What's New in Version 7.0 of TPL

TPL Version 7.0 is a major upgrade to TPL. It contains several new features and enhancements to existing feature. For additional details on any of these features, see the Reference manual or Help files.

Major New Features in TPL 7.0

PERCENTS

Several new options have been added to the TPL Tables language to make it easier to specify tables with percents. Also, Table Builder can now produce table requests with percents interactively. With the new features many tables with percents can be specified without using percent markers. Tables with both numbers and percents are also easier to create.

If your desired table consists of percent cells or alternating percents and number cells, a percent variable can be placed in the title line of a table rather than in the individual dimensions of the table.

Examples This request produces the table which follows:

```
Percent P1 "Percent" mask 999 %;

    Table one "Mean Income for New England States by Education"  Percent p1:
        Stub Mean_income by New_England;
        Head Education;
```

Mean Income for New England States by Education

	Education						
	Total	8 years or less	High school, 1 to 3 years	High school, 4 years	College, 1 to 3 years	College, 4 years	College, 5 or more years
New England Total							
Percent	100.0%	53.0%	67.3%	91.4%	108.3%	141.6%	173.8%
Connecticut							
Percent	100.0	51.0	78.4	88.9	97.0	147.2	158.9
Maine							
Percent	100.0	53.3	74.4	98.3	113.3	126.8	172.2
Massachusetts							
Percent	100.0	47.0	59.3	87.3	101.8	132.8	166.0
New Hampshire							
Percent	100.0	54.5	63.0	103.2	106.4	135.6	130.1
Rhode Island							
Percent	100.0	56.3	71.5	102.1	105.1	132.2	145.1
Vermont							
Percent	100.0	44.3	69.2	97.7	76.4	149.6	140.1

If we wish to move the base markers to the first row we need only change the percent statement by adding a base term:

Percent P1 "Percent" mask 999 % **base first row**;

Mean Income for New England States by Education

	Education						
	Total	8 years or less	High school, 1 to 3 years	High school, 4 years	College, 1 to 3 years	College, 4 years	College, 5 or more years
New England Total							
Percent	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
Connecticut							
Percent	123.5	118.9	143.8	120.0	110.5	128.4	112.8
Maine							
Percent	81.2	81.7	89.7	87.3	85.0	72.8	80.5
Massachusetts							
Percent	120.1	106.5	105.7	114.6	112.9	112.6	114.7
New Hampshire							
Percent	106.7	109.9	99.9	120.4	104.9	102.2	79.9
Rhode Island							
Percent	107.5	114.2	114.1	120.0	104.3	100.4	89.7
Vermont							
Percent	101.9	85.3	104.7	108.8	71.9	107.6	82.1

If we want both percents and values in our table we can add percent conditions to our percent statement:

```
Percent P1 "" mask 999 %;
        "Value" mask $ 99,999 : value;
        "Percent" mask 99.99 % : percent;
```

Mean Income for New England States by Education

	Education						
	Total	8 years or less	High school, 1 to 3 years	High school, 4 years	College, 1 to 3 years	College, 4 years	College, 5 or more years
New England Total							
Value	$31,689	$16,787	$21,338	$28,978	$34,320	$44,859	$55,087
Percent	100.00%	52.97%	67.33%	91.45%	108.30%	141.56%	173.84%
Connecticut							
Value	$39,122	$19,968	$30,682	$34,772	$37,932	$57,594	$62,160
Percent	100.00%	51.04%	78.43%	88.88%	96.96%	147.22%	158.89%
Maine							
Value	$25,747	$13,720	$19,143	$25,310	$29,166	$32,642	$44,333
Percent	100.00%	53.29%	74.35%	98.30%	113.28%	126.78%	172.19%
Massachusetts							
Value	$38,052	$17,875	$22,564	$33,214	$38,750	$50,519	$63,185
Percent	100.00%	46.98%	59.30%	87.29%	101.83%	132.76%	166.05%
New Hampshire							
Value	$33,822	$18,450	$21,308	$34,890	$35,998	$45,847	$44,005
Percent	100.00%	54.55%	63.00%	103.16%	106.43%	135.55%	130.11%
Rhode Island							
Value	$34,068	$19,165	$24,352	$34,775	$35,794	$45,030	$49,426
Percent	100.00%	56.25%	71.48%	102.07%	105.07%	132.17%	145.08%
Vermont							
Value	$32,279	$14,314	$22,332	$31,533	$24,667	$48,279	$45,216
Percent	100.00%	44.35%	69.18%	97.69%	76.42%	149.57%	140.08%

DATA DRILLING (Windows Only)

A Data Drilling feature has been added to Ted. Select table cells of interest and use the data drill option. You will then be asked to select which fields you wish to display. TPL will create a report of the individual records which contributed to the selected cells showing the selected fields. The feature is useful for finding errors in your data file or in finding individual records in your data which are of special interest.

STATISTICAL TESTING (Windows Only)

Statistics Testing has been added to TPL. TPL can perform T-Tests and several other statistical tests. The results are displayed in the table in a publishable format. An additional report is created with more details about the tests. (Windows only). Currently supported tests are:

- Student's T-Test
- Z Test
- Anova F-Test
- F-Test of Standard Deviations
- Chi Squared Test
- Tukey HSD Test
- Mann-Whitney U-Test

The following are the results of a T-test and Anova Test.

TABLE 1

	Count	Mean Value
subject		
01	2	11.70
02	2	7.90
03	2	17.20
04	2	[1]26.10
05	2	[2]25.65
06	2	26.15
07	2	17.10
08	2	[1]21.35
09	2	22.25
10	2	19.00
11	2	[2]24.20
12	2	20.30
13	2	27.45
14	2	17.40
15	2	16.80

[1] A 1 tailed T-test on the means showed the difference to be significant at the 5% level.
[2] A 2 tailed T-test on the means showed the difference to not be significant at the 10% level.

TABLE 1

	case				Total
	0	1	2	3	
Count	5.00	5.00	5.00	5.00	20.00
dosage	144.30	125.20	112.50	111.50	493.50
Mean Dosage	28.86	25.04	22.50	22.30	24.68

A 1 tailed ANOVA F-test performed on CASE in the shaded area showed the difference to be significant at the 5% level.

SPECIAL CHARACTERS

In previous versions of TPL, special characters such as long dashes and accented characters were entered into a TPL request by use of a backslash followed by a 3 digit numeric code. The code was determined by the character mapping (codeset) used for the table. In addition to making the request hard to decipher, the requests were not portable between Windows and Unix.

TPL now provides an alternate way of entering special characters -- the character name preceded by an ampersand (**&**) and followed by a semicolor (**;**). For example, a long dash can be entered as **&emdash;** and an **E** with an acute accent can be entered as **É**. This alternate way of entering special characters is especially useful for entering euros in a table request. The problem with euros is that the appropriate numeric code depends on the output context. If you are printing a table with a euro you may need a different character code than if you are turning the table into a pdf. If you use of **€**, TPL will automatically select the right code for the output context.

PERCENT CHANGE

The **Percent Change** and **Numeric Change** statements create new observation variables which can be used in table statements. Specification of the statements requires 2 On variables, an observation variable and a control variable. The changes that appear in the table are the change in the On Observation value that occur when you move from one condition of the On Control variable to the next.

Example

In this example we look at the number of cases of certain medical conditions at a hospital and the percent change in counts from month to month. Note that January does not have percent change values since there is no previous month.

```
Define WHY on PROBLEM;
        "Not yet Classified"            if "000";
        "Brain & Nervous System"        if "001": "035";
        "Respiratory System"            if "076": "102";
        "Heart & Circulatory System"    if "103": "145";

Percent Change PCH "Monthly Change" Mask 99.99% center
on count and Month;

Table TABLE_3:
        Stub YEAR by MONTH;
        Head WHY by (COUNT then PCH);
```

TABLE 3

	Not yet Classified		Brain & Nervous System		Respiratory System		Heart & Circulatory System	
	Count	Monthly Change	Count	Monthly Change	Count	Monthly Change	Count	Monthly Change
1985								
JANURARY	1	–	39	–	66	–	90	–
FEBRUARY	3	200.00%	28	-28.21%	89	34.85%	87	-3.33%
MARCH	4	33.33	31	10.71	65	-26.97	79	-9.20
APRIL	5	25.00	32	3.23	49	-24.62	69	-12.66
MAY	8	60.00	29	-9.38	30	-38.78	64	-7.25
JUNE	2	-75.00	21	-27.59	33	10.00	68	6.25
JULY	–	–	25	19.05	22	-33.33	59	-13.24
AUGUST	2	–	18	-28.00	16	-27.27	63	6.78
SEPTEMBER	2	0.00	21	16.67	29	81.25	63	0.00
OCTOBER	4	100.00	17	-19.05	27	-6.90	58	-7.94
NOVEMBER	4	0.00	18	5.88	34	25.93	64	10.34
DECEMBER	21	425.00	23	27.78	39	14.71	60	-6.25

– Data not available.

DEFINE ON MULTILE VARIABLES

Defines can be done on multiple variables using language similar to a conditional compute.

Example

In this example, people have been asked to check one or more of several reasons why they like their neighborhood. We would like to know how many people checked each reason and also have a total of the number of respondents in the survey. The following DEFINE statement will provide a single new variable with all of the categories we want. All people will be counted in the first (Total) category and also in one or more of the other categories.

```
DEFINE REASONS;
        'Total'         IF ALL;
        'Arts/Culture'  IF ARTS = 1;
        'Parks'         IF PARKS = 1;
        'Shopping'      IF SHOPS = 1;
        'Housing'       IF HOUSES = 1;

TABLE D2 'What do you like about your neighborhood?':
        HEADING REASONS;
        STUB TOTAL THEN GENDER;
```

What do you like about your neighborhood?

	Total	Arts/Culture	Parks	Shopping	Housing
Total	592	220	378	193	200
Female	362	127	231	104	119
Male	230	93	147	89	81

RANK ON VALUE

A new format statement, RANK ON VALUE, has been added. Rather than rearranging rows as with the Rank statement in the table request, this statement just replaces the value in a cell with its rank number. This feature enables you to create a single table containing separate rankings on multiple variables.

Table Request

```
Table One "":
        Stub COUNTRY;
        Head EXPORTS then EXPORTS then IMPORTS then IMPORTS;
```

Format Request

```
For Table One Columns 2,4: Rank on values;
For Table One Columns 2,4: Replace mask with Right 99;
For Table One Columns 2 Variable EXPORTS: Replace label with
  "Exports Ranking";
For Table One Columns 4 Variable IMPORTS: Replace label with
  "Imports Ranking";
```

	Exports in Millions	Exports Ranking	Imports in Millions	Imports Ranking
Country				
Canada	$461,800	10	$386,900	9
China	1,465,000	2	904,600	3
France	629,700	5	600,900	5
Germany	1,530,000	1	1,075,000	2
Italy	566,100	6	498,600	7
Japan	776,800	4	573,300	6
Netherlands	537,500	7	404,700	8
Russia	476,000	8	260,400	10
United Kingdom	468,700	9	621,400	4
United States	1,377,000	3	1,968,000	1

Additional New Features and Conveniences

Interactive Environment

- Ted now has an unlimited level UNDO option. This is especially useful for balancing banked tables and removing the changes made by statistical tests.
- In Table Builder, when windows are expanded, list of variables and conditions are also expanded so more items can be seen.
- TPL, Ted, Codebook Builder, and Table Builder have the option to display the combined Help rather than just the Help for the particular program.
- In Table Builder, a user can now reorder the terms in a quantile statement.
- For database jobs, if a password is not required, there is no prompt.
- The instructions which come up at the start of TPL, TED, Codebook Builder, and Table Builder now have buttons to do the most common actions.
- Table requests can be run from within Table Builder.
- In Codebook Builder, binary data types can be specified as signed or unsigned.
- In Table Builder, Ted, and Codebook Builder, holding the right mouse button down over a variable or symbol will display the full name of the variable or symbol in a box next to the item.

New Language Features

- Some table request may specify that column 3 should be the difference between columns 1 and 2. Because of rounding errors, the displayed results may not match perfectly. The new post compute function DISPLAY specifies that a term in a post compute be rounded to its displayed value before a calculation is performed. This prevents the problem. DISPLAY is actually an undocumented feature in V6 TPL rather than a new feature in V7.
- SUP and SUB now affect the vertical alignment of footnote symbols.
- Post Computes no longer require a variable. You can now write POST COMPUTE CONSTANT = 3; Before, to get a constant value in a request, you would need something like POST COMPUTE CONSTANT = 3 + 0 * COUNT;
- A new format statement has been added to specify the properties of a bank divider.
- Page markers now retain their original size when a table is scaled.
- Alternate Quantile and Median statements have been added to TPL. These statements use a fixed interval range rather than the logarithmic range used by the standard Quantile and Median statements.
- An option is available to suppress the often long list of printed rows in the output file.
- STDEV, STDERR, and VAR can now use weighting variables. In earlier versions of TPL, only MEAN supported weights.

- An empty ("") field in a CSV file is now treated as a Null value.
- Footnote text and other properties of footnotes can now be specified at the table level. Before they could only be specified for an entire request. If FOOTNOTES EACH WAFER is specified, footnote text and properties can be specified for individual wafers.

Export

- HTML Links and Anchors can now be specified for masks and labels.
- Ted supports specifying the divide character in CSV export. There is also a script option in Windows and Unix for this.
- Line skips are retained when tables are exported to HTML.

Scripts

- A codebook name can be passed to TPL either as a command line argument or in a script. This, combined with the use of MERGE in a datalist, allows a table to be created from multiple files with the same field names but different formats.
- Users can now write scripts which pass multiple tables to change into eps files. In earlier versions of TPL, the eps files would overlay each other because they had common names.
- Scripts options have been added to support export of tables to pdf format.

Bug Fixes

Note: When bugs are found, they are fixed and new versions of the system are made available. So many of the bugs listed below have been fixed in the Version 6 TPL system that you are using.

WTPL

- CALL in scripts may not work.
- When multiple ejects are specified for some filled rows, they may appear in later, unspecified rows.
- If rank display variables are in a different dimension from an observation variable, an inappropriate message about nested observations may result.
- For hierarchical and database files, if the cell buffer is emptied multiple times, data may be sent to the wrong cell.
- If a table is exported as a CSVfile and the last column of the table is deleted, output records may be run together since no new line character is generated.
- For a multi-line page marker, the space between lines may be incorrect.
- In a database codebook, if a character field is broken into parts by using SUBSTR, the resulting codebook may be incorrect.
- If one table in a request has a table select and a later table does not, the later table may have data deleted by the earlier select.
- If TABULATE INCOMPLETE HIERARCHIES and select or data errors make the last hierarchical unit incomplete, then the last record may be included in the tabulation even though it shouldn't be.
- If a table run is cancelled, the run button may be left inactive.
- CSV export cells are affected by column width even though they should not be.
- TPL 6.0 supports only one table level select per table.
- In line printer mode, column banking results in the same data being displayed for all banks.
- Footnote symbols renumber for each page when a start page number is specified.
- TPL 6.0 does not correctly handle some select statements which contain multiple NOT (v IN (a,b,c)) type phrases.
- Built-in statistics functions did not handle NULLs correctly.
- Having too may inserts (color, font, etc) at a single point in a label may cause system to crash.
- In a few jobs with COMPRESS HEADINGS, heading is not displayed correctly.

- HTML tables with the letter **u** with an acute accent may incorrectly extend to a new page.
- In psp, the help message listing font abbreviations is incorrect.

TABLE BUILDER

- Trying to create a quantile without specifying a rank variable correctly produces an error message. When a user corrects the error, Table Builder crashes.
- Table Builder crashes when no option is explicitly selected in the Statistics Window.
- When Defines with condition names but no labels are edited, the "automatic" labels are discarded leaving no label.

CODEBOOK BUILDER

- If an old ASCII codebook is being edited and Codebook Builder is quit while a field is being edited, Codebook Builder will crash.
- If you close Codebook Builder after having created a codebook but before processing the codebook, Codebook Builder may crash.
- For database codebooks, check boxes for including and excluding database tables may be ignored.
- For old CSV codebooks, masks on variables may be discarded.
- For new CSV codebooks, if the quote symbol is not ", the resulting codebook may be incorrect.
- Updating of a CSV codebook generated from a data list may not work correctly.
- In a new CSV codebook, specification of default labels may result in a crash.

TED

- If a control variable condition is selected, and then user switches to (control) variable, the mask option is incorrectly still active. Clicking on it may result in a crash.
- HTML export in scripts may not work correctly.
- When a down rule is selected and Hidden is clicked, the wrong menu is displayed.
- If you click on the row bank divider and make a change, an incorrect format statement may be generated.
- Printing of multiple files in a script may fail.
- Printing of a non-postscript table may be incorrect.
- BMP export of a rotated table is incorrect.
- If while creating a new footnote, you change windows to change the footnote symbol, the information from the first window may not be saved.

- Footnote changes such as changing a label or symbol will not produce the "Do you want to save before closing?" message.
- In the expand cell scope window, if you switch from ALL to Filled rows, an incorrect row range may be displayed.
- If the location of the Distiller program for producing pdf's has not been specified, you may browse to find it. Pressing the browse button may case Ted to crash.
- Specifying a shade color for Total will have no effect.
- Euros may not display, print, or export to PDF correctly because the same code does not work for all of them. Version 7 allows independent specification for the Euro symbol for each of these contexts.
- Postscript print of an ASCII file may leave the printer in a unfinished state since no end of file symbol is sent.
- If footnote justification is specified, the first line of the footnotes may not be displayed.
- If the font size for text is changed, the scroll bars may be the wrong length.
- If you edit an existing footnote, the old properties associated with the footnote symbol are not filled in.
- In some cases, the size and location of displayed files was not retained when the displayed file was maximized at the time it was closed.

UNIX/LINUX

- If the person installing TPL does not specify an editor or table viewer, the profile.tpl file may be incorrect and jobs will not run.
- In the 64 bit Sun version, label indents may not work nor will specifying number of digits in text masks.
- In Linux TPL Report, if a directory is reused, old reports may not be overlaid.