

EspressReport

Overview, Technology & Features
White Paper



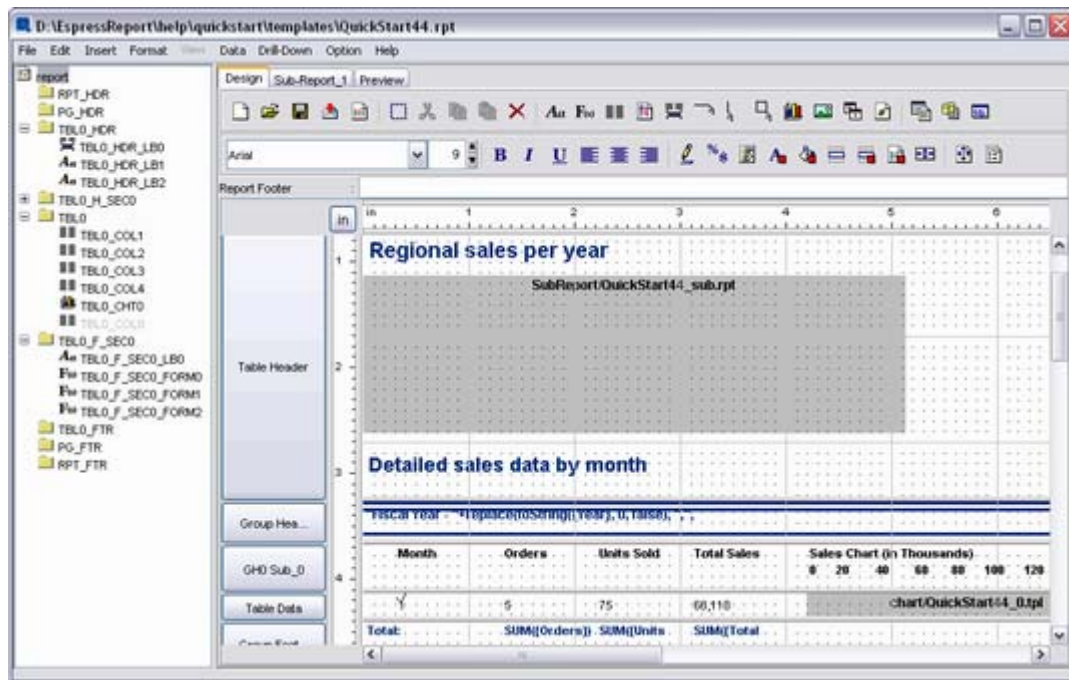
株式会社クライム
TEL: 03-3360-9336
FAX: 03-3660-9337
soft@climb.co.jp
www.climb.co.jp

I. オーバービュー

EspressReport は 100%Java で構築されたレポートングツールです。ユーザは各種プラットフォームで高度なレポートを簡単に導入、開発できます。レポートング機能をサブレット、JSP、アプリケーションに容易に組込むことができる堅牢な API と同様に、ビジュアルなデザイン環境を持っています。多くのアドバンスな機能によりユーザは強力なデータプレゼンテーションの作成、各種フォーマットで配信することができます。

II. レポート デザイン

EspressReport のユーザ インターフェイス Report Designer でレポートを開発、デザインすることができます。イージーツーユースのグラフィック ユーザ インターフェイスでレポートをドラッグアンドドロップスタイルでカスタマイズできます。旧来のレポートツールを使用したことがあるユーザは Report Designer のバンドスタイルのインターフェイスがすぐに理解できます。他の多くの伝統的なレポートング ツールに似たインターフェイスで、デザインプロセスをすぐにスタートできます。



Report Designer は旧来のレポートングツールに似ていますが、さらに Web ベースのレポートング用にデザインされています。また、純粋な Java コンポーネントであるので、どんなプラットフォームで稼動することもできます。そして Web サーバにインストールでき、クライアントにソフトウェアをインストールすることなく、Web ブラウザから直接稼動させることができます。

コード・サンプル：Report Designer の起動

```
import java.awt.*;
import javax.swing.*;
import java.io.*;
import quadbase.ReportDesigner.designer.*;

public class ReportDesigner {
```

```

public static void main(java.lang.String[] args) {
    try {
        Report Designer doReport = new ReportDesigner();
    } catch (Exception ex) {
        ex.printStackTrace();
    }
}

public Report Designer() {
    // Begin Code : Start Designer in default mode and show the Designer
    QbReport Designer designer = new QbReport Designer(null);
    designer.setVisible(true);
    // End Code : Start Designer in default mode and show the Designer
}
}

```

A. レポート タイプ

EspressReport はレポートデザインのフレームワークとして 5 つの基本的なタイプを使用します。各レポートタイプは特別なデータマッピング オプションを持ち、直感的なウィザードを使用して実行します。最初のレポートにデータが多くなった時、ユーザはレポートのほとんどすべてのエレメントをカスタマイズできます。

シンプルな段組によるレポート (Simple Columnar Report) : これはすべてのレポートタイプでもっともベーシックなものです。グループ化やブレイクがないシングルテーブルでのデータです。

概要説明レポート (Summary Break Report): シンプルな段組によるレポートのように、概要説明レポートは縦式データをとったり、表形式で表されたりします。しかし、ユーザはセクションでデータをブレイクしたり、サマリーのフィールドを挿入したりできます。

クロスタブ レポート: クロスタブ レポートはミックス型のフォームで縦データの表示、要約するレポートフォーマットです。クロスタブ レポートはスプレッドシートに似ています。列と縦列は要約され、複数次元データは 2 次元データに表示が可能になります。

総合及び詳細レポート: 総合及び詳細レポートはマスターフィールドによるグループ化された表データのセットです。マスターフィールドは通常はデータフィールドにデータと 1 対複数の関連をもっています。

メール・ラベルのレイアウト : メール・ラベル用のレイアウト出力

各レポートタイプはデータがマップされているレイアウトのみを対象とします。データがマッピング後、それぞれのレポートエレメントはカスタマイズが可能で、ユーザはそれぞれのレポートタイプのレポートを複数作成可能です。

B. Report Designer の使用

EspressReport はレポート デザインに対して斬新なアプローチを取っています。EspressReport はバンデッド スタイルのインターフェイスを使用し、デザイン ウィンドウのそれぞれのバンドはレポートのセクションを意味します。オブジェクトはフリーフォームで

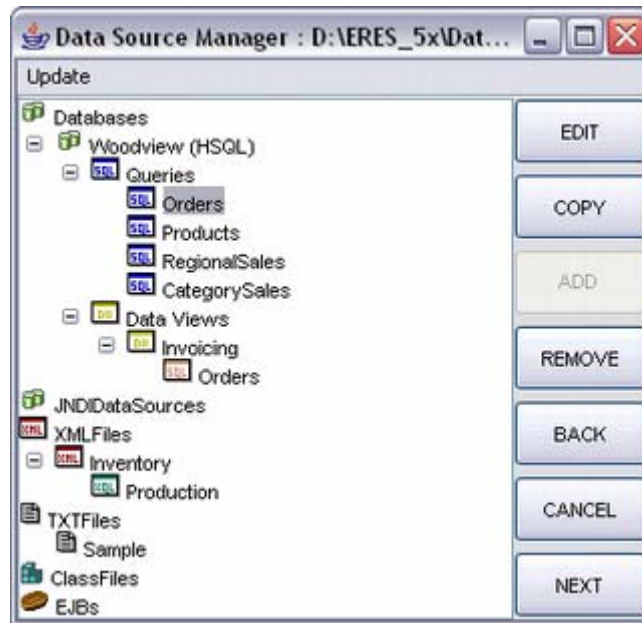
各セクションに挿入、操作ができます。EspressReport はまたオブジェクトのグループを一括で移動、フォーマットできるユニークな操作機能を持っています。この機能により縦フォーマットでのオブジェクトの操作を同様に行うことができます。

ユーザはオブジェクト フォーマットをグローバルに操作することができます。この機能を使ってレポートワイドなレベルで、レポート オブジェクトの各タイプ（レーベル、フォーミュラ、コロン フィールド等）のオブジェクト プロパティをアサインできます。これによりオブジェクトの指定されたプロパティを変更でき、オブジェクトのデフォルト属性を変更できます。グローバルなフォーマットは XML ファイルにエクスポートしたり、レポート間を行き来したりできます。

このユニークなデザイン機能で、莫大な時間を割くことなく、プロのようなレポート デザインが可能で。

C. データ・ソース

EspressReport はすでに組み込まれたデータソース マネジャーとクエリビルダーでデータのハンドリングができるユニークなシステムを提供します。Report Designer は JDBC/ODBC、JNDI、SOAP ソース、XML ファイル、テキスト ファイルから直接データを呼び込みます。アプリケーション・オブジェクト/アレー・データは API または Java クラスと EJB 経由で抽出することができます。データソースマネジャーによりレポート作成に使われるすべてのデータソースを管理することができます。

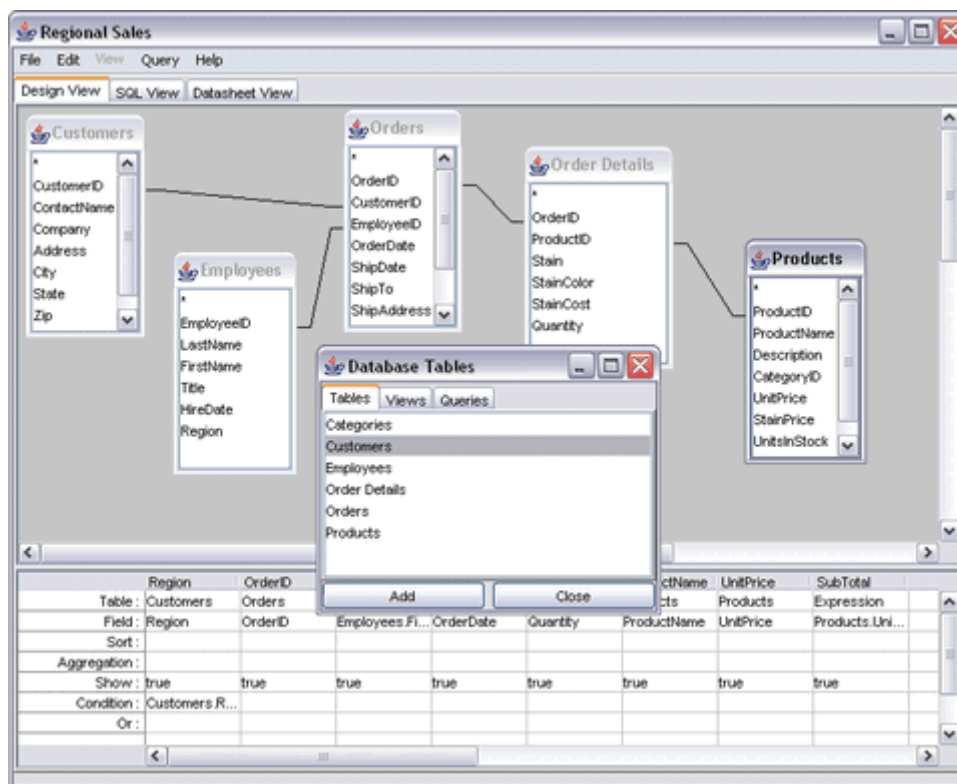


データソースマネジャー ウィンドウは階層ファイルシステムのようになっていて、テキスト、XML ファイルのデータベース接続情報、クエリ、ファイル位置をストアできます。データソース レジストリは Report Designer 外での変更を可能にして、XML フォーマットでストアされます。

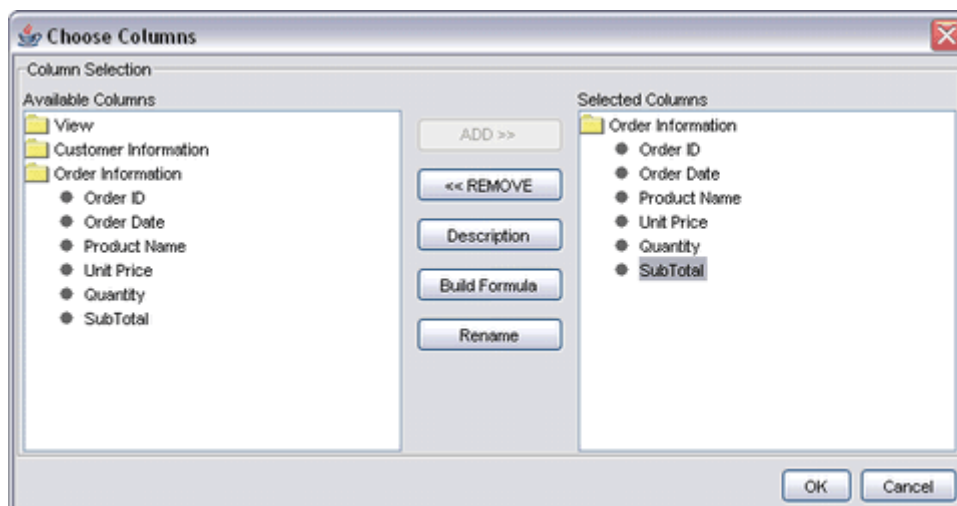
1. 完全な SQL コントロール

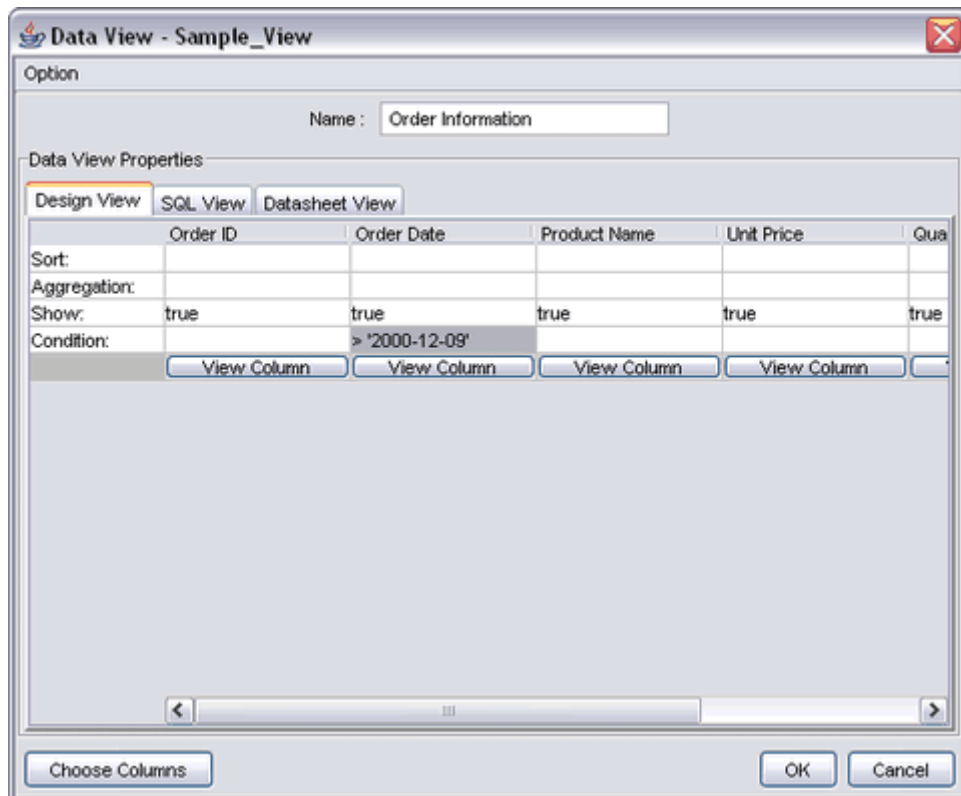
EspressReport はレポート・データを抽出できる SQL の完全なコントロールをユーザに与えています。さらにデータベースをクエリするための 3 つの追加機能があります。さらにユーザは直接 SQL 記述を入力したり、インポートしたりすることができます。通常は EspressReport のグラフィックなクエリ・ビルダを利用することができます。このクエリ・

ビルダは QBE スタイルのウィンドでデザインされ、クエリデザインに関して高い柔軟性を持っています。クエリはセーブ、変更ができ、複数のレポートが同じクエリから作成できます。クエリビルダーでデータベース式の作成、ユーザ自身のデータフィールドの構築が可能です。



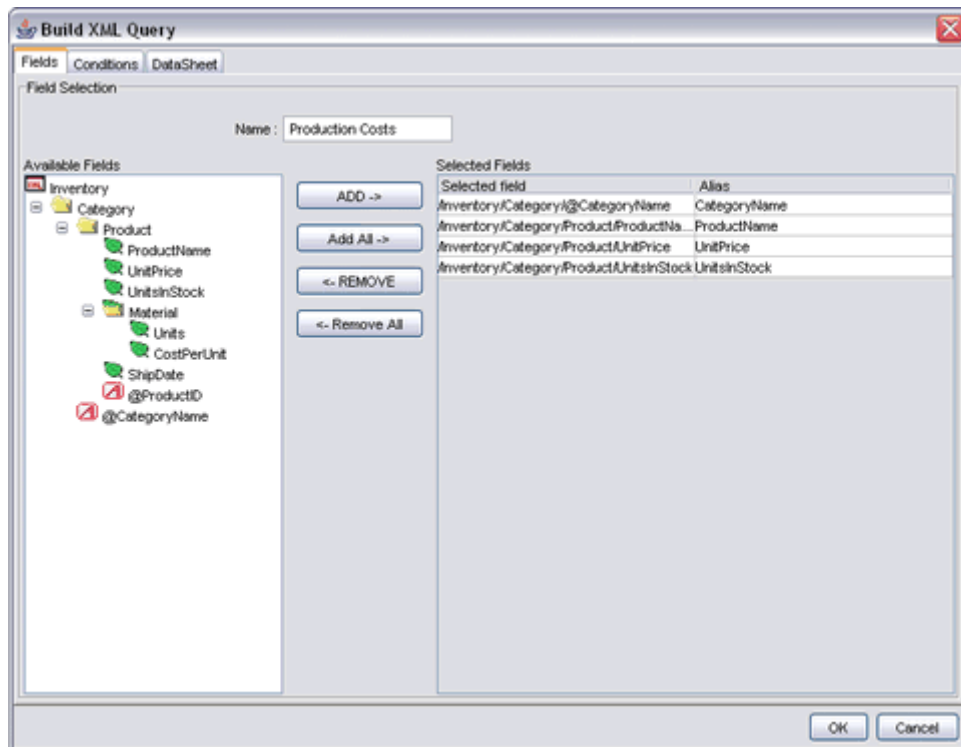
データベースの基礎に関する経験が不足しているユーザは EspressReport のデータビュー・インターフェイスを活用することができます。管理者はユーザが選択できるように予めテーブル、ジョイン、フィールド、フィルタを定義します。ユーザはデータベースの構造の知識が無くてもフィールドの選択、結果のフィルタリングを行うことができます。





2. 拡張 XML サポート

多くのアプリケーション・モデルはアプリケーション・コンポーネント間のデータエンコードと転送に関して XML を活用しています。このタイプのデータをレポートできるように EspressReport は XML データの定義とクエリができるインターフェイスを実装しています。DTD ファイルをベースに、ユーザは XML ファイルからクエリや、サーブレット/JSP から XML アウトプット・ストリームを生成することができます。



3. Java オブジェクト/アレー・データ

リレーショナル・データベースや XML ソースからデータを直接抽出する以外に EspressoReport は Java オブジェクトやアレー、または EJB 経由で直接データをレポートに渡すことができます。ランタイムでデータを直接レポートに渡したり、デザイン・タイムで Java クラスファイルからオブジェクト/アレー・データを引き出したりすることができます。EspressoReport はアプリケーション・データを EspressoReport にインターフェイスする各種の方法を提供します。

コード・サンプル：レポートに簡単なデータを引き渡すクラス

```
package help.examples.DataSources.classes;

import java.awt.*;
import quadbase.Report Designer.ReportAPI.*;
import quadbase.Report Designer.util.*;

public class SampleData implements IDataSource {

    // Setting DbData for passing data as arguments
    String dataType[] = {"string", "string", "double", "double"};
    String fieldName[] = {"OrderID", "Product", "Price", "Quantity"};
    String records[][] = {{"1001", "Chair", "325.00", "4"},
        {"1001", "Table", "1211.00", "1"},
        {"1020", "Dresser", "2214.00", "1"},
        {"1020", "Table", "1211.00", "1"},
        {"1020", "Cabinet", "4021.00", "2"},
        {"1031", "Chair", "325.00", "2"},
        {"1044", "Table", "1211.00", "1"},
        {"1044", "Dresser", "2214.00", "1"}};

    DbData data = new DbData(dataType, fieldName, records);

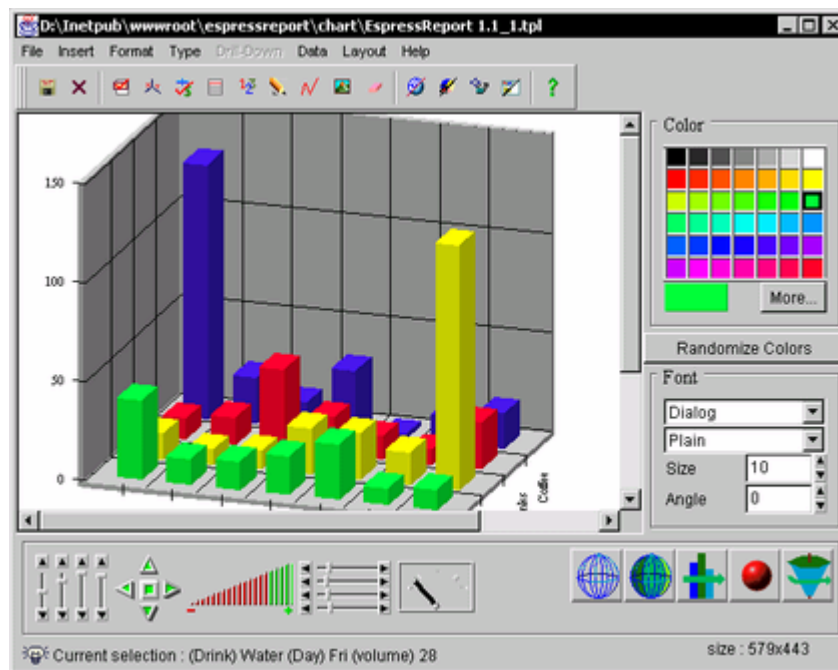
    // create a empty constructor
    public SampleData() {};

    // implement getResultSet
    public IResultSet getResultSet() { return data; }

}
```

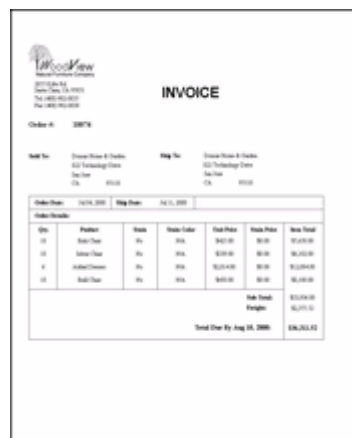

D. ビジュアル データの表現

EspressReport はビジュアルなデータ表示に対して多くの違った手法を提供します。組み込まれたチャートデザイン ユティリティーで30種以上の2次元, 3次元のチャートを選択でき、レポートデータにプロットできます。EspressReport はチャートを3D レンダリング、パン/ズーム、回転、光源変更をしながら描写できます。チャートのテンプレートはレポートとは別にサブされ、チャートの再デザインの必要なく、何度でも必要な時に再使用できます。



E. 結論

EspressReport は Java レポートング ツールではもっとも拡張性のあるデザイン機能を持つツールセットです。ヘビー ユーザーも開発者も、時間と労力なく、EspressReport デザイン ツールを使用してプロの高い品質のレポートを作成できます。



III. レポートの稼動

EspressReport はレポートを稼働させる上で多くの異なったオプションを持っています。ユーザはクライアントにアプレットをダウンロードしたり、Report API を使用してサーブレット、JSP、アプリケーションにレポート機能を組み込ませたりして、レポートを展開できます。

A. 出力フォーマット

EspressReport は各種色々なフォーマットでレポートの生成が可能です。ベーシックなレポートはどんなブラウザにもコンパチブルな標準 HTML でのレポート作成です。さらに高度なデザインでは、さらに複雑なフォーマット要求が処理できる DHTML スタイル シートのレポート作成です。レポートのプリント、番号付き出力用として EspressReport は高品質な PDF 出力を生成します。また、すべてのレポートデータを含んだコマ挿入型のテキスト ファイルを生成できます。MS Office ユーザには Excell スプレッドシートと RFT(MSWord)用を準備しています。

B. アプレットを使用して

EspressReport はレポートとチャートの表示と使用に関していくつかのアプレット・ビューアを提供しています。

1. Report Viewer

Report Viewer はアプレット コンポーネントを使用した Web ブラウザ経由でのクライアント・メカニズムです。Report Viewer は2つのバージョンがあり、AWT と Swing バージョンです。AWT バージョンではダウンロード、プラグインが無くても Report Viewer を使うことができます。アプレットはどんな EspressReport のテンプレートをビューでき、レポートは直接 Web ページ同様に他のレポートにもハイパーリンクできます。Report Designer から直接組み込まれたアプレットで HTML ページを作成できます。カスタマイズしたポップアップメニューを使用して、ページとリンクしたレポートを指示することができます。

2. PageViewer

大規模なレポートではレポート全体をクライアントにローディングすることは意味がありません。PageViewer は Report Viewer と同じ機能ですが、リクエストされた時のみクライアントにレポート・ページのみ転送するページ・サービス技術です。PageViewe はクライアントでメモリを抑えて、大規模なレポートのローディング、プレビューを可能とします。

C. API の使用

EspressReport に組み込まれた API はさらなる柔軟性を提供します。Report API を使用してサーブレット、JSP、アプリケーションにレポート機能を組み入れることが可能です。Report API でユーザ パラメータをベースにレポートをすばやく作成できます。そしてクライアントのブラウザに直接ストリームしたり、サーバ上に作成したレポートをキャッシュしたりすることができます。API を使用してほとんどのレポート オブジェクトをコントロールでき、レポートは単純にプログラム化してデザインができます。また API は実行時にレポートのプロパティを変更している間、Report Designer で構築されたテンプレートを稼働できます。

コード・サンプル：サーブレットを使用してクライアントへレポートをオープン、ストリーズ

```
import java.awt.*;
import java.applet.*;
import java.io.*;
```

```

import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
import quadbase.Report Designer.ReportAPI.*;
import quadbase.Report Designer.ReportElements.*;
import quadbase.Report Designer.ReportViewer.*;
import quadbase.Report Designer.util.*;
import quadbase.Report Designer.lang.*;

public class SampleServlet extends HttpServlet implements SingleThreadModel {

    public void doGet(HttpServletRequest req, HttpServletResponse res) throws
ServletException, IOException {

        // Set the "content type" header of the response
        res.setContentType("text/html");

        // Get the response's OutputStream to return content to the client.
        OutputStream toClient = res.getOutputStream();

        try {
            // Open up specified Report
            QbReport report = new QbReport((Applet)null,
"templates/SampleReport.rpt");

            // Export (Stream) the report to DHTML
            report.export(QbReport.DHTML, toClient)
        } catch(Exception e) {
            e.printStackTrace();
        }

        // Flush and close the outputStream
        toClient.flush();
        toClient.close();
    }

    public String getServletInfo() {
        return "SampleServlet servlet for EspressoReport";
    }
}

```

D. サーバ・サイド機能

EspressoReport はユーザが短期間でベストな方法でレポート機能を導入できるようにいくつかのキーのサーバ・サイド機能を提供します。

1. パラメータ化したレポート

パラメータ化したレポートは EspressoReport のユニークな機能の 1 つで、レポート・パラメータを含む HTML フォームで可能です。この機能を使用して、開発者は単純にレポートをオープンし、HTML フォームを呼び出し、レポート・テンプレートからパラメータ情報を取得して、ユーザからパラメータ値を取得する独自のフォームを構築できます。この機能を使用して生成された HTML フォームは選択されたフォーマットでクライアントにレポートをストリームするサーブレットを呼び出すことができます。

2. ドリルダウン

ドリルダウンはトップのレポート/チャートから下位の情報を引き出すドリル的な手法です。EspressoReport はこの機能をサポートし、レポートとチャートが簡単に展開できるようにいくつかの手法を提供します。この EspressoReport のドリルダウン手法を利用することによりユーザは書くドリルダウンのレベルで 1 つのレポート・テンプレートを利用するだけすみません。レイヤー間のリンクは自動で処理され、関連するサーブレットに展開されます。この機能を使用することにより高機能なレポート作成が可能です。

E. Scheduler (スケジューラ)

EspressReport にはユーザがレポートとプロセスをスケジュールできるスケジューリング・アプリケーション (Scheduler) を含んでいます。そのインターフェイスを使用してレポート・テンプレートか、コマンドラインで時間か、時間間隔でスケジュールすることができます。レポートはファイルシステムにエクスポートか、電子メールで転送することができます。スケジュールは別々のパラメータ値とセキュリティレベルで設定可能です。API を使用してユーザがスケジュール・ジョブをプログラムでアクセス・変更することもできます。

コード・サンプル：スケジュールしたジョブの作成

```
import java.io.*;
import java.util.*;
import quadbase.scheduler.*;
import quadbase.Report Designer.util.IExportConstants;

public class MySchedule {

    public static void main(java.lang.String[] args) {

        try {
            MySchedule ScheduleReport = new MySchedule();
            ScheduleReport.start();
        } catch (Exception ex) {
            ex.printStackTrace();
        }
    }

    public void start() throws Exception {

        // Create a schedule object to run a report template
        ScheduleObject sObj = new ScheduleObject("SCH_OBJ1",
ScheduleObject.REPORTOBJ);

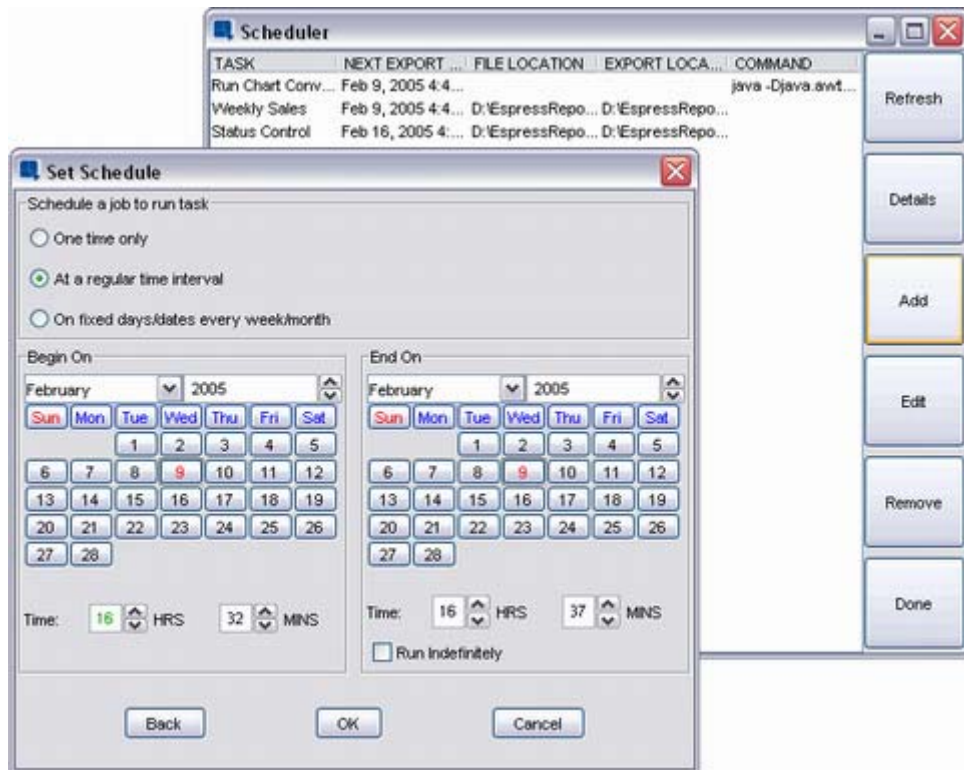
        // Specify which template to use
        sObj.setFileLocation("Templates/SAMPLE.rpt");

        // Specify the export format
        sObj.setReportType(IExportConstants.DHTML);

        // Specify the periodicity of the schedule (here run once) and start
time
        sObj.setTaskOption(ScheduleObject.ONE_TIME);
        Calendar calendar = Calendar.getInstance();
        calendar.add(Calendar.MINUTE, 5);
        sObj.setStartDate(calendar.getTimeInMillis());

        // Don't send email notification
        sObj.setSendEmail(false);

        // Add new schedule to jobs list
        ScheduleModifier.addScheduleTask(sObj);
    }
}
```



IV. テクノロジーとアーキテクチャ

EspressReport は 100%Java ツールで、ほとんどすべてのプラットフォームで稼動する柔軟性があります。JDK1.2.2 同等か以上の JMV で稼動します。EspressReport はほとんどのアプリケーション サーバとサーブレット ランナーとコンパチブルです。JDBC データベース、ODBC データベースと JDBC-ODBC ブリッジ経由で直接接続できます。

A. EspressReport コンポーネンツ

EspressReport は Report Designer, Report API, Report Viewer, EspressManager の 4 つのコンポーネンツで構成されています。

Report Designer: Report Designer は EspressReport のビジュアルなデザイン環境を構成します。スタンドアローン アプリケーションとして稼動することも、Web ブラウザを経由して、サーバ上で直接デザインすることも可能です。イージーユーザーのインターフェイスにより初心者でも簡単に高度なレポートを作成することができます。さらに他の 2 つのインターフェイスでそれぞれクエリ、チャートのデザインが可能です。

Report API: Report API はサーバ サイド、クライアント サイドに関係なく、アプリケーション、サーブレット、JSP にレポート機能を組み込ませるためのイージーユーザーのアプリケーション プログラミング インターフェイスです。その 100%Java により、ほとんどのプラットフォームで変更なしに稼動します。API を使用してレポートのどの部分もカスタマイズ化が可能で、それによりユーザは実行時のレポートのフォーマティングをフルにコントロールできます。レポートは 1 行か 2 行のコードで作成が可能です。

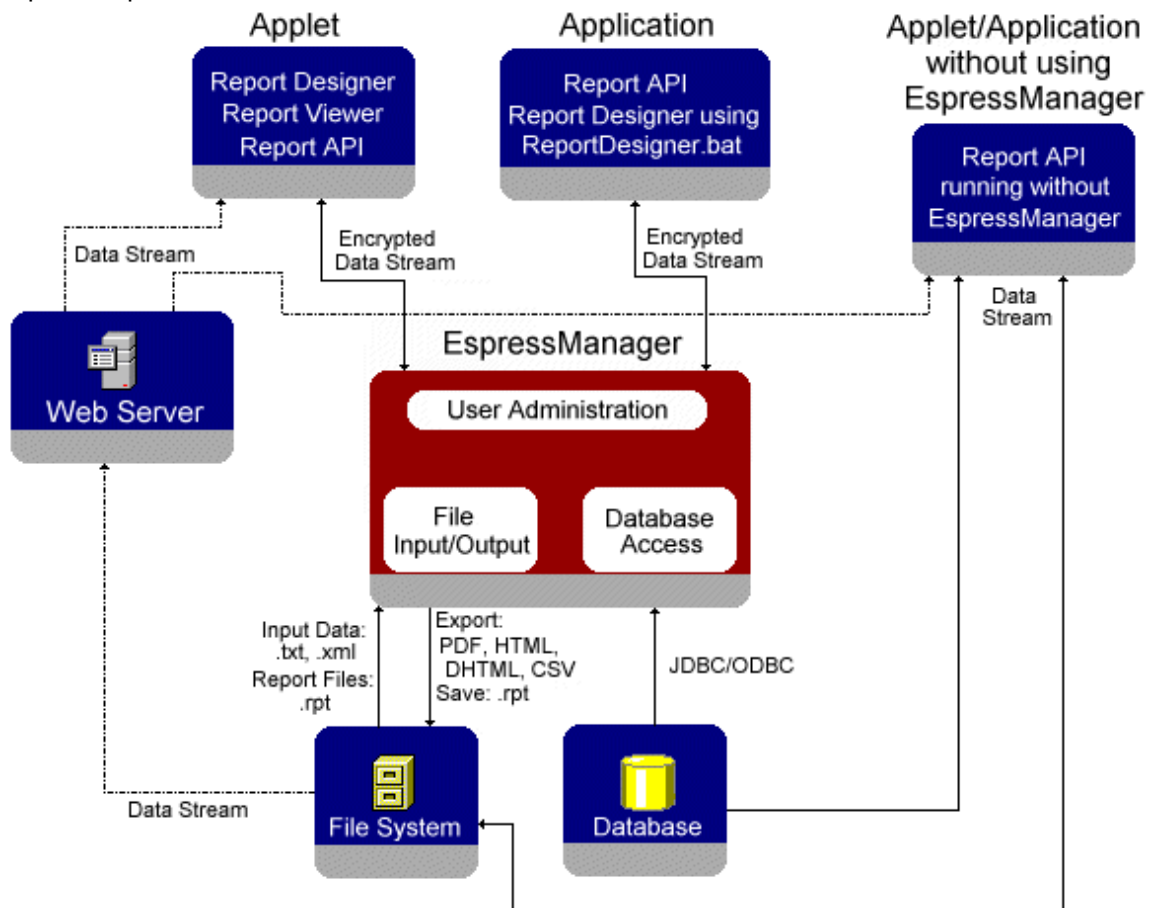
Report Viewer: Report Viewer はインテグレートされたアプレットで、レポートのビューと相互作用が可能です。アプレットはページ番号付きのフォーマットでレポー

トを見せ、ユーザはポップアップメニューでレポート関連を指示できます。レポートのテンプレートはアプレットを使用して直接お互いをハイパーリンクできます。EspressReport はコーディングの必要が無く、組み込まれたアプレットで HTML ページを作成できます。

EspressManager: EspressManager は EspressReport のバックエンドを提供します。主なファンクションはユーザ アドミシトレーション、ファイルの入出力、データのバッファリング、そしてデータベースのアクセスを構成します。クエリビルダーとローカルなサーバとネットワーク及びインターネット上のリモートでの使用可能な幅広いデータベース間でのシームレスな接続を提供します。EspressManager は Report Designer を使用するために稼動する必要がありますが、Report API 使用時は必要ありません。

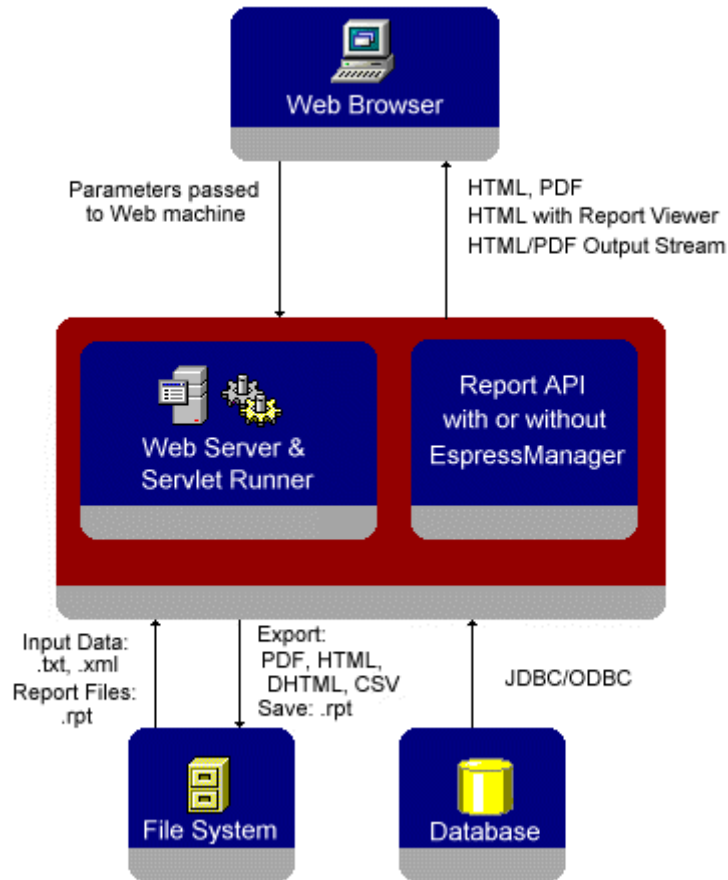
B. EspressReport の構成

EspressReport はいくつかの違った構成で稼動できます。次の図は Web 環境で各種の EspressReport コンポーネントが稼動していることを示しています。



EspressManager は通常 EspressReport の「バックエンド」機能を実行します。サーバマシンにインストールした時は、ソフトウェアのインストールをしなくても、Web ブラウザ経由で直接 Report Designer をロードすることができます。Report Designer はまた EspressManager として同じマシンのスタンドアローンアプリケーションとして稼動することができます。他の EspressManager クライアントは Report Viewer や Report API を使用してアプリケーション、サブレット、JSP になることができます。

EspressReport はまたサーバ サイドにレポートを作成したり、クライアント サイドにレポートを送ったりできます。



この構成では、サーバは Web サーバ、サーブレットランナー、同様に Report API を稼働させています。EspressManager は使用できませんが、もしアクセスされるデータがすべてサーバマシンにあれば、必要ありません。情報はブラウザから渡され、サーバ上でレポートは作成され、HTML ページや、PDF ファイル、the Report Viewer アプレットを含む HTML ページとしてブラウザに押し付けられます。また出力ストリームとして HTML と PDF を作成します。

サーブレットはレポートの生成、HTML ページと PDF ファイルの作成に使用され、HTML/PDF を含むページを表示します。サーブレットはまた HTML または PDF としてクライアントサイドに直接レポートをストリームすることができます。この手法の利点はエクスポートしたレポートがサーバサイドにセーブする必要のないことにあります。

サーブレットはまた.rpt ファイルにレポートをエクスポートできます。しかし、このフォーマットを使用するときは EspressManager が必要となります。またクライアントはページがアプレットである Report Viewer を使用してレポート(.rpt ファイル)を表示する時、Java のブラウザ機能を使用する必要があります。JSP もまたアプレット同様に使用することができます。

WebSphere や WebLogic のようなアプリケーションサーバは web サーバやサーブレットランナーの代わりに使用できます。

V. 結論

EspressReport はレポートリング マーケットが必要とするニーズの変化を掴み取った革新的な製品です。単にパワー ユーザがレポートの作成、開発ができるだけでなく、ユーザが提供するパラメータをベースにすばやく高度でパワフルなレポートが可能です。

100%Java によるデザインで、EspressReport はどんなプラットフォームでもバーチャルに稼動し、エンタープライズ レベルのレポートリング ニーズまで処理できる拡張性を有しています。パワフルな機能セット、柔軟性のあるアーキテクチャー、イージーツーユースのデザイン環境で EspressReport は次世代の Web レポートリングの新たな標準を確立しました。

Copyright © 2001 Quadbase Systems, Inc. All rights reserved.

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.

日本語訳 © 株式会社クライム

本社: 〒103-0014 東京都中央区日本橋蛸殻町 1-25-4 日本橋栄ビル 4F