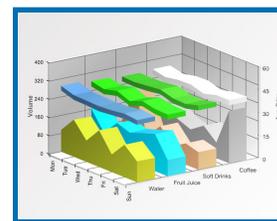
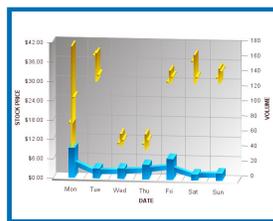
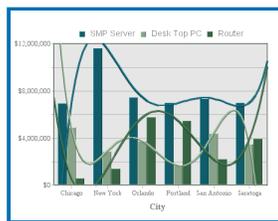
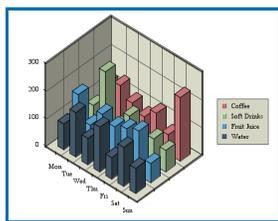


## High-Powered Data Visualization

**A picture is worth...** You know how the cliché ends, but if the picture is an interactive, dynamic chart than it's probably worth a great deal more than 1000 lines of data. In fact nothing conveys salient information better than a well designed chart or graph. EspressoChart® from Quadbase Systems provides a feature-rich, powerful Java™ solution that allows users to easily enhance Web pages and applications with dynamic charts.

Written in Pure Java, EspressoChart easily deploys in most application environments, and can generate over 30 different types of charts for applications and Web content. The powerful Java API allows charts to be created and modified programmatically. With hundreds of customization features, EspressoChart makes it easy to create powerful data presentations for virtually any situation.



## Over 30 Different Chart Types

EspressoChart supports over 30 different two-dimensional and three-dimensional charts. With advanced 3D algorithms, EspressoChart was the first product to offer true 3D charts for Java, allowing real-time pan/zoom, rotation, and light source modification. In addition, EspressoChart supports many different types of composite/combination charts, allowing users to plot different data sets on the same chart in a number of configurations.

### Chart Types:

- Column (2D & 3D)
- Bar (2D & 3D)
- XY(Z) Scatter (2D & 3D)
- Line (2D & 3D)
- Stack Column (2D & 3D)
- Stack Bar (2D & 3D)
- Area (2D & 3D)
- Stack Area (2D & 3D)
- High Low (2D & 3D)
- HLCO (2D & 3D)
- % Column (2D & 3D)
- Pie (2D & 3D)
- Bubble (2D)
- Overlay (2D)
- Box (2D)
- Radar (2D)
- Dial (2D)
- Gantt (2D)
- Surface (3D)
- Polar (2D)
- Donut (2D & 3D)

as well as fonts, text, annotation, control lines, control areas, background images. EspressoChart also offers customizable pop-up labels, and mouse events. EspressoChart supports internationalization with locale-specific, and time zone formatting. Advanced features include time-series zooming, data drill-down, parameterized charts, run-time text substitution, and customizable axis labels.

## Advanced Statistical Analysis

In addition to the large array of customization features, EspressoChart offers many advanced statistical analysis features. Users can add

## Highly Customizable

Nearly every chart element can be modified or customized in some fashion. In fact there are over 240 different customizable chart attributes. Users can manipulate color, axis scale, grid step, labels. Legends can be customized

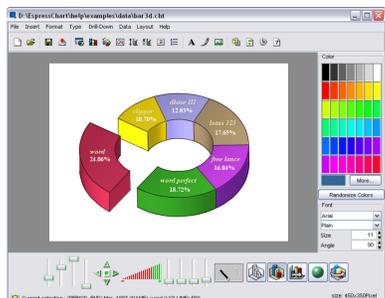
many different types of trend lines to charts including linear, polynomial of any degree, power, exponential, logarithmic, B-Spline curve, as well as normal distribution curves. Other features allow users to create histogram plots, Pareto charts, and box charts. For SPC-type applications, users can add control lines and areas.

## Key Features:

- **Pure Java Architecture:** Deploy on any platform and run with most Java-supported application servers.
- **Full Data Connectivity:** Draw data from databases, XML files, SOAP, Text files, and Java arrays/objects/classes/EJBs.
- **Many Chart Types:** Over 30 different 2D and 3D chart types are supported with many combination/composition charts as well. True 3D rendering allows pan/zoom/rotation and light source adjustments.
- **Visual Designer:** Chart Designer interface allows users to build and customize charts visually. Can be integrated into 3rd party applications.
- **Powerful API:** Deep level Java API integrates charting into applications, applets, servlets and JSPs, and allows full customization at run-time.
- **Flexible Configuration:** Small footprint applet viewers allows dynamic chart interaction within application windows. Run server-side to stream charts as images for Web content.

## Chart Designer

The intuitive Chart Designer is a powerful GUI that allows users



to create even complex charts in a simple point and click environment. Most of EspressoChart's features through the Designer allowing users to go directly from design to deployment. User's can run templates created in the Designer directly or apply their properties and

formatting onto to other charts.

## Full Data Connectivity

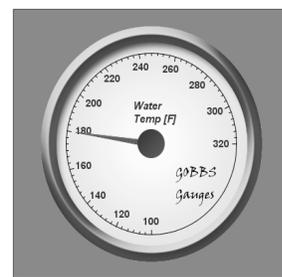
EspressoChart allows users to build charts from virtually any data source. Users can draw data from relational databases, XML files, SOAP, text file, and Java objects/arrays.

EspressoChart provides three different query interfaces, giving users full SQL control. Advanced users can simply write, or import a SQL statement. Intermediate users can employ the integrated

Query Builder which provides a point and click QBE style interface. For users with limited database knowledge, EspressoChart provides the Data View Interface. It insulates users from database complexity, by allowing administrators to pre-define tables, joins, fields, and even filters creating in effect a local schema. End-users can then select (typically aliased) fields, and perform ordering, sorting, filtering, and aggregation without any knowledge of the underlying database structures.

## Gauges

EspressoChart includes dozens of high quality gauge templates. Users can apply the existing templates onto their own data or mix and match background and foreground images to achieve their desired look. Gauge designs are available in circular, square, semi circular, and quarter circle styles. Designers can also create custom images for their gauges. These customized images will automatically scale with the gauge plot taking the guess work out of the customization process.



## Chart API

Chart deployment is generally handled through the powerful Java application programming interface. Charts can be created and run with as little as one line of code. Developers can easily integrate charts into applications, applets, servlets and JSPs. The API can be used to create charts on the fly or to run templates created in the Chart Designer. Charts can also be customized programmatically. With low-level detail the API can be used to access and modify any chart element.

## Many Deployment Options

Charts can be run server-side to be integrated with Web (HTML) content. Dynamic charts can be streamed from the server in many different image formats (GIF, JPEG, PNG, WMF, animated Flash, and SVG), and embedded with existing content. Even exported charts are actionable, supporting hyperlinks, drill-down and pop-up labels. EspressoChart also provides several lightweight applet viewers for showing charts in thick-client/application interfaces. The viewers allow users to interact with and manipulate charts in real time.

Copyright ©2010 Quadbase Systems Inc. All rights reserved. EspressoChart is a registered trademark of Quadbase Systems Inc. All other trademarks belong to their respective owners.

## Sample Chart API Code:

### Example 1

```
import quadbase.ChartAPI.*;
import quadbase.jdbc.*;
import java.awt.*;
import java.applet.*;

// Construct a three-dimensional bar chart using data from a
// database.

public class SampleChart1 extends Applet {

    public void init() {

        setLayout (new BorderLayout());
        ColInfo colinfo = new ColInfo();
        // Use the third data column in the query
        // result set as the value column.
        // Use the second column as the categories.
        // Use the first column as the data series.
        colinfo.value = 2;
        colinfo.category = 1;
        colinfo.series = 0;
        DBInfo dbinfo = new DBInfo ("jdbc:quadbase/
        /machine/schema", "quadbase.jdbc.Qbdriver",
        "usermane", "password", "select * from
        table");

        add ("Center", new QbChart(this,
        QbChart.VIEW3D, QbChart.BAR, dbinfo,
        colinfo));

    }

}
```



**Quadbase Systems Inc.**  
 275 Saratoga Ave., #105  
 Santa Clara, CA 95050  
 Tel: (408) 982-0835  
<http://www.quadbase.com>