

EspressChart[®]

White Paper



株式会社クライム
TEL: 03-3360-9336
FAX: 03-3660-9337
soft@climb.co.jp
www.climb.co.jp

概要

EspressChartRは、様々な種類の2D および3D のチャートをデザインしたり、カスタマイズするための複合ツールです。マウスによる指定とクリックだけで迅速にチャートをデザインおよび編集することができ、Java 開発者は強力なオブジェクト指向API を使用してプログラム中にチャートを組み込むことができます。EspressChart で作成したチャートをウェブページに組み入れて、アプレットによってインタラクティブなレンダリングを行うこともできます。例えば、3D のチャートを移動したり、特定の軸に沿ってサイズ変更を行ったり、任意の角度に視点を変えて表示することができます。チャートを一般的に使用されるJPEG、GIF、PNG、PDF等のフォーマットでサーバ側において生成し、ストリーミングでブラウザに表示させることができます。どちらの場合においても、新しいチャートがサーバから供給される最新のデータによって常に更新されるため、動的なチャートの表示が可能です。使用できるデータソースは、すべてのODBC/JDBC データベース (Excel スプレッドシートを含む)、テキストデータファイル、HTML ファイル、およびXML ファイルのリアルタイム入力またはファンクションコール中のアーギュメントです。XML ファイルを使用することにより、入力データを供給したり、チャートの属性を定義およびアップデートすることができます。EspressChart は、**Chart Server**、**Chart Designer**、**Chart Viewer**、および**Chart API** の4つの強力なウェブベースのチャートツールによって構成されています。

1. Chart Designer

Chart Designer は、データファイルやデータベースへの照会から得られるデータに基づいた様々な種類のチャートを作成するためのツールを提供する、インタラクティブなフロントエンドのアプリケーションです。マウスによる指定とクリックによって、無限のバリエーションを持つチャートを迅速かつ簡単に作成することができます。Chart Designer は100% pure Java によって記述されており、互換性があるすべてのブラウザで実行することができます。

Chart Designer は、チャートを異なる視点から表示したり、任意の角度にチャートを配置することができます。チャートは数種類のバイナリフォーマットで書き出すことができ、静止画像として表示したり、印刷することができます。また、チャートを.tpl フォーマットで保存することにより、数値データを保存せずにチャートの属性(色、サイズなど)のみを保存することができます。テンプレートファイルである.tpl がブラウザに読み込まれると、その都度数値データがデータソースから読み込まれます。このようにすれば、プログラミングをすることなくチャートを継続的にアップデートすることができます。これに加えて、チャートをテンプレートとして保存すれば、外見や雰囲気が一統された複数のチャートを作成することができます。また、チャートは本アプリケーション固有の動的フォーマットである.cht で保存することもできます。.cht フォーマットは、Chart Viewer Java アプレットでチャートを表示することを可能にします。EspressChart は、生成されたチャートをgif、jpeg、png、bitmapおよびPDFなどの標準的なフォーマットで保存および書き出すことが可能です。

チャートタイプ

チャートには主に二次元（2D）と三次元（3D）の2種類の表現方法があります。主要なチャートに二次的な値を表示するオプションもあります。このようなコンビネーションチャートには30種以上のバリエーションがあります。

カラムチャート

バーチャート

XY(Z)分散チャート

ラインチャート

スタックカラムチャート

スタックバーチャート

パイチャート

エリアチャート

スタックエリアチャート

High-Low チャート

HLCO チャート

パーセンテージカラムチャート

サーフェスチャート（2Dのみ）

バブルチャート（2Dのみ）

オーバーレイチャート（2Dのみ）

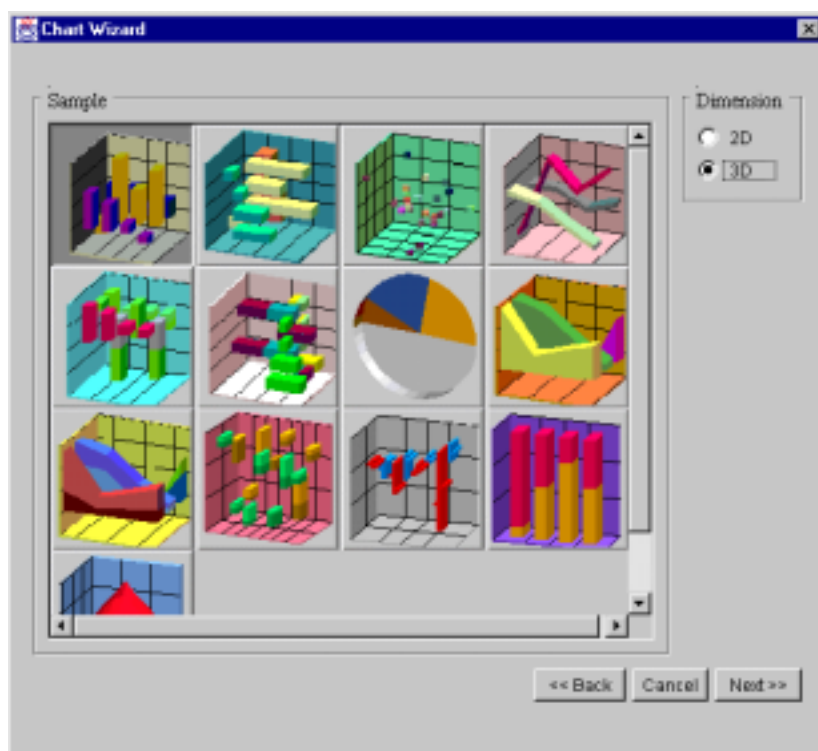
ボックスチャート（2Dのみ）

レーダーチャート（2Dのみ）

ダイアルチャート（2Dのみ）

ガントチャート（2Dのみ）

タイムチャート（2Dのみ）

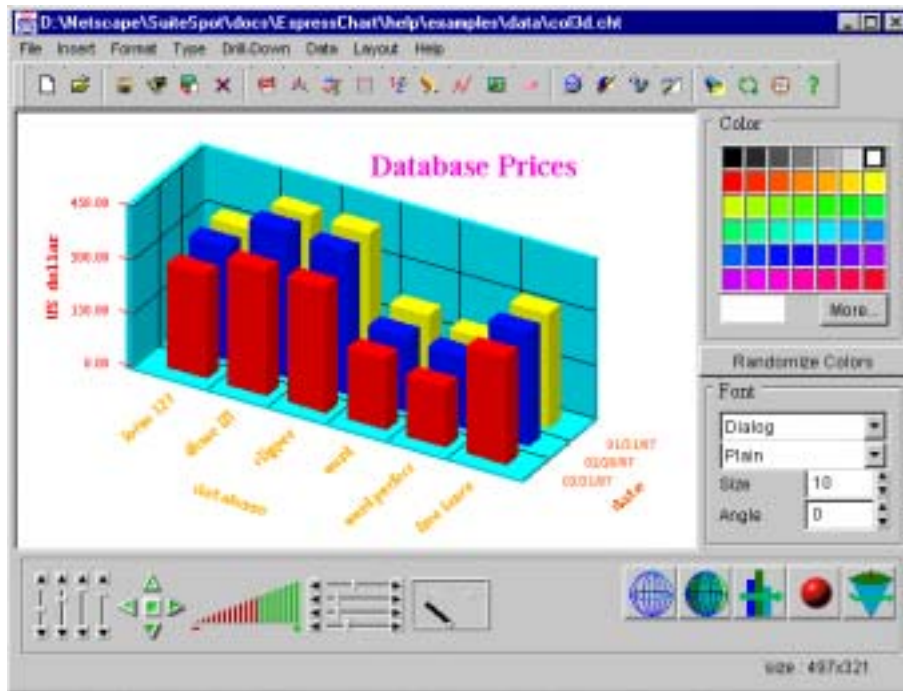


3Dチャート選択ダイアログボックス

Chart Designerダイアログボックス

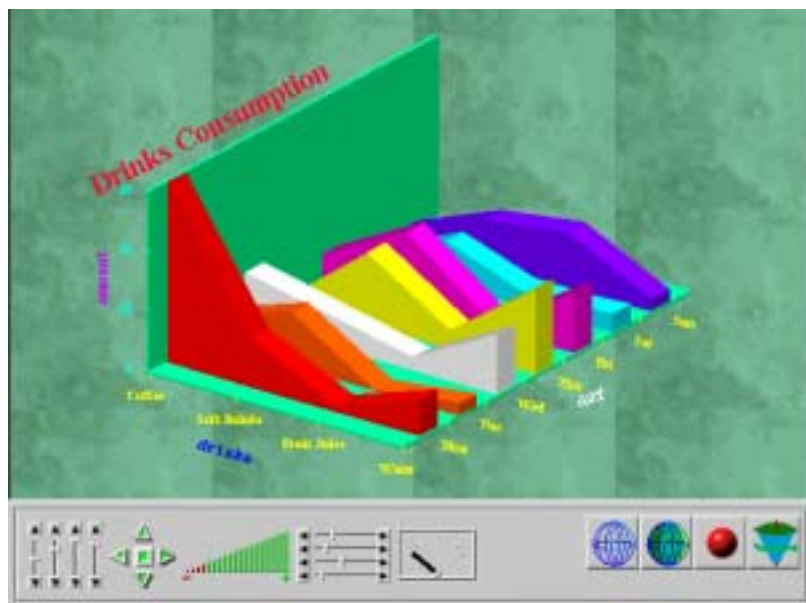
必要に応じてチャートをカスタマイズし、外見や使い方を変更することができます。チャートの各コンポーネントのフォントやカラーを変更したり、各軸のスケールを変更したり、あるいはチャートのサイズ、位置、方向を任意に変更することができます。これらの機能は以下に示される Chart Designer の最終ダイアログボックス（この例では3Dのカラムチャート）で使用すること

ができます。



II. Chart Viewer

Chart Viewer は、チャートを別個に表示するためのアプレットです。これによってチャートを回転、サイズ変更、ズーム、および移動表示させることが可能になります。個別のデータポイントをクリックすることによってそのデータポイントに関連するデータを復元したり、他のチャートにリンクしたりすることができます。



ChartViewer 画面サンプル

III. Chart Server

Chart Server はEspressChart の“バックエンド”で、データベースにアクセスしたり、照会を行うための機能を提供します。このアプリケーションはChart Designer と、ローカルサーバやリモートネットワーク、インターネット上のデータベースとをシームレスに連携させることができます。また、Chart Server はデータのバッファリング、およびローカルソースやリモートソースからのデータの読み込みやそれらへのデータの保存を行うことができます。

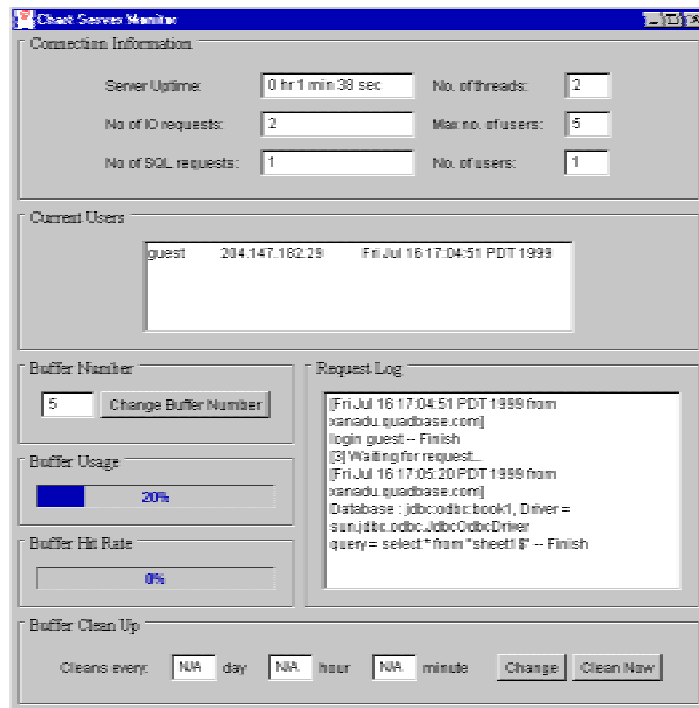


Chart Server モニタ

IV. Chart API

Chart API は100% pure Java によって記述されており、洗練されかつ簡単に使用できるJava パッケージによって高度な2D および3D のチャートライブラリ機能を使用することができます。高機能な3D グラフィックアルゴリズムを採用し、リアルタイムにチャートの表示や操作（移動、ズーム、XYZ 軸のスケール、回転、変換など）を行うことができます。チャートを作図するために、たった1 行のコードを使用するだけで済む場合もあります。Chart API はODBC-JDBC 接続をサポートし、すべてのデータベースから直接データを検索することができます。Chart API はアプレットとしても、またサーブレットやJSP（JavaServer Pages）アプリケーションとしても使用することができます。

EspressChart のアーキテクチャ

既に述べました通り、EspressChart はChart Designer、Chart Viewer、Chart Server、およびChart API の4つのコンポーネントで構成されています。詳しくは下図を参照してください。

EspressChart のアーキテクチャ

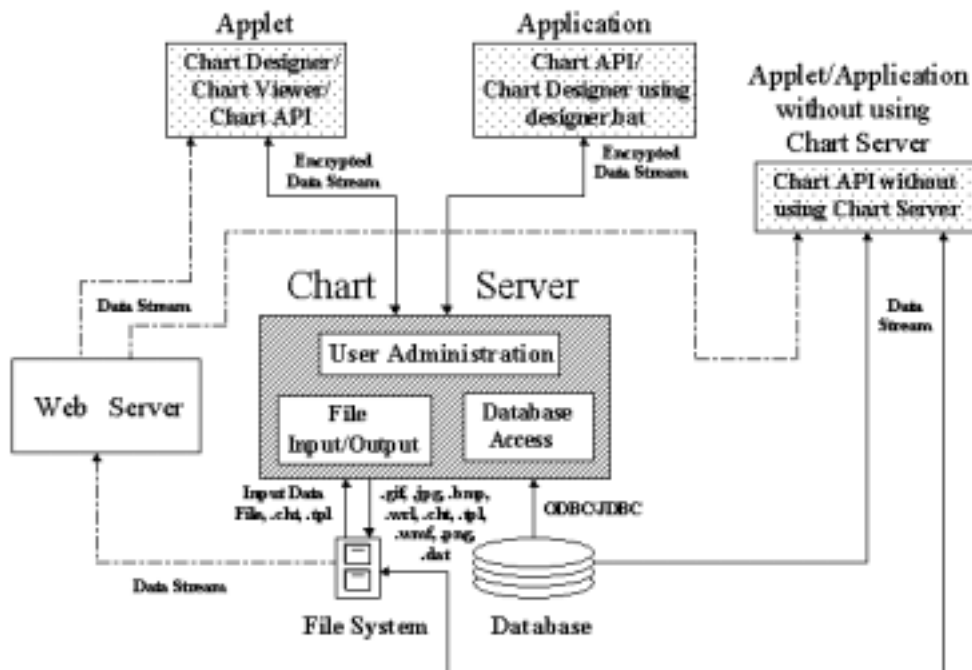
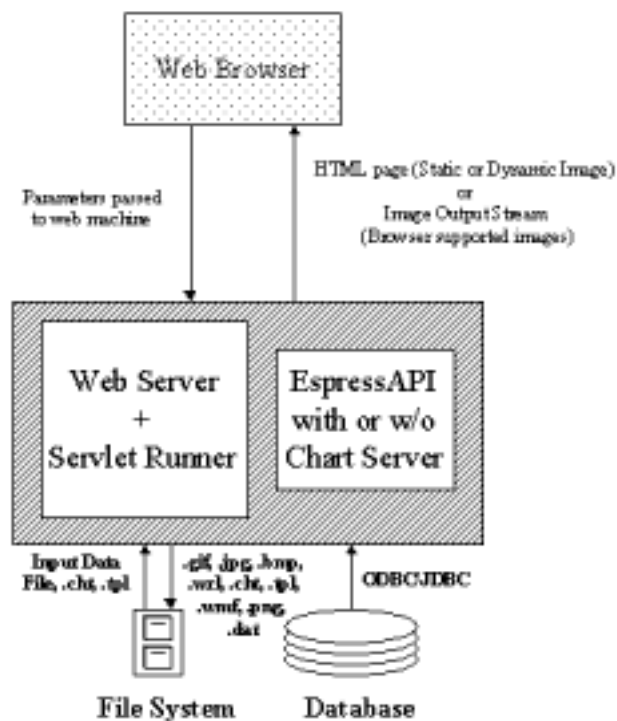


Chart Server はEspressChart にバックエンドを供給します。通常はウェブサーバの機能を持つマシンに置かれ、プロセスまたはサービスとして実行されます。ウェブサーバによって、インターネットやイントラネットを通じてユーザがチャートをデザインしたり表示させたりすることができます。Chart Server は、Chart Designer アプレットのユーザがファイアウォール内に保存されたデータにアクセスすることを許可します。Chart Server の主な機能は、ユーザ認証、ファイルの入出力、データのバッファリングおよびデータベースへのアクセス（データがデータベースから入手される場合）です。Chart Server は複数のクライアントが同時にログオンしてチャートをデザインすることを可能にします。Chart Server はクライアントがファイルシステムから入力されたデータやチャートファイルを利用することを可能にすると共に、クライアントのデータを受け取ってチャートファイルへの書き込みを行います。データベースへアクセスする場合、Chart Server はJDBC ドライバを読み込むか、またはJDBC-ODBCブリッジを使用してODBCデータソースに接続することが可能です。さらに、Chart Server はチャート画像を書き出すための複数のネイティブライブラリを持っています。

Chart Designer またはChart Viewer アプレット、あるいはChart API を使用するプログラムが Chart Server のクライアントになることができます。入力されたデータファイルやデータベース、必要なユーザID やパスワード、読み込まれるチャートファイルなどのすべてのデータソース情報は、Chart Server に引き渡されます。Chart Server はこれらの情報を利用してデータを取得し、暗号化されたデータストリームとしてクライアントに送ります。ファイルのロケーションがURL として入手可能な場合には、クライアントはウェブサーバを使用して直接データにアクセスすることも可能です。これらのデータは解析され、クライアントが実際のチャートを作成します。すべてのEspressChart のクライアントがこの機能を持っています。完成したチャートを保存する場合、クライアントは完成したチャートと共にロケーション情報(ディレクトリやファイル名など)を暗号化されたデータストリームとしてChart Server に引き渡し、Chart Server はそれらをユーザが指定したロケーションに書き込みます。

Chart API を使用すれば、アプリケーションはChart Server を使用することなく直接データベース/ファイルシステムディレクトリにアクセスすることができます。この場合は直接データにアクセスすることが可能なため、より円滑に完成したチャートを入手することができます。

サーブレット環境における EspressChart のアーキテクチャ



サーブレット環境においてEspressChart を使用すると、サーバ側でチャートを生成したり、クライアント側にチャートをプッシュすることができます。この場合、サーバはウェブサーバとサーブレットランナーを実行し、さらにChart API も保有しています。Chart Server を使用する

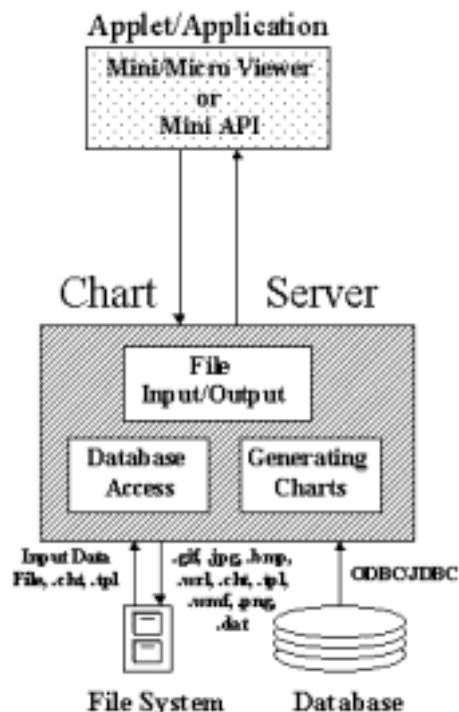
こともできますが、アクセスしているデータがすべてサーバ上にある場合にはその必要はありません。情報がブラウザから引き渡されると、チャートがサーバで生成され、静止画のチャートやインタラクティブなチャートを含むHTML ページ、またはストリーミングによる画像出力としてブラウザにプッシュされます。

サーブレットはチャートを生成し、静止画像を作成し(チャートの書き出し)、さらに静止画像を含んだページを表示するために使用されます。あるいは、サーブレットはクライアント側に直接画像のストリーミングを行うために使用することもできます。この方法を使用すれば、書き出された画像がサーバ側に保存される必要がなくなります。

サーブレットは、チャートを .cht ファイルに書き出すために利用することもできます。しかし、このフォーマットを使用する場合にはChart Server が必要になります。また、インタラクティブなチャートを表示するページ(.chtファイル)はアプレットを使用するため、クライアントがJava を使用可能なブラウザを使用している必要があります。

アプリケーションサーバ (WebSphere やWebLogic など) をウェブサーバおよびサーブレットランナーの代わりに使用することができます。

Mini/Micro ViewerまたはMini API を使用したEspressChart のアーキテクチャ



先に述べました通り、EspressChart はアプレットでも、サーブレットやJSP でも、JavaBeanとしても使用することができます。アプレットは最も高いインタラクティブ性を可能にし、チャー

トがサーバ側で画像として生成されるために大きいアプレットをダウンロードする必要がなくなります。EspressChart は、チャートをサーバ上で生成するという画期的なクライアント/サーバアーキテクチャを可能にします。しかし、チャートをビットマップ画像として書き出すかわりに、ベクトル情報がブラウザにストリーミングされ、チャート画像が即座に描画されるのです。この機能をサポートするために、EspressChart はMini Viewer およびMicro Viewer と呼ばれる、2 つのChart Viewer の縮小版を含んでいます。また、Chart API の縮小版も同様に含まれています。実質的には、これらは標準のChart Viewer およびChart API のサブセットで、サイズ的にそれらよりもずっと小さなものです。これらのフォームはより迅速なダウンロードを可能にし、ベクトル画像をより良く表示しつつ、ユーザによるある程度のインタラクティブな操作を可能にします。

Mini/Micro Viewer およびMini API を使用する場合は、Chart Server が必要です。サーバはチャートを生成し、クライアント側にプッシュします。チャートに何らかの変更が加えられると(チャートタイプの変更、大きさの変更など)、クライアント側に変更がプッシュされる前にサーバ側で変更が行われます。

Copyright © Quadbase Systems, Inc. & 株式会社クライム All rights reserved. Feb. 2002

EspressReport, Quadbase Systems, Inc., and the Quadbase logo are trademarks of Quadbase Systems, Inc. Java, JDK, AWT, Swing, and JDBC are trademarks of Sun Microsystems, Inc. in the United States and other countries.

All other trademarks belong to their respective owners.

株式会社クライム

本社 〒103-0014 東京都中央区日本橋筋御町1-25-4 日本橋栄ビル4F