules, the columns header description text, etc. Furthermore, the code executes the data numeric formatting, the font family selection and more. For example, to the rows description text the \add and \perc commands adds a list of uppercase alphabetical character to explicate the computed rows.

Potentially all of these *visual property* could be adjust by the CALCTAB user.

2.1.2   Quantity and price columns

The CALCTAB package provides another environment called xcalctab where the x character suggests a multiply operation and an extension of the base calctab too. The inner commands are the same already discussed except the \amount command that accepts the description test and *two* numbers indeed only one.

The final table will appears with three columns typesetting with the two numeric arguments and their product. The following table with its corresponding code clarify this structure:

	Items description	Q.ty	Price	€
A	Newspapers	3,00	1,30	3,90
B	Coloured pencils	12,00	6,00	72,00
C	Maths exercise book	6,00	2,50	15,00
D	Compass	1,00	9,90	9,90
E	Ruler	1,00	5,90	5,90
F	Protractor	1,00	6,80	6,80
G	Deduction (−5,00% on A+B+···+F)			−5,68
	Total (A+B+···+G)			**107,82**

```
1  \begin{xcalctab}
2      \amount{Newspapers}{3}{1,30}
3      \amount{Coloured pencils}{12}{6,00}
4      \amount{Maths exercise book}{6}{2,50}
5      \amount{Compass}{1}{9,90}
6      \amount{Ruler}{1}{5,90}
7      \amount{Protractor}{1}{6,80}
8      \perc{Deduction}{-5}
9      \add{Total}
10 \end{xcalctab}
```

The help text $A+B+C+D+\cdots+\langle LAST\rangle$, will be shorted when the items are more than *five* elements, in $A+B+\cdots+\langle LAST\rangle$. The abbreviation condition is true only if the items in the subgroup are a consecutive sequence.

2.1.3 Items relations

If more complex tables require *relational* information between rows is necessary to assign an items names and to recall later these identifiers in a list. This simply way can be achieved with the command optional argument.

A name can be assign to a table row by using the `id` key (a significance name of the key improve code clarity). The syntax is based to $\langle key\rangle = \langle value\rangle$ assignment, where key is the fixed `id` string. Later on, compute sum or percentage only on a subset of previous calctab items, can get recall in a comma separated list, the user row names.

The next example explains the language support to the rows relation by naming with commands optional argument at work. For example, we want to apply a different discount on each expense or on each group of expense.

	Items description	€
A	Five dinner	146,00
B	Five breakfast	35,00
C	Discount ($-8,00\%$ on A)	$-11,68$
D	Discount ($-4,00\%$ on B)	$-1,40$
	Grand total (A+B+C+D)	**167,92**

```
1  \begin{calctab}
2      \amount[id=dinner]    {Five dinner}{146}
3      \amount[id=breakfast]{Five breakfast}{35}
4      \perc[dinner]    {Discount}{-8}
5      \perc[breakfast]{Discount}{-4}
6      \add{Grand total}
7  \end{calctab}
```

2.1.4 Subtotals

To the table of the last example we will add a subtotal as the sum of the discount rows. These \perc rows handling together own identifiers and a list of previous names (one id name list in this case). As the reader can be notice, the optional *table description* is typeset as a paragraph having the same (variable) table width.

Hotel bill:
Period from 12 october 2008 to 18 october 2008
(room 456)

	Items description	€
A	Five dinner (29,20 × 5,00)	146,00
B	Five breakfast (7,00 × 5,00)	35,00
C	Discount (−8,00% on A)	−11,68
D	Discount (−4,00% on B)	−1,40
	Total of discount (C+D)	**−13,08**
	Grand total (A+B+C+D)	**167,92**

```
1  \begin{calctab}[Hotel bill:\\
2      Period from 12 october 2008 to 18 october 2008 (room 456)]
3      \amount[id=dinner]     {Five dinner}   {29,2 * 5}
4      \amount[id=breakfast]  {Five breakfast}{   7 * 5}
5      \perc[id=dis8,dinner]  {Discount}      {      -8}
6      \perc[id=dis4,breakfast]{Discount}     {      -4}
7      \add[dis8,dis4]        {Total of discount}
8      \add                   {Grand total}
9  \end{calctab}
```

2.1.5   An arithmetic argument

The last example silently uses a feature of the \amount command (only for calctab and not for xcalctab) available from the version 0.6. The numeric argument can optionally be a *binary* math expression. The parse algorithm is very simple and the only syntax allowed are show in Tab. 1 where $\langle a \rangle$ and $\langle b \rangle$ are two decimal numbers.

Table 1: calctab \amount command optional binary operation

Floating point operation	Symbolic description	Numeric example
Addition	$\langle a \rangle + \langle b \rangle$	$15{,}20 + 41$
Addition	$-\langle a \rangle + \langle b \rangle$	$-15{,}20 + 41$
Subtraction	$\langle a \rangle - \langle b \rangle$	$15{,}20 - 41$
Subtraction	$-\langle a \rangle - \langle b \rangle$	$-15{,}20 - 41$
Multiplication	$\langle a \rangle \cdot \langle b \rangle$	$15{,}20 * 41$
Multiplication	$-\langle a \rangle \cdot -\langle b \rangle$	$-15{,}20 * -41$
Division	$\langle a \rangle \; / \; \langle b \rangle$	$15{,}20/41$
Division	$-\langle a \rangle \; / \; \langle b \rangle$	$-15{,}20/41$

Sometimes the applied round off number must be show in the table. This is possible because calctab reports the \amount binary operations. Here is an example.

	Items description	€
A	A rounded cost (4 506,00 − 6,00)	4 500,00
B	Positive rounded cost (3 598,30 + 1,70)	3 600,00
	Total (rounded) (A+B)	**8 100,00**

```
1  \begin{calctab}
2      \amount{A rounded cost}{4506-6}
3      \amount{Positive rounded cost}{3598,30+1,70}
4      \add{Total (rounded)}
5  \end{calctab}
```

2.1.6 Sum of sums

The next table a 20% deduction amount must be apply on the total of the first three values (see the F row). A solution is to assign a name for each rows and to write its list in the optional argument of \perc command. Nevertheless, this sum is just computed in previus row, thus we can assign a name only to row \add[id=total]{Total amount of the design task}, and later recall the result and then apply percentage on it with the line \perc[total]{Deduction}{-20}.

Plant project Fee Note:
Italian restaurant and pizzeria "La Margherita"

	Items description	€
A	Design of the air conditioned plant	5 400,00
B	Design of the electric plant and video surveillance	8 000,00
C	General expence (8,55% on A+B)	1 145,70
	Total amount of the design task (A+B+C)	**14 545,70**
D	Tax (2,00% on A+B+C)	290,91
E	VAT (20,00% on A+B+C+D)	2 967,32
F	Deduction ($-20,00\%$ on A+B+C)	$-2\,909{,}14$
	Total (A+B+$\cdots$+F)	**14 894,79**

The complete code is showed here:

```
1  \begin{calctab}[Plant project Fee Note:\\
2          Italian restaurant and pizzeria ''La Margherita'']
3      \amount{Design of the air conditioned plant}{5400}
4      \amount{Design of the electric plant and video surveillance}{8000}
5      \perc{General expence}{8,55}
6      \add[id=total]{Total amount of design task}
7      \perc{Tax}{2}
8      \perc{VAT}{20}
9      \perc[total]{Deduction}{-20}
10     \add{Total}
11 \end{calctab}
```

2.1.7 Interval selection

The *interval* selection of a elements list is simply an identification of consecutive elements. So we only need to specify the first and the last elements delimiters of the series. If the first delimeter is not an explicit information, we can give to the first list element this role. Equally, the default last element of the interval can be the last list element, if one is not available.

The CALCTAB commands \perc and \add supports the interval selection with from and until keys that referenced row name. For example in this follow table the first five items and the last four are two separated groups.

Is possible to combine the key until with the key from in the same optional argument to select a completely defined inner interval. In case of a selection overlap, a package warning will be throw at compile time.

The interval selection table

	Items description	€
A	One	100 000 000,00
B	Two	20 000 000,00
C	Three	3 000 000,00
D	Four	400 000,00
E	Five	50 000,00
F	Six	6 000,00
G	Seven	700,00
H	Eight	80,00
I	Nine	9,00
	From one to five (A+B+C+D+E)	**123 450 000,00**
	From six to nine (F+G+H+I)	**6 789,00**
	From five to six (E+F)	**56 000,00**
	As well as total (A+B+ ··· +I)	**123 456 789,00**

```
1  \begin{calctab}[The interval selection table]
2      \amount           {One}{100000000}
3      \amount            {Two}{20000000}
4      \amount          {Three}{3000000}
5      \amount           {Four}{400000}
6      \amount[id=five]{Five}{50000}
7      \inrule
8      \amount[id=six]  {Six}{6000}
9      \amount          {Seven}{700}
10     \amount          {Eight}{80}
11     \amount           {Nine}{9}
12     \add[until=five]{From one to five}
13     \add[from=six]{From six to nine}
14     \add[from=five,until=six]{From five to six} % same of [five,six]
15     \add{As well as total}
16 \end{calctab}
```

An inner rule is adding *manually* to the table with command \inrule (that just sounds *inner rule*).

2.2   Environments formal syntax

2.2.1   calctab syntax

The calctab environment syntax will be described in succession:

```
\begin{calctab}[⟨description paragraph⟩]
    ⟨calctab commands list⟩
\end{calctab}
```

The ⟨*calctab commands list*⟩ is a sequence of the internal commands \amount, \perc, \add and \inrule.

```
\amount[⟨key-value identifier⟩]{⟨description⟩}{⟨value⟩}
```

⟨*key-value identifier*⟩ = id=⟨*rowID*⟩;

⟨*description*⟩ = ⟨*text line*⟩;

⟨*value*⟩ = ⟨*decimal value*⟩ or ⟨*basic binary operation*⟩.

```
\perc[⟨key-value list⟩]{⟨description⟩}{⟨percentage⟩}
```

    ⟨*key-value list*⟩ = `id=`⟨*rowID*⟩, `from=`⟨*rowID*⟩, `until=`⟨*rowID*⟩, ⟨*rowID list*⟩;

    ⟨*description*⟩ = ⟨*text line*⟩;

    ⟨*value*⟩ = ⟨*percentage decimal value*⟩.

```
\add[⟨key-value list⟩]{⟨description⟩}
```

    ⟨*key-value list*⟩ = `id=`⟨*rowID*⟩, `from=`⟨*rowID*⟩, `until=`⟨*rowID*⟩, ⟨*rowID list*⟩;

    ⟨*description*⟩ = ⟨*text line*⟩.

```
\inrule
```

2.2.2   xcalctab syntax

The xcalctab environment syntax will be described in the next paragraph:

```
\begin{xcalctab}[⟨description paragraph⟩]
    ⟨xcalctab commands list⟩
\end{xcalctab}
```

    The ⟨*xcalctab commands list*⟩ is a sequence of the internal commands `\amount`, `\perc`, `\add` and `\inrule`.

```
\amount[⟨key-value identifier⟩]{⟨description⟩}{⟨value1⟩}{⟨value2⟩}
```

    ⟨*key-value identifier*⟩ = `id=`⟨*rowID*⟩;

    ⟨*description*⟩ = ⟨*text line*⟩;

    ⟨*value1*⟩ = ⟨*decimal value*⟩;

    ⟨*value2*⟩ = ⟨*decimal value*⟩.

```
\perc[⟨key-value list⟩]{⟨description⟩}{⟨percentage⟩}
```

    ⟨*key-value list*⟩ = `id=`⟨*rowID*⟩, `from=`⟨*rowID*⟩, `until=`⟨*rowID*⟩, ⟨*rowID list*⟩;

    ⟨*description*⟩ = ⟨*text line*⟩;

    ⟨*value*⟩ = ⟨*percentage decimal value*⟩.

```
\add[⟨key-value list⟩]{⟨description⟩}
```

    ⟨*key-value list*⟩ = `id=`⟨*rowID*⟩, `from=`⟨*rowID*⟩, `until=`⟨*rowID*⟩, ⟨*rowID list*⟩;

    ⟨*description*⟩ = ⟨*text line*⟩.

```
\inrule
```

    Please note that the calctab and xcalctab environments are based on the standard LATEX tabular environment, so you can manage your table as a float object all-surrounding it with a table environment.

2.3   The table appearance commands (at the moment)

The first row of the table typeset by calctab environment contains a pair of texts. The title of description column, that can be modify by the user with the command \ctdescription, and the title of economic column that describe currency, set by the command \ctcurrency. Default values are "Items description" and \euro.

These command save the argument in defined internal macro, in the same way of the common \title and \author command of the article class. Insert these commands before the calctab environment to change it.

$$\boxed{\texttt{\textbackslash ctcurrency\{}\langle \textit{user text}\rangle\texttt{\}}}$$

$$\boxed{\texttt{\textbackslash ctdescription\{}\langle \textit{user text}\rangle\texttt{\}}}$$

A table example show explicit assignment of the description text and currency unit after this commands:

```
1  \ctdescription{Your bill, thanks a lot.}
2  \ctcurrency{USD}
```

	Your bill, thanks a lot.	USD
A	Five dinner	146,00
B	Five breakfast	35,00
C	Discount ($-8{,}00\%$ on A)	$-11{,}68$
D	Discount ($-4{,}00\%$ on B)	$-1{,}40$
	Grand total (A+B+C+D)	**167,92**

The default english text in percentage row can be changed in the same way with the command \ctontranslation.

$$\boxed{\texttt{\textbackslash ctontranslation\{}\langle \textit{on traslation}\rangle\texttt{\}}}$$

and the headers of the two xcalctab numeric columns can be setting with

$$\boxed{\texttt{\textbackslash ctheaderone\{}\langle \textit{first text header}\rangle\texttt{\}}} \quad \boxed{\texttt{\textbackslash ctheadertwo\{}\langle \textit{second text header}\rangle\texttt{\}}}$$

2.4   Incrementing the table width

The width of the CALCTAB table can be adjust via a standard LaTeX length called \ctsep (default value is 8mm). The horizontal space will be add to the central column of the table. The next line is an example:

```
\setlength{\ctsep}{28pt}
```

2.5   New command with CALCTAB environments

Some cases requires repetitive tasks with tables. How CALCTAB can help to do these?

2.5.1   One command, one table

A table similar to calctab basic structure can to be build with a unique command.

```
1  \newcommand{\billtab}[2][10]{% two arg command
2  \begin{calctab}[This is my table]
3  \amount{Confirmed cost}{#2}
4  \perc{Tax}{#1}
5  \add{Total}
6  \end{calctab}}
```

When you write: `\billtab{2800}` you achieved this table:

This is my table

	Description	€
A	Confirmed cost	2 800,00
B	Tax (10,00% on A)	280,00
	Total (A+B)	**3 080,00**

and with the command `\billtab[20]{2000}` you achieved:

This is my table

	Description	€
A	Confirmed cost	2 000,00
B	Tax (20,00% on A)	400,00
	Total (A+B)	**2 400,00**

2.5.2 Inner table command

Sometimes is useful to insert only one command to insert several rows of CALCTAB computation. The next code is an example.

```
1 \newcommand{\closetable}[1][10]{
2 \add{Subtotal}
3 \perc{Tax}{#1}
4 \add{Total}
5 }
```

With this command the user can write shortly:

```
1 \begin{calctab}
2     \amount{Cost}{1200}
3     \amount{Save pricing}{-400}
4     \closetable
5 \end{calctab}
```

	Description	€
A	Cost	1 200,00
B	Save pricing	−400,00
	Subtotal (A+B)	**800,00**
C	Tax (10,00% on A+B)	80,00
	Total (A+B+C)	**880,00**

3 CALCTAB idea

The language efficacy is particularly important for the LATEX user. The CALCTAB underline concept is to build an expressive language for the model – view paradigm.

So one *model* of structured data, can have more and more presentation *view*. For example, a text with tabular structure can became a report or a graphic plot image.

4   ToDo

A big work remain to do until the release 1.0 version: a language design and implementation too of a new user interface that make easy modify the view paramenter of the table such as the type of label row (number, letter and so on), or to define the description text in the header column.

This list report tasks:

- write in good english the package documentation;

- internationalization of the fixed text in table and the currency unit;

- data input and inner commands language development;

- new infrastructure to manage the table view properties;

- . . .

5   Package License

This work may be distributed and/or modified under the conditions of the LaTeX Project Public License, either version 1.3 of this license or any later version. The latest version of this license is in http://www.latex-project.org/lppl.txt and version 1.3 or later is part of all distributions of LaTeX version 2005/12/01 or later.

6   Thanks and acknowledgments

The enthusiasm raised during GuIT*meeting*2008 in Pisa (Italy)[1] by the talk *"Una tabella che fa calcoli"* and the tip suggest me by Norbert Preining (Vienna University of Technology) to upload on CTAN the code in experimental section, finally persuade me to publish this work.

Thanks GuIT*meeting*2008 people, thanks Norbert.

Thanks a lot to Professor Enrico Gregorio for help me to resolve some code problem: balance the table description paragraph width to the table variable width, save in a tokens registry the material of table so as over the implicit group barrier of the tabular cells, and solve consequent `\expandafter` *jungle*.

Thanks a lot to the authors of the packages used by calctab.

And finally, thanks to my family.

Every comment are welcome as well as any language discussion, so don't esitate to send to me an e-mail message with your opinion, suggestion or desiderata, or to leave a public comment in my site blog[2]. Thank you.

---

[1] For information please visit the site www.guit.sssup.it/guitmeeting/2008/2008.it.php
[2] Blog site address: http://robitex.wordpress.com/calctab-package/

7   Change history

Ver.	N.	Date	Improvement description
0.5		2009/02/23	first CALCTAB CTAN release
	1	2009/03/03	New \inrule command to manual draw a rule among the table rows
	2	2009/03/03	Fixed bug: the absense of a rule after an add row
	3	2009/03/05	New xcalctab environment (extended environment with the "quantity" and "price" columns)
	4	2009/03/10	New feature: automatic short of the description string (A+B+C+···) if it's long than more five items
	5	2009/03/10	Substituted the standard LaTeX command \Alph with \AlphAlph by Heiko Oberdiek to eliminate the 26 rows limit in label column
	6	2009/03/12	New feature for \amount command: basic binary operation with two numbers implemented with the xstring package
	7	2009/03/12	Some minor internal code adjustments
0.6		2009/03/29	CTAN upload of the new version
	1	2009/06/26	Fixed parsing of negative numbers in the \amount command binary operations
	2	2009/07/12	User manual improvement
0.6.1		2009/07/14	CTAN upload of the maintenance release