Visualization of Client Service Events Statistics

Alexandre Zolotovitski, *Member, IEEE*

***Abstract* — This paper summarizes work performed to develop new types of charts for the visualization of statistics associated with client service events: the matrix bar chart and the flower bed chart. In the problem that led to these charts, we had to visualize time series data of service events sequence related to specific clients. The flower bed and matrix bar charts can be plotted for any specific interval of time.**

The raw data were aggregated into two tables: Frequency(PrevEv, Ev) and Time(PrevEv, Ev). Then these two tables were visualized.

The charts give the user the ability to identify immediately the most business important factors.

The program was implemented as VBA - MS Excel macro.

I. Objective

In this work we have deal with the problem of visualization for Client service events to optimize work of client service. The raw data had form

Table 1.
Raw data

	DateTime			Case Types		
Case	**Ev1**	**Ev2**	**…**	**Type1**	**Type2**	**…**
Case1	T_{11}	T_{12}		$Type_{12}$	$Type_{11}$	
Case2	T_{21}	T_{22}		$Type_{22}$	$Type_{21}$	
Case3	T_{31}	T_{32}		$Type_{32}$	$Type_{31}$	
Case4	T_{41}	T_{42}		$Type_{42}$	$Type_{41}$	
……						

In our example events were service cases so variable *Case* was the foreign key identifying service case; the following columns are for DateTime stamps for service events, that could be *Creation – Received – First_Contact_SW – First_Contact_HW – Request – Pending – Closed*.
The Type columns could contain such variables as *HW_Platform*, *Product*, Geographic variables, *Customer*, *Case_Owner* and can be used for the Classification of cases. For simplicity we will show only one Type variable.

The same type of visualization can be done for analysis of events in other areas: reliability (failures), survival analysis(deceases), transport network flow analysis, network performability analysis, cross-sell and up-sell analysis in marketing, e.g. in the last case events could be purchases of specific products by a customer, e.g. we could have deal with opening a bank sequence of accounts; then instead of *Case* we have *CustomerID*, and *Event* can be *Open_Checking_Acct, Open_Saving_Acct, Open_Loan, Close_Checing_Acct* and so on;
The *Type* could be *BranchID* or *Group of Clients* and can be used for the Classification of cases. Tasks of this type are quite common in OLAP [1, 2].

Alexandre Zolotovitski is with the Sun Microsystems Inc, 9525 Towne Centre Dr, San Diego, CA 92121, e-mail: alex.zolot@sun.com, phone:858-526-9452 .

More convenient is to present the data of Table 1 in the "long" format:

Table 2.
Raw data in the "long" format

Case	DateTime	Event	Type
Case1	T_1	Ev_1	$Type_1$
Case1	T_2	Ev_3	$Type_1$
Case1	T_3	Ev_7	$Type_1$
Case2	T_4	Ev_3	$Type_1$
……			

To analyze the table we sort it by *Case, DateTime* and create variables *Previous Event* (*PrEv*) and *Time between Events (T)*:

Table 3.
Sorted data

Case	DateTime	Event	Type	Time	PrEv
Case1	T_1	Ev_1	$Type_1$	0	0
Case1	T_2	Ev_3	$Type_1$	T_2-T_1	E_1
Case1	T_3	Ev_7	$Type_1$	T_3-T_2	E_3
Case2	T_4	Ev_3	$Type_1$	0	0
……					

To analyze the quality of service we aggregate the data in Table 3 calculating count and average through Case and obtain two tables: *Frequency* (or Count) and *Time* that is average of *Time* in Table 3 :

Table 4.
Aggregated data

Event	PrEv	Freq	Time	Type
Ev_1	Ev_1	Fr_{11}	T_{11}	$Type_1$
Ev_1	Ev_2	Fr_{12}	T_{12}	$Type_1$
…				
Ev_2	Ev_1	Fr_{21}	T_{21}	$Type_1$
…				

During data aggregation from Table 1 instead of mean(T) we could use another aggregating function, e.g. mean(1/T) or Scale(T) =exp(mean(ln(T))). The latter makes sense because the distribution of time between events could be Weibull rather than normal. We will discuss this choice of aggregating function later.
Now transform the Table 4 to two "wide" (or pivot) tables:

TABLE 5.
PIVOT TABLE FOR **Frequency**

PrEv \ Ev	Ev_1	Ev_2	Ev_3	…
0	Fr_{01}	Fr_{02}	Fr_{03}	
Ev_1	Fr_{11}	Fr_{12}	Fr_{13}	
Ev_2				
…				

and

TABLE 6.
PIVOT TABLE FOR **Time**

PrEv \ Ev	Ev_1	Ev_2	Ev_3	…
0	T_{01}	T_{02}	T_{03}	
Ev_1	T_{11}	T_{12}	T_{13}	
Ev_2				
…				

The traditional way of visualizing these two tables – "Pivot Chart" – creates two stacked bar charts and we should match elements of these two charts to identify business important cases, because both Frequency and Time are important.

The simplest way to improve the pivot charts to visualize these two tables is to put in the cells of the table rectangles (bars) with width proportional to *Time* and length proportional to *Frequency*:

TABLE 7.
MATRIX BAR CHART FOR **Freq** AND **Time** (2D)

PrEv \ Ev	Ev_1	Ev_2	Ev_3	…
0				
Ev_1				
Ev_2				
…				

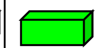
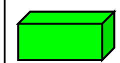
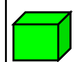
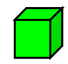
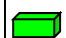
In this table the rows show frequency and average time of transactions following events PrEv and the columns show transactions that led to events Ev.

During data aggregation from Table 3 we could use the same type of chart but length of rectangle could be proportional mean(1/T) or Scale(T) =exp(mean(ln(T))). The latter makes sense because the distribution of time between events could be Weibull rather than normal.

We prefer to plot length of bars proportional mean (T) rather than scale(*T*) because sometimes lost for servicing company is proportional to time of service multiplied number of cases; in such situation areas of rectangles (bars) are proportional to $$$ amount of loss related to these transactions, so just a short glance at the chart shows which process creates the majority of issues for the company.

Usually Frequencies are distributed in wide range of values and more convenient to plot height of bars proportional to square root of frequency and plot the chart in 3D form:

TABLE 8.
MATRIX BAR CHART FOR **Freq** AND **Time** (3D)

PrEv \ Ev	E_1	E_2	E_3	…
0				
E_1				
E_2				
…				

In 3D representation volume of each cuboid is proportional to dollar amount of loss related to these transactions.

One disadvantage of this method is that each event is presented in the table twice: in raw header as *Previous Event* (*PrEv*) and in a column header as *Event* (*Ev*).

To visualize this table without doubling the events, we present events as circles or other figures (e.g. "houses") with area proportional frequency of the events and represent frequency F_{12} and Time T_{12} as arrow (or bar or petal) from Ev_1 to Ev_2 with width proportional to F_{12} and length proportional T_{12}, color of the arrow is the same as the color of circle Ev_2:

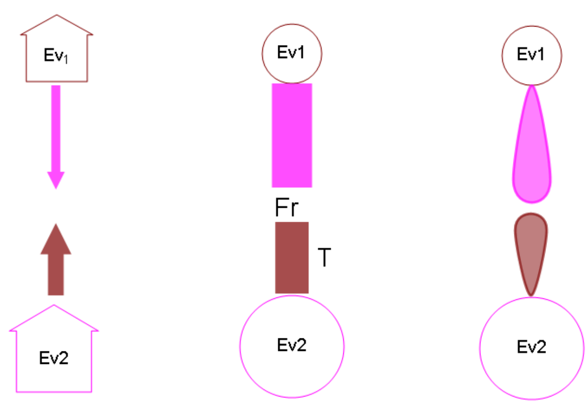


Fig. 1. Three variants of visual representation: arrows, bars and petals

We can choose positions of the circles arbitrarily; the simplest case is to put it on a big circle where all event circles "can see" each other. We use the order of event circles by increasing mean time from Event 0 (so the most petals are directed clockwise):

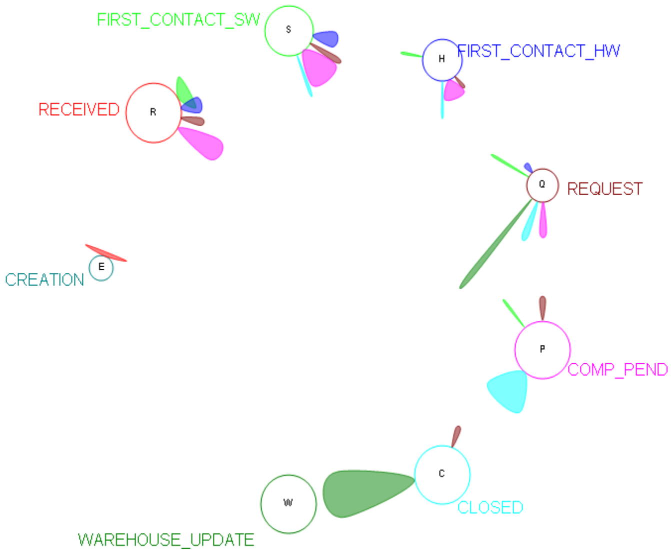


Fig.2. "Flower bed" chart

In the Flower Bed chart areas of petals again are proportional to dollar amount of loss related to these transactions, so just a short glance at the chart shows which process creates the majority of problems for the company: wide petals indicate business processes that happen frequently, long petals indicate business processes that take long time and the most important business task is to optimize processes that are both long and wide.

In our special case we did not consider the possibility that an event can follow itself, which can be expected in many other real-world process-domains (for e.g., opening checking account followed by opening another checking account). The visualization technique itself has the power to show this ("a purple circle should also have a purple petal that could be plotted out of center").

We named the chart "flower bed" chart. Another alternative could be to use standard techniques for weighted multidigraph visualization[3], but we think our "flower bed" chart is easier for interpretation and visual perception.

We have to create the "flower bed" chart (Fig. 2) for each Type of case to compare quality of service between different Types.

To increase amount of information presented by the chart, instead of bars or petals we can draw more complicated figures reflecting not only mean time between events, but also distribution of the time, e.g. histograms or violin plots.

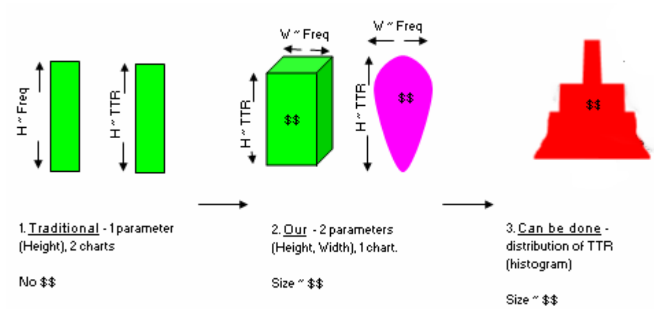


Fig.3. Evolution of chart elements

Some additional information can be reflected by position of event circles as in widely used bubble charts.

References

[1] Kesaraporn Techapichetvanich and Amitava Datta, Interactive Visualization for OLAP, ICCSA 2005, www.springerlink.com/content/wxwdvwcp5p0qjlng

[2] Andreas S. Maniatis, Advanced Visualization for OLAP, Proceedings of the 6th ACM international workshop on Data warehousing and OLAP, 2003, pp: 9 - 16

[3] Graphviz - Graph Visualization Software Available: www.graphviz.org/Gallery.php

ACKNOWLEDGMENTS

I would like to express my gratitude to Richard Castro, Dr. Jeff Glosup and Dr.David Trindade for fruitful discussion.