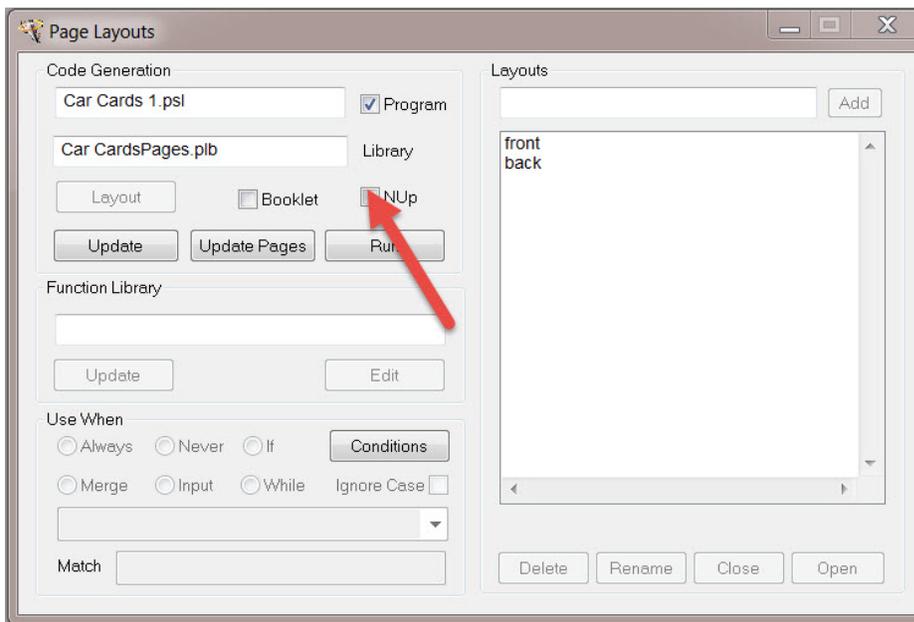


## Automatic Composition

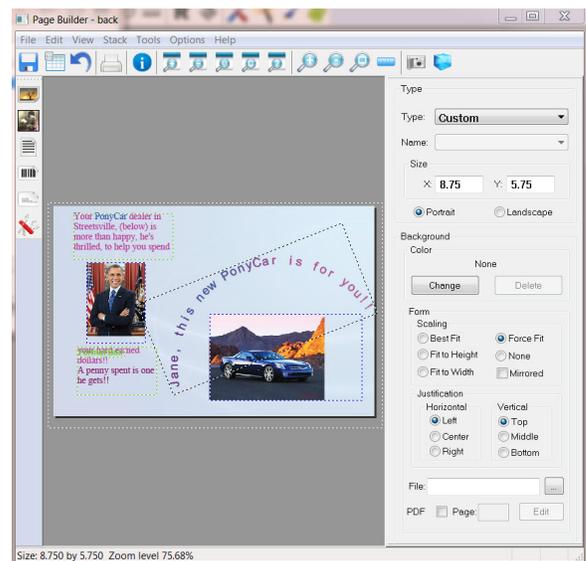
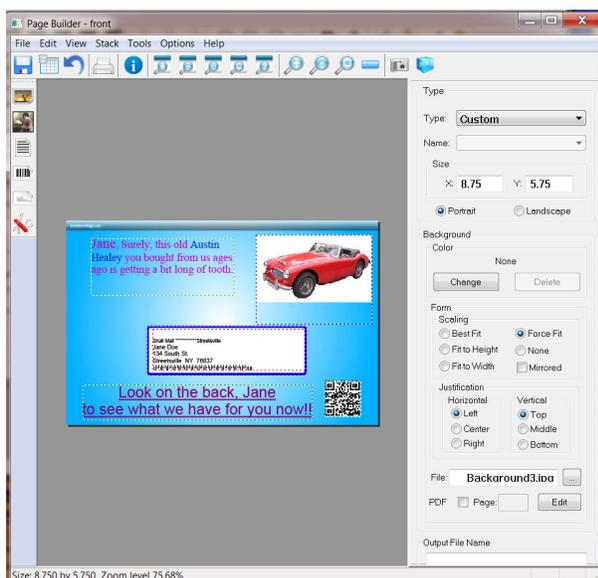
Workflow automation seems to be the buzzword in our industry these days -- and it should be it saves time and reduces errors. It makes direct mail firms more competitive.

However there is another form of automation often overlooked -- Composition Automation. Automation in this area also delivers two significant benefits; 1) It saves time -- in a major way, and, 2) Since such automation is often data dependent, it delivers a very serious competitive edge. That, of course requires explanation and that is what this document is about. The first example:

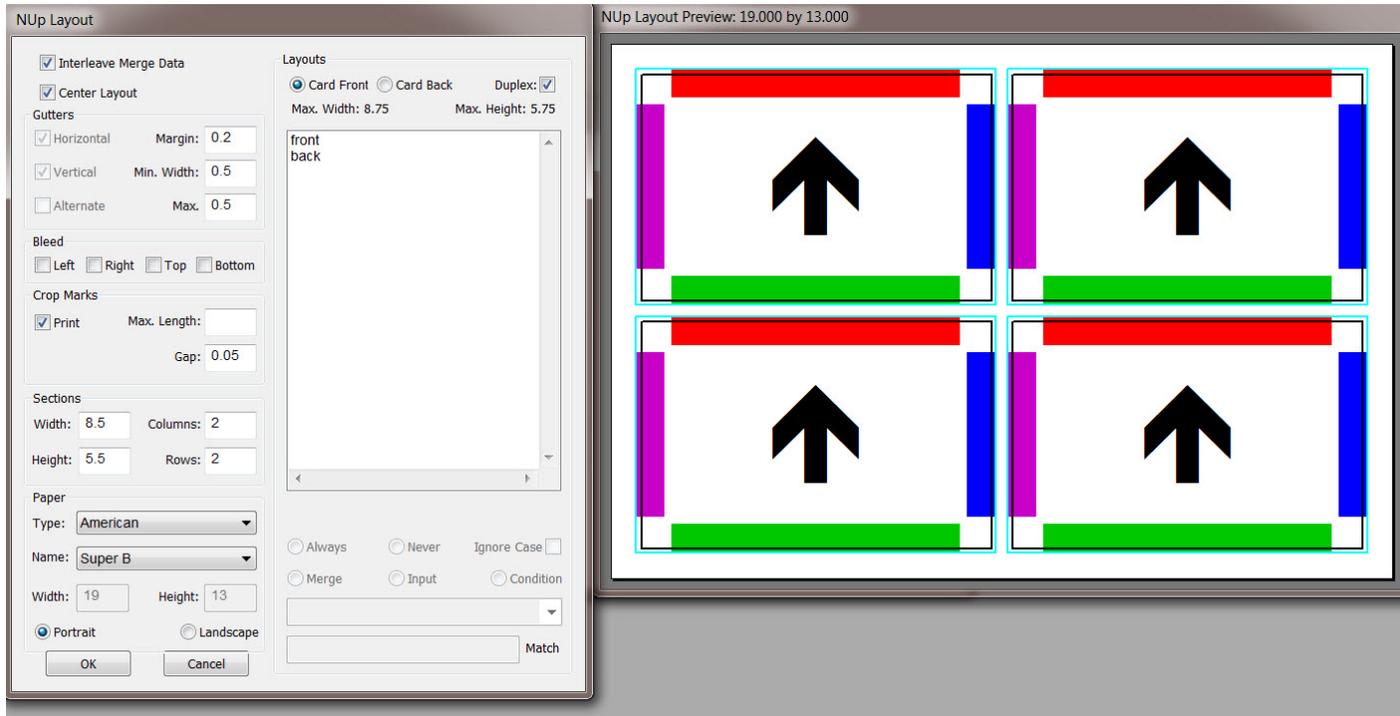
**N-Up** – Just compose front and back sides (if any) of the postcards, choose stock size, click a few radio buttons (such as “Duplex” and “Interleave” and the rest is done by PSL for you. **And it won't print if you make a mistake** – clicks/paper cost money. Just select N-Up as shown below;



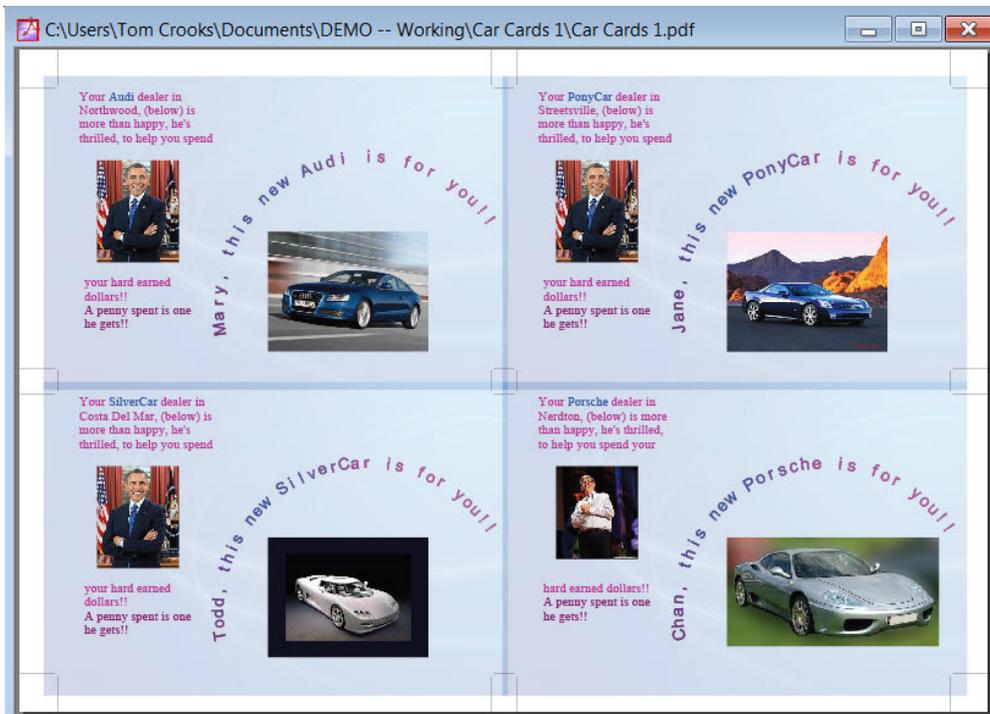
Then compose one or more fronts and backs.



Then go to "Layout" and find that PSL has placed as many cards as it could on the default stock size of 13x19.



A user can, of course change stock size, set batch size, margins, gutters, etc. and PSL automatically produces a PDF ready for printing in "cut and stack" order. Note that the default layout is left-right, north-south.

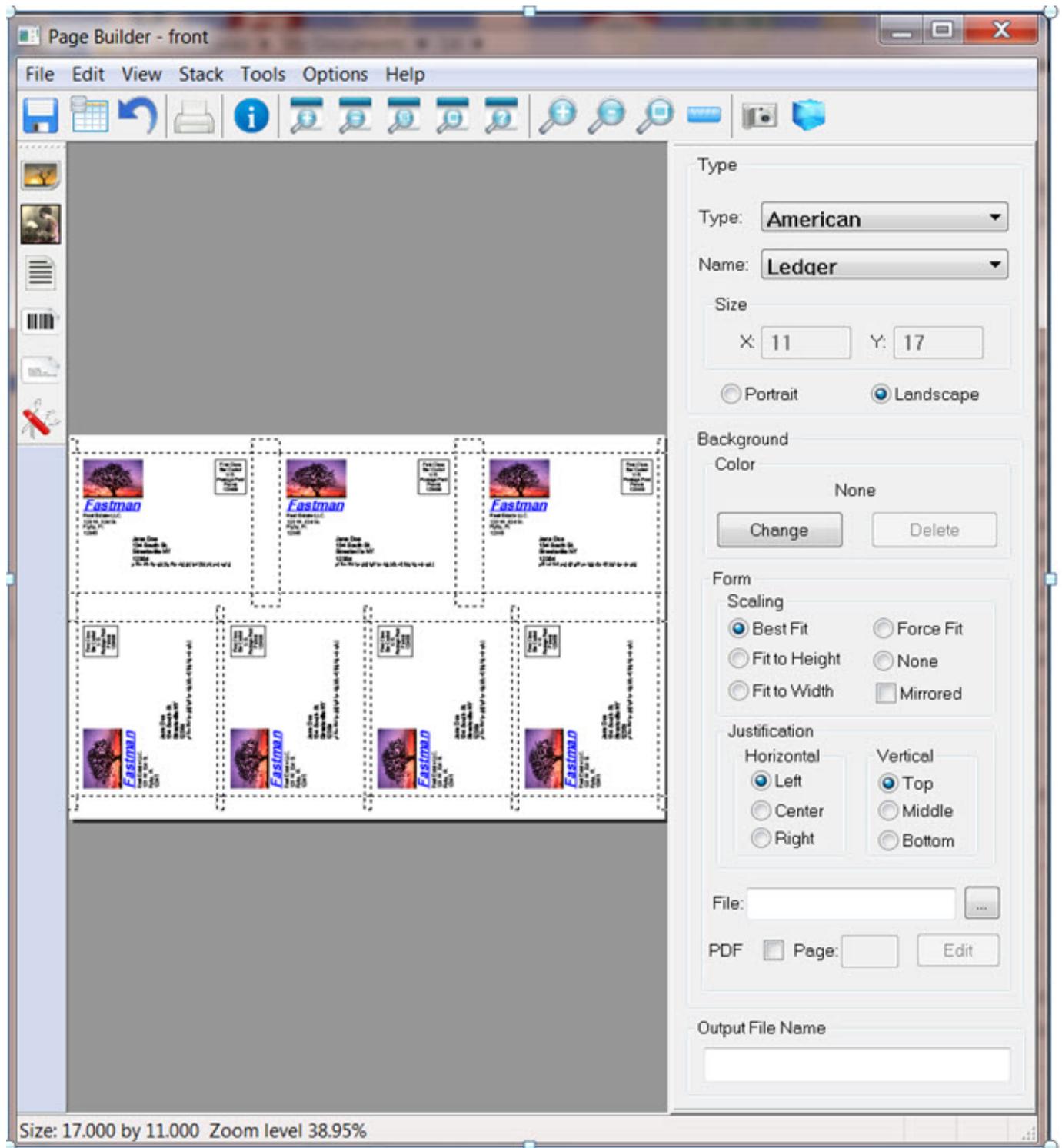


If you override the defaults such that the job won't print correctly -- PSL won't print it!!

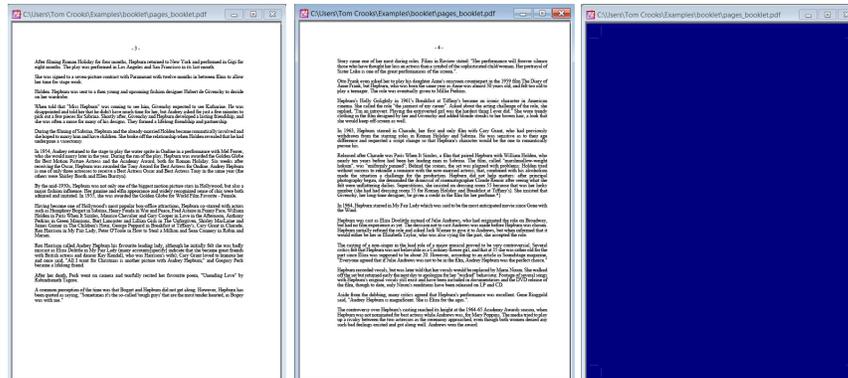
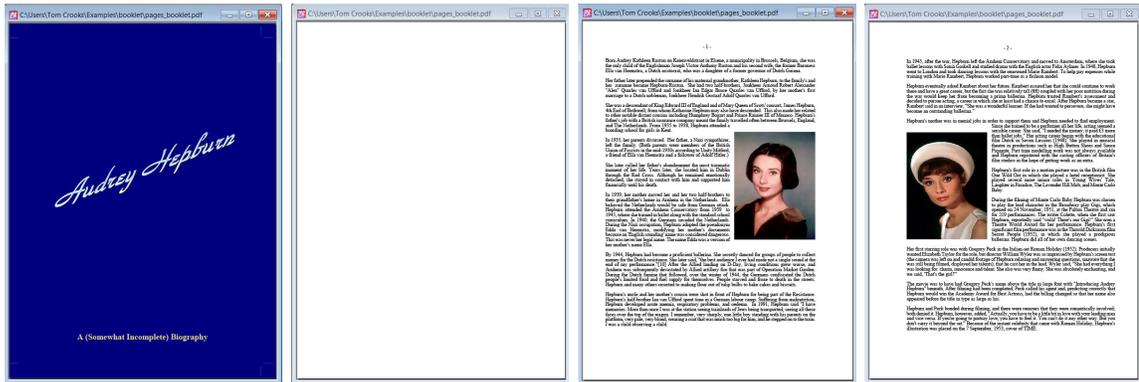
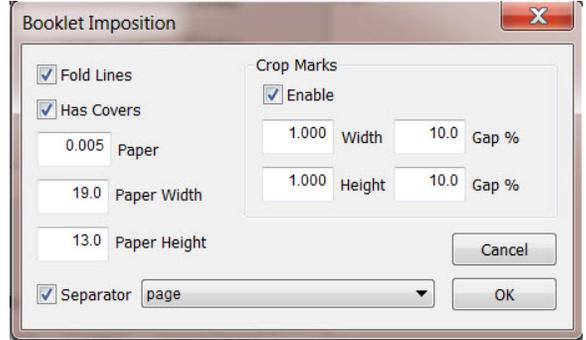
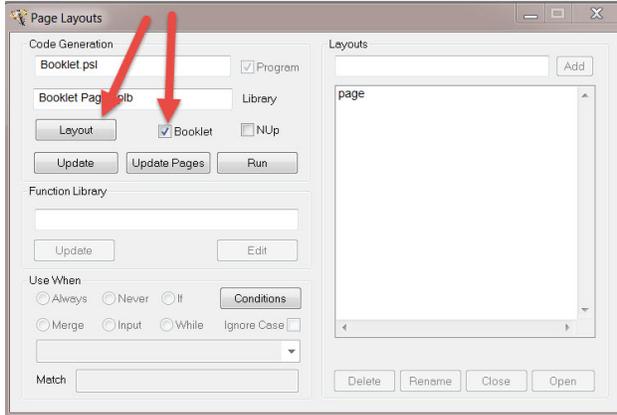
## Dutch Cut N-Up postcards

This is a variation of N-Up where the user can place the cards on the stock as they see fit sometimes allowing a better yield. As above the user creates fronts and backs, using a special custom box, which facilitates copy paste such that the cards are in the order in which they were created, fronts only, backs follow automatically.

It should be noted that while the small color images are static in this example, they easily could be drawn from a supporting file of images based on data in the merge file or any supporting files.



**Booklets** – It's the same idea as N-Up. Just compose your pages, and cover if any, the number of pages, being based on data of course, will vary as might the number of signatures. And PSL will layout the signatures with proper imposition for book trimming -- automatically. The user composes the variable pages, enters some parameters such as paper thickness, and cover issues (see below), and the booklets are automatically produced.



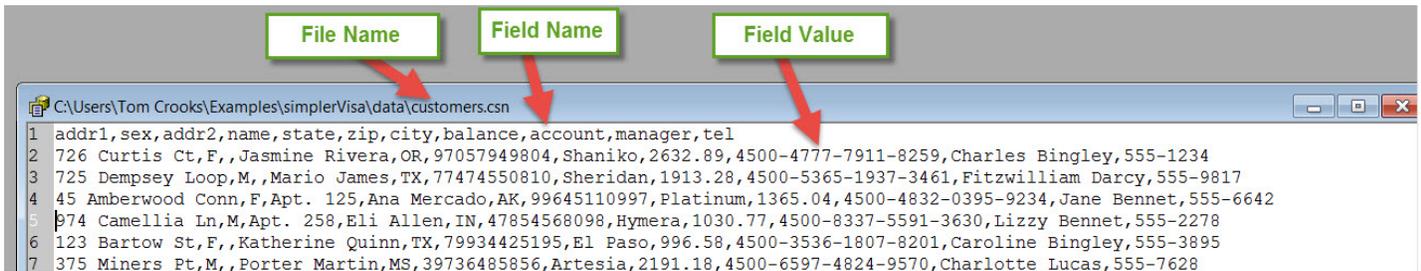


## Transactional Work -- Invoices and Statements

Composition of tabular information such as commonly found in statements and invoices can be challenging. PSL's Composition Automation can make it quite straight forward and quick. Here are the steps involved.

Most commonly data from end users is delivered as a customer file and a transaction file. Often the transactions appear in the order they occurred -- sometimes known as the "time stamp" order. As well, often there are supporting files, such as seen in the visa example with a table providing data, such as manager's pictures or vicinity maps, based on some field in the customer file (such as zip code). Examples appear below.

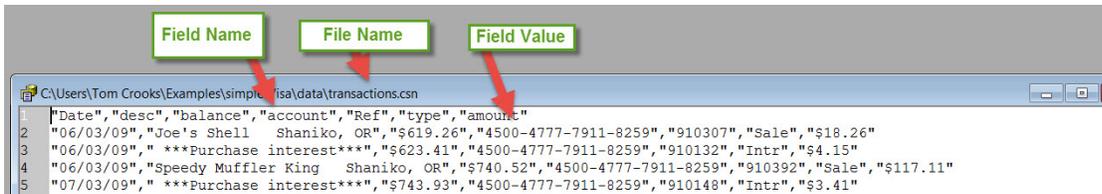
The "**Merge File**" is the customer file.



The screenshot shows a text editor window titled "C:\Users\Tom Crooks\Examples\simplerVisa\data\customers.csn". Three green boxes with red arrows point to specific parts of the file: "File Name" points to the file path, "Field Name" points to the first line of the header, and "Field Value" points to a value in the first data row.

```
1 addr1,sex,addr2,name,state,zip,city,balance,account,manager,tel
2 726 Curtis Ct,F,,Jasmine Rivera,OR,97057949804,Shaniko,2632.89,4500-4777-7911-8259,Charles Bingley,555-1234
3 725 Dempsey Loop,M,,Mario James,TX,77474550810,Sheridan,1913.28,4500-5365-1937-3461,Fitzwilliam Darcy,555-9817
4 45 Amberwood Conn,F,Apt. 125,Ana Mercado,AK,99645110997,Platinum,1365.04,4500-4832-0395-9234,Jane Bennet,555-6642
5 74 Camellia Ln,M,Apt. 258,Eli Allen,IN,47854568098,Hymera,1030.77,4500-8337-5591-3630,Lizzy Bennet,555-2278
6 123 Bartow St,F,,Katherine Quinn,TX,79934425195,El Paso,996.58,4500-3536-1807-8201,Caroline Bingley,555-3895
7 375 Miners Pt,M,,Porter Martin,MS,39736485856,Artesia,2191.18,4500-6597-4824-9570,Charlotte Lucas,555-7628
```

The "**transactions**" file in this case is in time stamp order and includes, for each record, an account number. Consequently, as directed by the "tableBox" for every record in the customer file, each, of course, having a field name of "account", PSL finds all records in the transaction file with that account number and prints the contents as a transaction in the output.



The screenshot shows a text editor window titled "C:\Users\Tom Crooks\Examples\simplerVisa\data\transactions.csn". Three green boxes with red arrows point to specific parts of the file: "Field Name" points to the first line of the header, "File Name" points to the file path, and "Field Value" points to a value in the first data row.

```
1 "Date","desc","balance","account","Ref","type","amount"
2 "06/03/09","Joe's Shell Shaniko, OR","$619.26","4500-4777-7911-8259","910307","Sale","$18.26"
3 "06/03/09"," ***Purchase interest***","$623.41","4500-4777-7911-8259","910132","Intr","$4.15"
4 "06/03/09","Speedy Muffler King Shaniko, OR","$740.52","4500-4777-7911-8259","910392","Sale","$117.11"
5 "07/03/09"," ***Purchase interest***","$743.93","4500-4777-7911-8259","910148","Intr","$3.41"
```

The "**Branches**" file contains information about branches of a number of these credit union offices, such as phone numbers and images of the branch manager. The customer file contains a "manager" field which is used as a key to bring in manager pictures, phone numbers and the like for each statement.



The screenshot shows a text editor window titled "C:\Users\Tom Crooks\Examples\visa\data\branches.csn". The file contains a table with four columns: id, manager, picture, and phone.

id	manager	picture	phone
1	Charles Bingley	Bingley.jpg	555-1234
2	Fitzwilliam Darcy	Darcy1.jpg	555-9817
3	Jane Bennet	Jane2.jpg	555-6642
4	Lizzy Bennet	lizz-b.jpg	555-2278
5	Caroline Bingley	pridepub34.jpg	555-3895

The composition automation tool involved here is the custom box called "**tableBox.**"

Below is an example of the tableBox "at work." As can be seen the tabular part of the statement is generated by filing out a form. The tableBox performs what old timers might call a "join" function. It allows designation of the field in the customer file (in this case) to use as a key to search the transaction file for records associated (in this case) with that account number.

As well it provides tools for formatting and annotating the table.

The screenshot shows the Page Builder interface with a credit card statement layout. The layout includes a header for "National Bank Of Engraving" and a "Credit Card Center" section. Below this is a table with columns for Date, Ref, Particulars, Transaction, Amount, and Balance. The table contains 10 rows of data. To the right of the layout is a "Custom" configuration panel for the tableBox. The panel has tabs for "Box", "Text", "Type", "Frame", and "Dim.". The "Type" tab is selected, showing a "Library" dropdown set to "tableBoxLib" and a "Type" dropdown set to "fileTable". Below these are property-value pairs:

Property	Value
name	trans
overflow	yes
fileName	transactions.csv
key	account
value	<account>
fieldNam...	date,ref,desc,type,
showTitles	yes
rowHeight	0.2
colWidths	0.75,0.75,3.0,75,1.2,
justificati...	ml,ml,ml,ml,mr,mr

Four red arrows point from green callout boxes to specific parts of the configuration panel:

- Arrow 1: Points to the "tableBoxLib" library name. Callout: "The name we gave this box"
- Arrow 2: Points to the "transactions.csv" file name. Callout: "The file to be searched for"
- Arrow 3: Points to the "account" key. Callout: "The primary key to use in the search"
- Arrow 4: Points to the "<account>" value. Callout: "Value from the customer file to match in the transaction file"

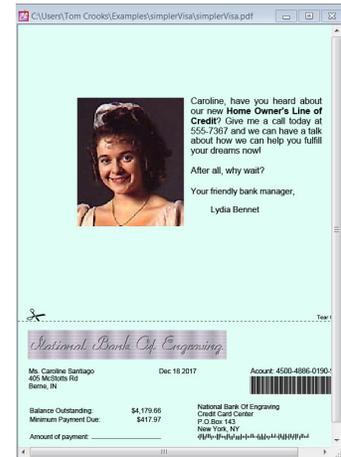
There are no limits. A project can have as many files (tables) as needed. Files can be as big as needed, records can be as long as needed with as many fields as needed.

Furthermore, for the more technical types, things like fonts and colors are data types.

Of course the usual composition tools help with the material that is not "Tables," and the result might be:



Date	Ref	Description	Debit	Credit	Balance
27/03/09	910236	Young's Optical Berne, IN	Sale	\$163.98	\$2,547.76
27/03/09	910276	Erin's Barber Shop Berne, IN	Sale	\$178.01	\$2,726.89
28/03/09	910256	Big A's Berne, IN	Sale	\$66.44	\$2,660.44
28/03/09	910274	Shoesh & Corner Berne, IN	Sale	\$68.96	\$2,591.48
30/03/09	910339	Western Union Berne, IN	Sale	\$118.78	\$3,033.40
30/03/09	910346	Baskin & Robins Berne, IN	Sale	\$106.03	\$3,228.33
30/03/09	910303	Cash Advance	Adv	\$100.00	\$3,328.33
30/03/09	910261	Gen's Italian Berne, IN	Sale	\$68.09	\$3,378.42
31/03/09	910326	Purchase Interest	Int	\$4.04	\$3,382.46
31/03/09	910235	Macy's Dept. Store Berne, IN	Sale	\$135.44	\$3,517.90
31/03/09	910219	Taunton Marine Berne, IN	Rcpt	\$15.95	\$3,501.95
01/04/09	910216	Taunton Marine Berne, IN	Rcpt	\$12.47	\$3,514.42
01/04/09	910204	Burns Photography Berne, IN	Sale	\$96.37	\$3,610.79
01/04/09	910385	Wal N Floor Decor Store Berne, IN	Sale	\$69.66	\$3,680.45
02/04/09	910334	Krazy Gym Berne, IN	Sale	\$69.74	\$3,750.19
02/04/09	910362	Krazy Gym Berne, IN	Rcpt	\$88.02	\$3,838.21
03/04/09	910248	Chyveda Duet Cleaning Berne, IN	Sale	\$128.84	\$3,967.05
03/04/09	910226	La Hacienda Berne, IN	Sale	\$39.12	\$4,006.17
05/04/09	910198	Prisidge Arts Berne, IN	Sale	\$51.55	\$4,057.72
05/04/09	910140	Hal Heibel Unions Berne, IN	Sale	\$155.14	\$4,212.86
06/04/09	910277	Baskin & Robins Berne, IN	Sale	\$102.21	\$4,315.07
06/04/09	910345	Wet In Tubing Berne, IN	Sale	\$146.11	\$4,461.18
06/04/09	910137	City Cleaners Berne, IN	Rcpt	\$37.95	\$4,269.43
07/04/09	910270	The Handcutting Place Berne, IN	Rcpt	\$46.77	\$4,179.66



Now, it is quite important to understand that what is shown here;

- 1) This was created with no programming,
- 2) Thanks to a "Logical Function" invoked from a pull down menu, there could be as many "page 2s" as needed to make sure the whole table printed,
- 3) This is a simple example, done entirely without coding, but a bit mis-leading. Such work often requires coding for such things as white space management, and the usual last minute customer required "exceptions."

An example involving coding is delivered with the examples as "visa" while this one is known as "simpleVisa."

## Flat Files

Particularly when data arrives from "legacy" main frame systems, the data appears as what we call a "flat file." Such a file structure commonly has the first 8 bytes represent a record type. Each record type is defined in a document stating the field names of the data found in certain positions in the record (think of Hollerith cards.)

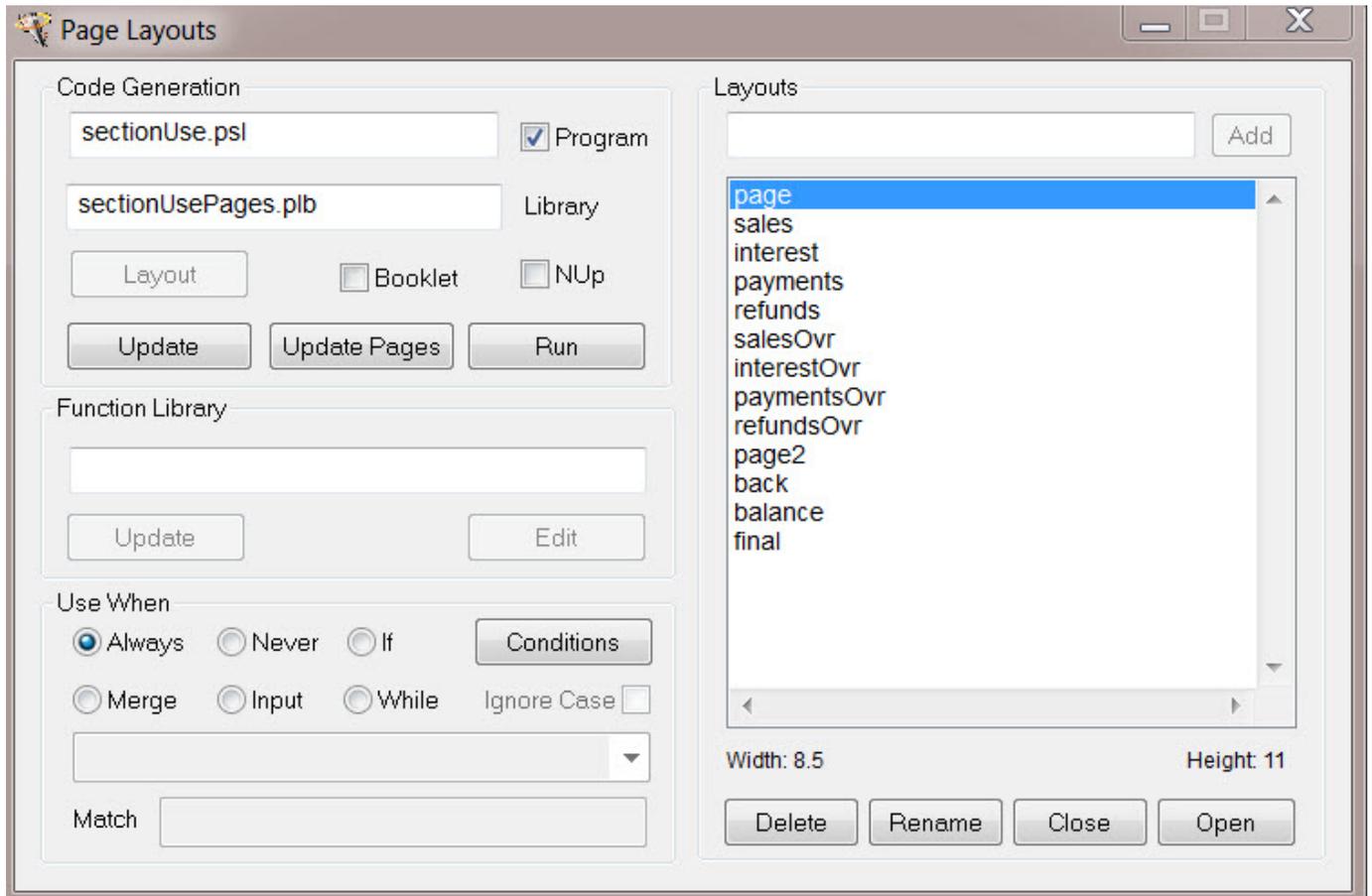
PSL has a special and very straightforward method of handling such "flat files."

**Sections** – This is a real “killer app.” It literally “automatically” handles invoices or statements where there are separate sections to the bill. For example the example shown from a credit union in which separate "sections" of the bill represent individual departments with varying billing format requirements. Another famous example is the medical practice bill in which an individual receives a monthly bill/statement from several different departments, each formatted differently. This is a very common requirement not well served by competitors.

This composition automation tool involves the tableBox discussed above when composing each of the variable sections.

There is likely a separate file format for each department.

As shown below the user identifies and composes the main layout page, then the first page for each section -- in this case sales, interest, payments and refunds; then composes the overflow pages (which may automatically be replicated as discussed above) for each of them.



The main page -- here just called "page" -- defines the order in which the sections will appear in the final document and the depth of such boxes is used to define the minimum space deemed necessary to start such a section (table).

If there is no data for a particular table it is ignored. Once the user has specified the "minimum" above PSL handles the white space management automatically.

The screenshot shows the Page Builder interface for a credit card statement. The main workspace contains a header with the bank name "National Bank Of Engraving" in a script font, followed by a "Credit Card Center" section. Below this, there is a cardholder's name and address, the date "April 1 2009", and an account number with a barcode. The main body of the statement is divided into four sections: "Section: sales", "Section: refunds", "Section: interest", and "Section: payments". A "Performance Report" link is visible at the bottom left of the main content area.

The right-hand side of the interface features a "Custom" panel with a table of properties and values:

Property	Value
contents	sales
bottomM...	0.5
backPage	#{if ([duplex]) "back
nextPage	page2
nextTop...	0.75
nextBott...	0.5
overCont...	salesOvr
overPage	page2
overTop...	0.75
overBott...	0.5
skip	#{not <status.nRec

At the bottom of the window, the status bar displays: "Size: 8.500 by 11.000 Zoom level: 77.90% section: StackIndex: 8".



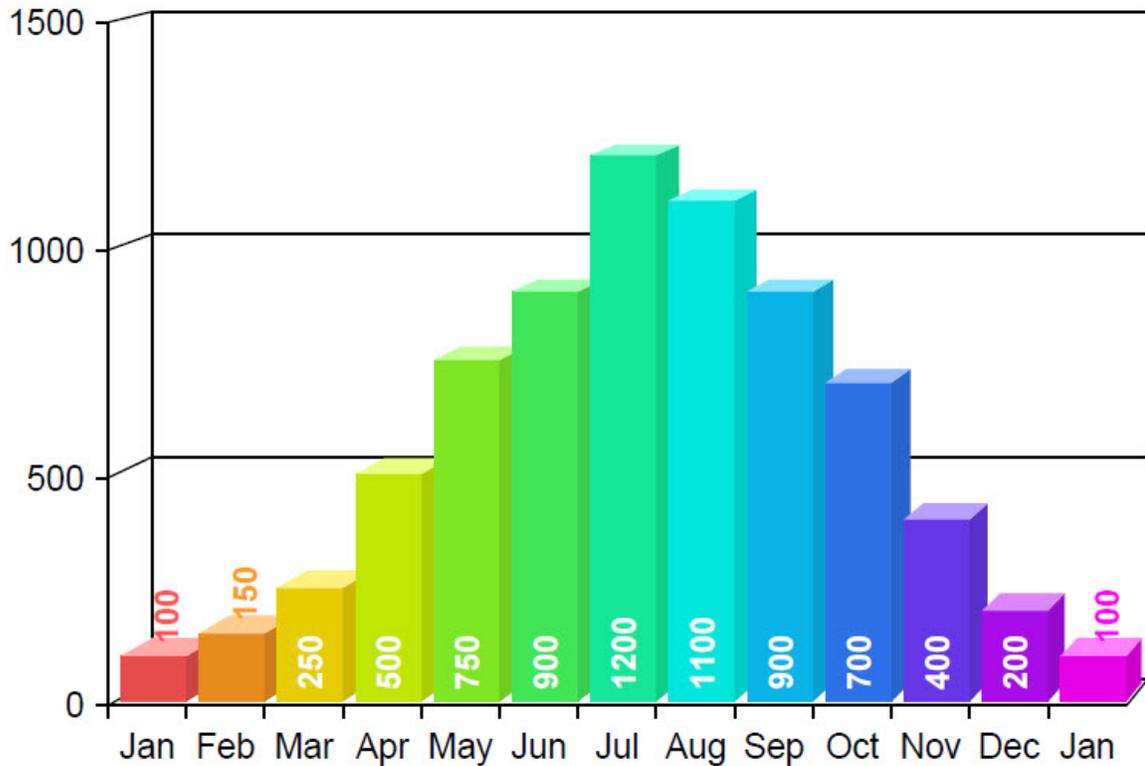
**Bar Charts** are a quite useful example of composition automation in PSL.

Coding of any sort is not required. Simply fill out a form as shown below on the right. The left part shows just some of the many options available. The table to be filled in is the one for the selected chart.

The screenshot shows the Page Builder interface with a central workspace titled "Bar Chart Modes" containing nine 3D bar chart examples. Each example is labeled with its mode and depth mode. The "Custom" panel on the right is active, showing a table of properties and values for the selected "barChart3D" type. A red arrow points to the "lineColor" property in the table.

Property	Value
viewPoint	
depth	3
values	1,2,3
colors	red,green,blue,cya
spacing	
mode	both
depthMo...	
labelMode	inside
labelDept...	inside 1
labelSpace	
minFontS...	0.5
maxLines	
exact	
lineThick...	
lineColor	

While the many options are somewhat obvious, such as using patterns rather than colors, the automation can easily result in a bar chart as shown below -- done in minutes and completely data driven. The user need only fill in the table as shown on the previous page.



The coordinate system, after having some parameters set, is automatically generated and suited to the necessary range of values to be charted. Since in PSL color is a data type, the colors can even be controlled by data.

In fact most all of the data in the table, as shown to the right, is simply a text "property" and thus can be entered;

- 1) Manually entered text, as the ones on the right are
- 2) By entry of a field name or names (data),
- 3) By use of a Value Function, for example "string functions", or
- 4) By use of a Value Statement, such as "If" Statements or "Select" statements.

