“Visualization Software: Deliver information quickly and effectively”

Slide 1 (title)

Hello and welcome to this introductory presentation “Visualization Software: Deliver information quickly and effectively”. In the next few minutes, we will explain what we mean by Visualization Software and explore a variety of advanced graphical displays used in today’s business applications.

This presentation is designed for anyone involved with the process of designing and building these types of user interfaces. We’ll use many screen shots and product demonstrations from IBM’s set of user interface products to show you what’s possible.

Slide 2

First, let’s define what we mean by “data visualization”.

Here we have an example of a user interface that is NOT very graphical. We have a set of data tables, the results of a set of calculations used to plan out a supply chain. This text data may be perfectly meaningful to the trained end user, but is this is really the best way to present the information? It is difficult to spot any sort of cluster or trend when the data is presented purely as text.

Slide 3

Now we see the exact same data, but this time it is presented more graphically. The information that was previously stored as rows in a table is represented here as lines that connect cities. It’s easy to spot where the two main distribution centers are with this visualization display. And by varying line thickness or colors, other information can be shown, and the end user can gain additional insight.

So And, Ie’ve shown you two different ways that the same data can be depicted. For some applications, text is enough and for others, the graphical display makes more sense. Often, both techniques are used together. Later in this presentation, we will see some examples of that.

But coming back to our definition of “data visualization”....well, let’s say that it means to use graphics to represent the underlying system data in a way that is more natural and more meaningful to the end user.

For the people responsible for designing and building software user interfaces—possibly, you—this is very important indeed.

Slide 4

We just saw one example of a graphical visualization technique, a map, and now lets see some others.

The first is a chart display.

There’s the classic X-Y plot. Here we have bar charts

Charts can also be combined, as in this stock chart example, which has 3 parallel charts.

Some applications benefit from showing the data in three dimensions.

Slide 5

And there are also more specialized chart displays, like

The pie chart
Spider chart
Polar charts

and Treemap charts, which is useful for trying to make sense of very large sets of data.

These are all good examples of the variations that can be found in just one type of data visualization technique: namely, charts.

Of course, besides charts, there are many other types of graphical displays that are frequently used in business applications.   Including:

Slide 6

Time-based displays. Here we have a Gantt chart, used for project management.

A resource load chart, used to show how resources are scheduled over time.

Here's a calendar display, like those seen in on-line planners.

Slide 7

Then there's the diagram display.   A diagram is a set of interconnected objects. The user derives meaning from the relative placement of the objects and, maybe also, from the way the connecting lines are drawn.

Here is a classic example: an organization chart. It shows the reporting structure of a business

Slide 8

This diagram depicts a business process model. The objects represent steps in a process and the lines show the flow between the steps. Note that objects in a diagram can be quite complex and so can the way in which they connect to each other.

Slide 9

And this third example is a class diagram display, such as that found in a UML editor.

Clearly, diagrams are quite graphical by nature…it's difficult to image how this data would have be represented as simple text.

Slide10

In business applications, map displays are used to show where their underlying system data is located.

Here we have a geographic map of a city, overlaid with graphics that represent system resources. The objects are placed on the map at their correct geo-spatial location.

Slide 11

Another type of map display is a heat map. It shows the distribution of data and draws attention to the clusters.

Slide 12

And, for some applications, maps must be 3 dimensional, so that the elevation of objects can be visualized. Here is a defense application's user interface, showing flying aircraft and dedicated airspaces.

Slide 13

There are many other types of data visualization displays, of course. Many are characterized by the use of domain-specific graphical objects.

Such as

- Gauges and dials. These are often used to display key performance indicators in a business dashboard.
- or custom graphics representing physical equipment, either as an abstraction…
- …or as a photo-realistic display.
- More traditional icons can also be used, with specific flags to represent different status.
- and they can be grouped together to represent networks of information

DEMO

So now let's put it all together. Here we have a typical network monitoring application. We have a map display, which shows objects which are geo-referenced, showing their correct physical locations. We also show the interconnections between the objects.

From time to time you'll see the "coding" on these objects change: the color or shape will change because it's reflecting the current status of the underlying system objects. So we have some very highly custom graphic objects on top of a map displays.

Synchronized with this map display is this table display, down below. This is more text-based, but these two displays, along with this tree view, on the left, give the end user a very good situational awareness of what's happening with his network.

Slide 15

Let's take a quick look at some of the challenges faced by developers who are tasked to deliver such an advanced user interface.

An early decision to be made is whether or not another product's user interface can simply be adapted to meet the needs of the new project.

If the requirements for the new project are too specific, the user interface team will be called upon to build a solution. Factors they will have to consider include the development language—for example, Java, or Javascript-- how the application will be deployed—will it be installed on each desktop, or run inside the browser?--, and the talents of the existing development team—do they know how to build a multi-layered map display, for example?

No development team today would ever consider building the entire user interface from scratch. There is a wealth of reusable software out there—let's refer to them as "components", or building blocks—that have been designed to facilitate the assembly of many types of advanced user interfaces. Some of these are available in the form of open source projects and some are commercially available, from companies like IBM.

Filtering through the options to meet these challenges is part of the job of the user interface development team.

Slide 16

IBM offers a wide variety of products for building the types of displays we've seen in this presentation, including charts, maps, diagrams, time-based displays and much more.

These products are packaged as ready-to-use and ready-to-extend components, along with tailoring tools—such as editors—and also with the underlying software development kits. In this way, the developers can quickly integrate new displays and, at the same time, rest assured that the SDK will give them full access to the underlying parts.

The IBM Visualization products are available across a number of development and deployment platforms. The JViews family of products, for example, consists of 7 Java-based products, each tailored to a specific type of advanced user interface display. Likewise, there are offerings for Adobe Flex and AIR, cross-platform C++, Ajax, and the .NET platforms.

It's important to note that the various IBM Visualization products do not have an identical feature set from platform to platform. The Flex charting displays are not identical to the Java-based displays—this is mostly due to the differences in the underlying platform technologies.

FIRST DEMO

Let's go ahead and look at some demos of some typical applications.

Here's a modeling application built with an IBM visualization product. We have a canvas in the center, and on the left we have a palette of different types of business process management objects.

So I'm going to go ahead and start drawing some objects on the screen. I'm going to show how they might be connected. I'm creating a process flow diagram. I'm going to connect these up…notice the nice visual feedback that you'd like to have when you're doing this kind of application.

Okay, I've done a little "time lapse" here and now I've got a little more of a diagram built up but it's a mess, isn't it? It's not organized at all and it's very difficult for a user to get any information out of this. So what we're going to do is apply some "graph layout" technology to it. This will organize the diagram so that it's much more readable. I'm going to pick a type of layout that I want—basically, I'm going to indicate that I want it to "flow" from left to right. I hit the button, I tell it to "lay out", and here we have the same diagram I just created but now everything's been automatically organized for me.

This was a very simple example. Here are some more sophisticated types of diagrams that can also be created.

NEXT DEMO

A second type of demos we will look here is another monitoring display. We have a heat map here that is synchronized with a table, a scrolling table down at the bottom. The table has textual data that is indicating where the users are using a particular application. Their locations are listed in the table and also on the map.

As I move the cursor over the map, I can see the number of “hits” each country has produced. So, China, here, has had 35 hits.

I could also color the map so that a color is used to indicate the total number of hits: green, yellow, orange, or red. Green would indicate 0 to 25 and, if you looked at the United States, it’s just updated to “yellow”, which indicates it now has more than 25 hits.

This is a very nice way to get a good overview of “what’s happening” and “where”.

NEXT DEMO

And in this final application example, we have a data analysis display. We have information about the different countries in the world and we’re able to view them in order to try to understand the data better. We can see the data displayed on the map here and the map is colored according to different criteria: population, population growth, or GDP per inhabitant (how rich a country is). We can see that the US is quite rich—it’s red—and Africa is one of the poorer continents.

We can drill down into a country, and get more information that way and see specific information for that country. The very nice panning and zooming effects make the application more appealing.

So right now the map is colored according to GDP per inhabitant. And there is an alternate way of looking at the data: we can look at the same information in a treemap view. Here, the size of the boxes indicates the relative population of the country whereas the color is showing how rich they are. So we see that Africa has the poor countries and Asia has the most populated countries..

Slide 17

As mentioned before, user interface developers have a variety of open source and commercial choices, so why select one of the IBM products?

IBM’s Visualization products have a long history of success.   They originate from IBM’s recent acquisition of ILOG, a company that has specialized in reusable software components since 1987. The components developed then are no longer available, of course, but the latest generation of products still has the same goals that the original product had: to meet the most demanding needs—in terms of both features and performance--of mission critical business applications.

Specifically, the IBM ILOG Visualization components all share the following characteristics:

- They’re all completely customizable—everything look and feel of the interface can be controlled
- They designed to handle very large data sets—thousands of objects in a diagram or chart, for example,
- And difficult problems, such as arranging the elements and connections in a diagram into a readable workflow, are provided out-of-the-box.

Naturally, all of these products are offered with technical support and, if needed, consulting. IBM offers these services on a worldwide basis.

Slide 18

Let's summarize what we've covered.

IBM ILOG Visualization products provide advanced graphical user interface components. These are used in a wide spectrum of mission-critical business applications.

The products are available on a variety of development and deployment platforms, and by display type.

The benefits to using these products are two-fold:

First, they save time—at least 50% of the user interface effort—because developers can immediately leverage these pre-built, tested components.

And, second, they ensure that a top-notch user interface experience is delivered to the end user.

Slide 19

We hope you have enjoyed this brief introduction to IBM's Visualization product line and we hope you come away with a better understanding of how these types of displays can help your software deliver information more effectively to its users.

We encourage you to visit the IBM Visualization website, where you can watch other presentations, try one of the many interactive demos, and download a free, full-featured trial version of one of our products.

Good luck with your development project! And thank you for listening.