

DataVista Spectrum™



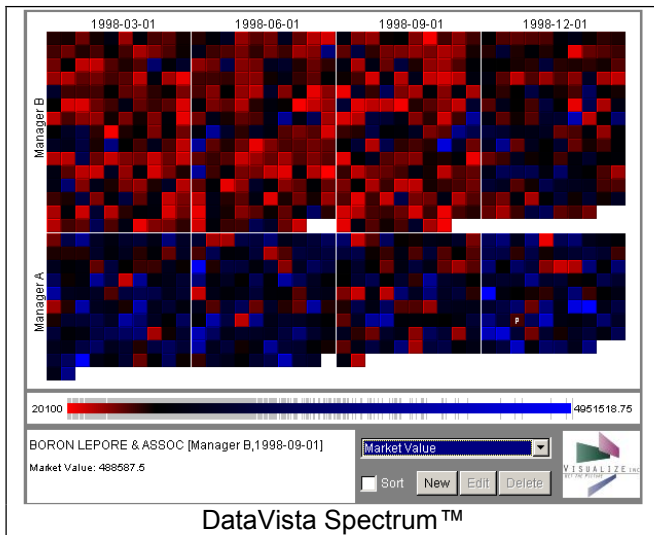
Intuitive display of large, dynamic data sources

Introduction

DataVista Spectrum™ is a Java applet that converts large collections of data into interactive color maps that provide easy, intuitive access to all available data. A powerful equation editing facility enables the creation of new data series' that are a function of existing data.

Typical applications of DataVista Spectrum include risk management, network monitoring and e-commerce site management. DataVista Spectrum is effective in any environment that requires the monitoring of large and dynamic data sets.

DataVista Spectrum supports a pluggable data architecture, enabling the display of any proprietary data source in addition to JDBC and data stored within its open, XML-based configuration file.



Visualization

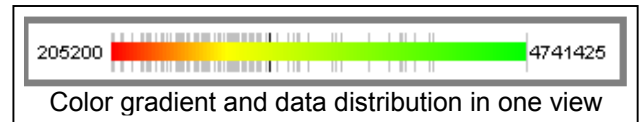
Data visualization techniques are applied throughout the DataVista Spectrum interface in order to improve the speed at which data are comprehended and acted upon.

DataVista Spectrum represents each piece of data as a colored square arranged within a regular grid. A data item is assigned to a row and column based upon user-defined attributes. There is no limit to the number of rows and columns or the size of the underlying database.

This simple technique of employing colors to represent values enables salient features and patterns of large databases to present themselves

much more quickly than similar data represented as tabular data or through common charts and graphs.

Colors are assigned to data through the use of a configurable gradient. This gradient, along with a visualization of the data distribution, is displayed below the actual data. Advanced users may develop custom gradients and plug them into DataVista Spectrum.



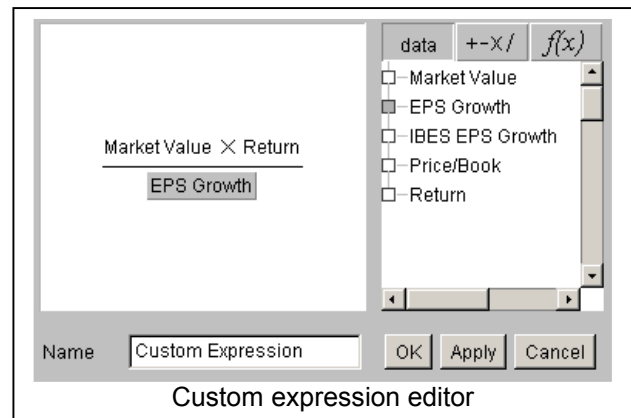
Expressions

DataVista Spectrum provides the user with the capability of extending upon existing data through the creation and modification of custom expressions that are represented and functions of the existing underlying data. Expressions may be included at execution time, or created by the user from within this flexible applet.

Expressions are created through a flexible, and easy-to-use expression editor interface. Through the use of the mouse, and without any typing, complex equations based upon the underlying data are developed, displayed and later refined.

Operations range from simple addition, subtraction, and multiplication to complex Boolean expressions and conditions. The various built-in functions include trigonometric, logarithmic, rounding and others.

The ability to add and modify custom expressions greatly enhances the flexibility and power of DataVista Spectrum to enable better decisions based upon your data.

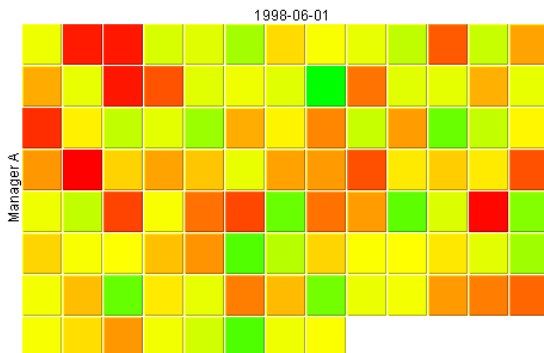


Dynamic Data

DataVista Spectrum also provides a powerful framework for monitoring dynamic data sources. The Internet has significantly increased the necessity for quick reactions to rapidly changing conditions. This situation is particularly relevant in financial, network and web-site monitoring applications. DataVista Spectrum aptly handles data that change over time, continually updating its display and custom expressions to present them in a continuous fashion.

Data Access

DataVista Spectrum may obtain its data from any of several sources, thereby providing seamless integration into most enterprise architectures. The first source is an open, XML-based format, compatible with many of the emerging XML-aware applications. Secondly, data may be imported from any JDBC-compliant data source (which includes all ODBC data sources through a JDBC-ODBC bridge). Lastly, a pluggable architecture allows any Java™ object implementing a simple, well-documented interface to serve as a data source.



XML Configuration

DataVista Spectrum is configured through a very flexible XML-based configuration mechanism. All aspects of DataVista Spectrum's appearance and functionality are configurable ranging from data access to chart presentation to exception handling. The same XML that configures the applet may be employed to configure the servlet, further reducing the effort expended in deploying a unified client-server solution.

Applications

DataVista Spectrum is able to import and display data of various types from a wide range of sources making it useful in many situations. Possible applications include financial research, risk management, real-time network monitoring, market analysis and customer relationship management.

For example, a risk management application would partition asset positions into various groupings such as portfolio manager and asset class. Each position could then be assigned a color that is proportional to its contribution to the overall risk of the portfolio. These data then would update over time to reflect current market conditions.

Features and Benefits

DataVista Spectrum™ possesses the following features

- Compatibility with any browser or operating system with a Java™ Virtual Machine (JVM)
- Use of visualization tools to improve data comprehension and accessibility
- Support for dynamic data from a variety of data sources, including proprietary databases and feeds
- Intuitive interface ensuring a shallow learning curve and quick adoption

These features provide the following benefits

- Reduced Total Cost of Ownership through thin-client implementation
- Improved decisions, reducing waste, costly delays and errors
- Dynamic data handling to ensure decisions reflect current conditions, reducing errors
- Easy integration into your systems, reducing implementation and maintenance costs

About Visual Technologies

Founded in 1996, Visual Technologies mission is to design, develop, and deliver high-end data visualization software for enterprise-level, Internet-based services. Our DataVista™ and VantagePoint™ software products provide major corporations and their customers with the ability to make timely decisions through visual representations of complex and dynamic data sets.

Visual Technologies, LLC
11811 North Tatum, Suite 3031
Phoenix, Arizona 85028
(877) 861-0999

www.visual-technologies.net