Visualizations and analytics for supply chains

Analyze data graphically

Areeb Kamran
Salman Ul Haq

February 26, 2013

The supply chain isn't about the flow of physical goods anymore. It is as much about the flow of information as it is about the products and the money that changes hands. With the advent of Enterprise Resource Planning (ERP), data is generally available, but converting or morphing that data into actionable information is the role of business analytics. Information and feature extraction is much easier when data is visualized, compared to working with raw numbers. This article helps developers with visualizing data for supply chains.

Introduction

Supply chain is a function of planning and forecasting

Physically, a supply chain is about the movement of goods. But in today’s fast-paced world, where a minute of lost time is not an option, idle manufacturing facilities are a waste of resources, where losses are not just measured in actual numbers, but potential numbers as well (opportunity cost). Supply chains have become much bigger than just stock movements and time related positioning of resources.

Supply chains have become complex, convoluted, and highly integrated systems. Such is the complexity that you just can't work on them efficiently anymore without a lot of help from advanced computing techniques and intricate data presentation systems. The numbers involved are too complex. ERP systems have helped to make the data generally available, but converting or morphing that data into actionable information remains a challenge.

Status quo for supply chain analytics: reporting raw data

The inherent nature of the way ERP systems work make reporting a problem. Every single minute, a transaction is recorded in ERP systems, which constitutes the data. When it comes to reporting, the data is compiled from the bottom up. These minute details are clustered up and compiled at each step to form the report for that particular level of function. For example, the report could contain every movement of a single material out of a store for a store keeper, or it can form an
aggregate of products moved in/out from an entire warehouse for a warehouse manager. For the warehouse manager, if any detailed analysis is required, a goods movement over six months can turn out to be a 50,000 row report, which can be tedious to work with. A part of the report might look like Figure 1 (from an SAP ERP system):

**Figure 1. SAP ERP system**

<table>
<thead>
<tr>
<th>SLoc</th>
<th>HyT</th>
<th>Mat.Doc.</th>
<th>Item</th>
<th>Posty Date</th>
<th>Quantity</th>
<th>BuN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1120</td>
<td>411</td>
<td>E 4902167899</td>
<td>15</td>
<td>20.07.2010</td>
<td>1.682</td>
<td>KG</td>
</tr>
<tr>
<td>1120</td>
<td>101</td>
<td>E 5000765749</td>
<td>8</td>
<td>20.07.2010</td>
<td>1.682</td>
<td>KG</td>
</tr>
<tr>
<td>1120</td>
<td>412</td>
<td>E 4902291967</td>
<td>1</td>
<td>19.11.2010</td>
<td>100,920</td>
<td>KG</td>
</tr>
<tr>
<td>1120</td>
<td>601</td>
<td>E 4902291975</td>
<td>1</td>
<td>19.11.2010</td>
<td>100,920</td>
<td>KG</td>
</tr>
<tr>
<td>1120</td>
<td>412</td>
<td>E 4902300130</td>
<td>1</td>
<td>26.11.2010</td>
<td>100,920</td>
<td>KG</td>
</tr>
<tr>
<td>1120</td>
<td>601</td>
<td>E 4902300131</td>
<td>1</td>
<td>26.11.2010</td>
<td>100,920</td>
<td>KG</td>
</tr>
</tbody>
</table>

**Need for data presentation: Morphed or processed data due to large data sets**

The basic problem with reporting data is its sheer size. Leaving the complexity of supply chains behind and just focusing on one aspect of the chain for example, there are stock movements as follows:

- One product across the chain in a number of months
- All products in one plant for a month
- The sales from one particular business area

These are all examples of data that can be extracted easily by using standard reports that are provided by most major ERP systems or custom developed reports, but just the amount of data they have can be overwhelming and unsuitable to be consumed in traditional report formats. Traditional reports present data in a row/column format as Figure 1 shows. But assuming it’s a mere 2,000 line report for any of the previous scenarios, there are four ways to analyze the report:

- **Option 1**: Study the 2,000 lines as displayed
- **Option 2**: Restrict the data that is shown in these reports by some criterion or filters
- **Option 3**: Summarize the data by clumping the data (date ranges or product ranges or physical storage ranges), or use numerical averaging
- **Option 4**: Use line graphs for trend analysis

Option 1 is tedious and not helpful in either trend analysis or anomaly spotting because of its size. Option 2 limits the data that can be analyzed, which usually results in a loss of insights or relationships. Option 3 suffers from that same loss, which can happen when the data is clumped or averaged. The only viable option to encompass all the data is Option 4, which is a form of visualization in the simplest form: a graph. Graphs work well when you view them in one dimension, such as time series data or sales versus production data. But what about sales versus production data as a series of time? This is where the limitations of normal graphs prohibit further analysis.
Non-conventional data visualization techniques for supply chain data: an example

First, why do visualizations work? The goal of visualizations is to assist you in understanding data by leveraging the human visual system’s highly tuned ability to see patterns, spot trends, and identify outliers. The only viable option for a deep and precise analysis is visualization of data and transforming it from numbers to visuals. All of the other three report types have been shown to restrict this kind of analysis. Figure 2 shows an example visualization of stock versus sales against time:

**Figure 2. Stock vs. sales against time visualization**

This graph is used for the *The Wealth & Health of Nations* (see Related topics), but the very same nature of the problem, with respect to data representation, means this can be equally applied to stock movements.

By using the circular shape in Figure 2, a third dimension has been added: the volume of the shape itself, which is independent of the other two dimensions (X and Y axis). This gives the graph an extra dimension. Then, by making the graph interactive and letting time change by itself with
animation or by controlling the motion manually by movement of a mouse, a fourth dimension is added. Figure 3 shows the graphs in 2 points of time. This achieves two advantages:

1. Patterns can be seen easily even with large numbers of data rows.
2. Higher dimensional data can be represented (in this case, four dimensions instead of the usual two for line graphs or Cartesian plots).

The reason why this approach works is simple: It relies on Edward Tufte's principle for visualizations, keeping things simple, uncluttered, and giving information visually rather than in numbers because the mind's visual systems are much more developed than any other sensory system.

Another example of an effective visualization that you can apply to many aspects of a supply chain is a choropleth map, as shown in Figure 3:

**Figure 3. Choropleth map visualization**

Figure 3 is an example of a static visualization using a unique technique. Whereas choropleth maps are traditionally used by a census or population-based surveys, you can use them just as effectively for supply chains involving geographic regions. The very shape of the canvas or plot adds an additional dimension that conveys the location. You can add another dimension by varying the color intensities. An interesting feature to note here is that in total, the map in Figure 3 only gives us two dimensions, which is the same as a traditional plot. However, the very nature by which it displays the information is very intuitive. Nothing in text or numbers can convey the geographical information as well as this illustration can. For example, by using such a map in supply chains to show sales *heat maps* over regions, vendors and trade volume, warehouses and stock positions, or purchase orders and relative demands from areas, you can gain deep insights easily that can also help in other decisions. For example, restocking various warehouses according to sales demand as seen here can be useful. Although such decisions would require a lot of thinking otherwise, with such visual aids, the actions you need to take seem evident.

There is one major implementation issue that you must address in such visualizations: When data presented in reports or numerical formats is sampled over time or space, it can have missing values, zero values, or sometimes extreme values. That works well with numerical reports, but
when converting that into visualizations, you need to clean and *smooth* the data. Otherwise, such animated visualizations would essentially *hop around* from place to place and figures can go missing in-between (representing missing values). This has an impact on the goal you are trying to achieve: looking at patterns. Such data distorts patterns, and erratic behavior takes the focus away from smooth trends, which beats the purpose of observing the trends. Just like extreme spikes and dips in a line plot would make observing trends difficult over there, such data also makes these visualizations less effective.

To solve that, you can perform data smoothing (removing obvious extreme values or *noise*) and fill in for missing values using interpolation, so that the end result is smooth and displays trends easily.

You can apply all the previous techniques in business scenarios by using the following chain:

**Extraction -> Analytic Engine (which could be native Business Intelligence (BI)) -> Visualization**

For most systems, it would be a useless exercise to develop the entire chain. As I stated, the data is not radically different in this approach as much as the presentation of data. So you can use the regular data sources, because the reports are deployed and being used.

For the second part of the chain, starting from scratch is not a good idea. Most existing ERP deployments already employ some kind of BI solution. The purpose of these BI systems is to get data and *slice and dice* it. This allows you to study data from different perspectives. However, it does not resolve one problem that visualization addresses: the volume of data.

After the BI systems have processed the data, you can remove it (in most cases in spreadsheet formats, CSV files, or even JSON) and then apply visualizations to it. A great number of libraries are available, and a lot of them use JavaScript and are browser-based, taking the data in using JSON or simple CSVs. One great advantage of using these libraries is that by making the solution browser-based, no new installations are required and simply running the instance of these implementations on the same server running the ERP makes them available to users without a lot of hassle.

One of the best such libraries available is D3.js. It is a JavaScript-based library that lets you bind arbitrary data to a Document Object Model (DOM) and then apply data-driven transformations to the document. It uses SVG for making these visualizations, so it only works with modern browsers (IE 8 and earlier versions do not support this). D3.js is properly documented and has many examples available that you can use as a great starting point.

However, not everything in reporting is complex. You can also deploy simple visualizations by following the same cycle. They still retain the advantage of keeping things visual rather than numerical, so you can effectively use simple line charts, bar graphs, scatter plots, box and whisker plots, or choropleth maps.
For example, the graphic shown in Figure 4 was taken from the New York Times (see Related topics) and shows the budget breakdown for President Obama's 2013 budget proposal. You can use it in the exact same way for costs incurred on stock or on production as a breakdown. Hovering the mouse on an individual circle brings up the details, while overall, the graphic gives a nice and intuitive idea of the cost breakdown, something that you cannot achieve using numerical reports alone.

Figure 4. President Obama's 2013 budget proposal

Another option available is the Circos software package for visualizing data and information. It visualizes data in a circular layout alone.

Trend analysis and forecasting for supply chains: methods and utilities

Tools like IBM SPSS are also available, but they are most often used at the enterprise level (see Related topics). Also, tools like SPSS are useful in the analytics part of analysis. Some other helpful tools for deploying visualizations are:

Google Refine: This is easy to use and to manipulate data with. You can do almost everything it can do using spreadsheets as well, but Refine does it much faster and easier. It acts as both a spreadsheet and a database by allowing row and column operations just like relational databases. Google Trends can simplify pre-processing, relabeling or mixing and matching of data, calculations, and other mathematical or statistical functions. It can also utilize its own scripting language (see Related topics).

R Platform: For the technically adept and detail-oriented user, you can use the Open Source R platform as a statistical analysis tool for the data once it has been extracted from the system.
can use R from multiple IDEs, the most popular being the one provided by the R project itself and RStudio, which is a cross platform IDE (see Related topics). These are options that are usually not available in the off-the-shelf systems and also are usually much more complex for ordinary business users. See Related topics for a link to an article published in the New York Times that compares R to other commercially available systems.

**Conclusion**

You can't manage what you don't measure, and you can't measure if you can't make sense of what data you have. The best way to make sense of data is to tap into the highly developed visual sense of the human mind. This article introduced you to some useful tools for visualizing data. However, you don't need to reinvent the wheel and start everything from scratch or redeploy installations. Web-based utilities are a great starting point to use available extracted data and build quick demonstration visualizations. If their use catches on and users find real value in it, more complex solutions can be built by using advanced packages. But as a starting point, you can use the methodology and tools that are mentioned in this article to build simple and straightforward proof of concept demonstrations.
Related topics

- For an excellent article on the growth in popularity of the R language, read "Data Analysts Captivated by R's Power" from the New York Times.
- Check out Mike Bostock's D3 visualization of the Gapminder's Wealth & Health of Nations.
- View a choropleth map that encodes unemployment rates from 2008 with a quantize scale ranging from 0 to 15%.
- Learn about IBM SPSS predictive analytics software and solutions.
- Visit the R Project for Statistical Computing to learn all about R and to download the programming environment.
- Follow your Rules, but listen to your Data: Watch Alex Guazzelli's presentation at the Rules Fest 2010 Conference, which focuses on the differences between data-driven and expert knowledge as well as the benefits of bringing the two together.
- "Predicting the future, Part 1: What is predictive analytics" (Alex Guazzelli, developerWorks, May 2012): This is the first article of a four part series that focuses on predictive analytics.
- "Predictive analytics in healthcare" (Alex Guazzelli, developerWorks, November 2011): Read this article on the challenges and applications of predictive analytics in healthcare.
- D3.js is a JavaScript library for manipulating documents based on data.
- Google Refine is a power tool for working with messy data, cleaning it up, transforming it from one format into another, extending it with web services, and linking it to databases like Freebase.
- Get the Circos software package for visualizing data and information.
- IBM SPSS Statistics 20 (formerly SPSS Statistics) puts the power of advanced statistical analysis in your hands. Whether you are a beginner or an experienced statistician, its comprehensive set of tools will meet your needs.
- To listen to interesting interviews and discussions for software developers, check out developerWorks podcasts.
- developerWorks on Twitter: Join today to follow developerWorks tweets.