



Jaspersoft

導入・活用ガイド（２）

JasperReportsによるレポートの Webアプリへの組み込み

野村総合研究所
オープンソースソリューション推進室



野村総合研究所のOpenStandia（オープンスタンディア）は、おかげさまで、2006年のサービス開始から2011年までの5年間で契約数累計が1,000件を突破いたしました！

株式会社 野村総合研究所
オープンソースソリューション推進室

Mail : ossc@nri.co.jp Web: <http://openstandia.jp/>



■ はじめに

皆様、お世話になっております。

第二回となります今回はJasperReportsを使用して帳票をWEBアプリで表示する手順をご紹介します。

以下の流れでご説明いたしますので、よろしくお願いいたします。

I JasperReportsのインストール

■ 環境

OS	WindowsXP SP3
DB	MySQL 5.1
Java	JDK 6 Update 22
JasperReports	JasperReports 3.7.6
Appサーバ	Tomcat 6.0.29

II JasperReportsによるレポートのWebアプリへの組み込み例

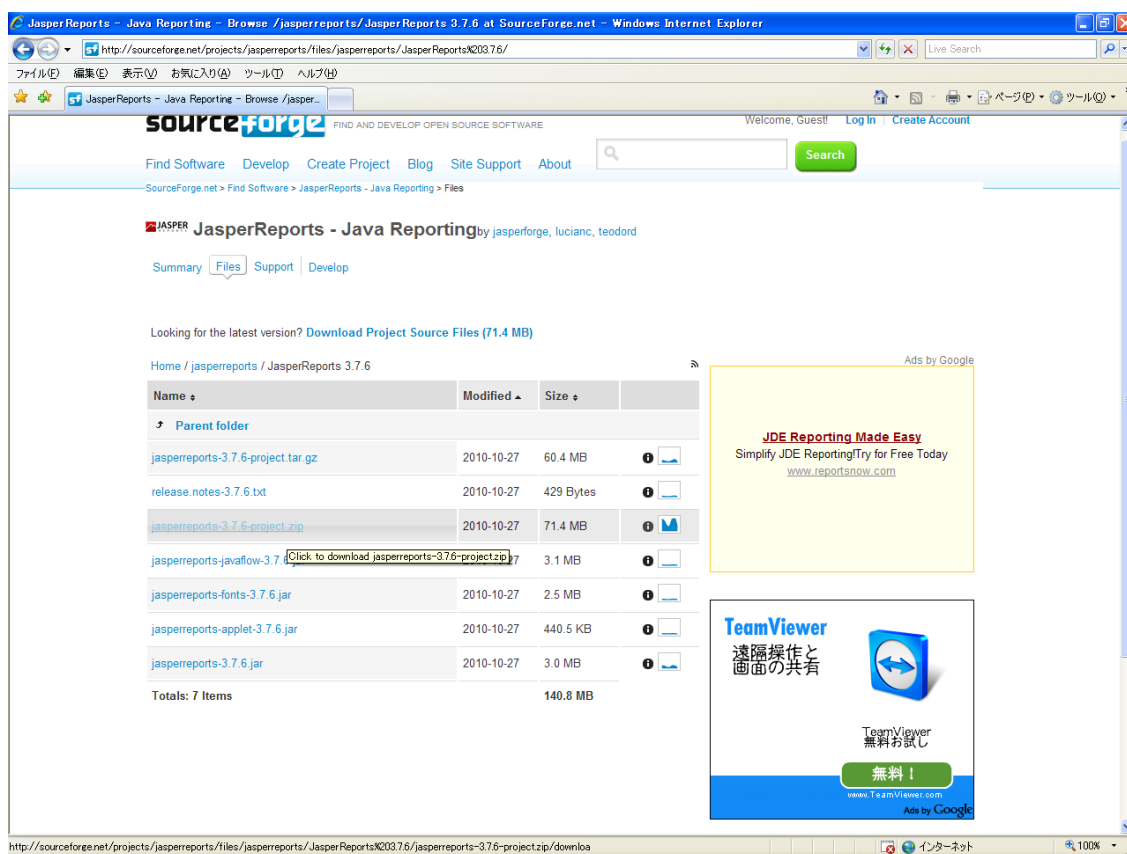
■ iReportで作成した請求書をJSPで表示する。

I . JasperReportsのインストール

■ ZIPファイルのダウンロード

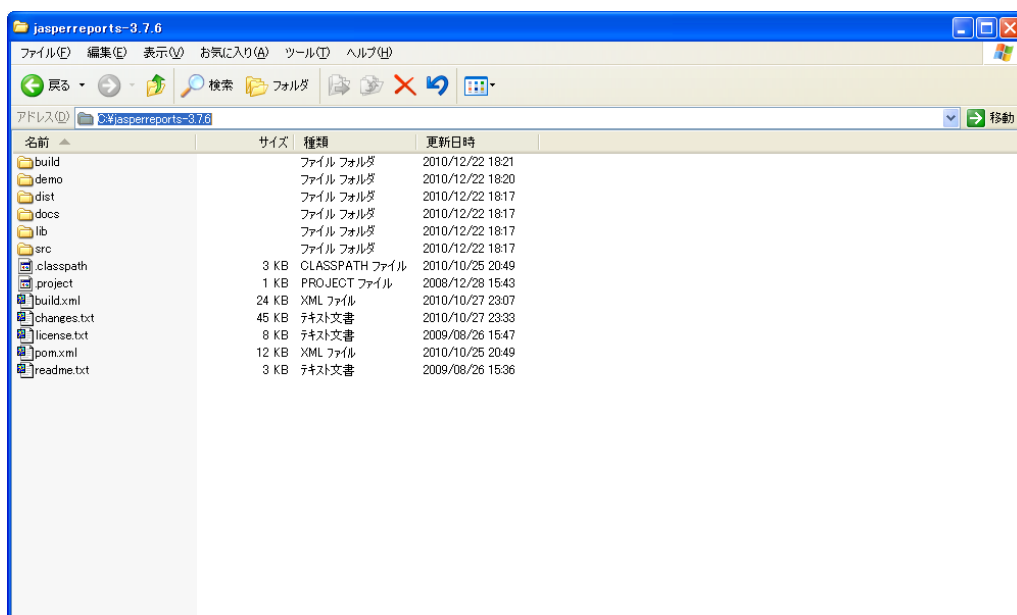
以下のURLからJasperReportsのZIPファイル
(jasperreports-3.7.6-project.zip)をダウンロードします。

URL:<http://sourceforge.net/projects/jasperreports/files/>
 ※今回は2010/11/8時点で最新のversion3.7.6を使用します。



■ ZIPファイルの解凍

ダウンロードしたZIPファイルを解凍ソフトで解凍してください。
以下のようなフォルダ構成になっているはずです。
インストールというよりただのダウンロードでしたが、
JasperReportsはJarライブラリの集合のようなものです。



Ⅱ. レポートのWebアプリへの組み込み例

■ iReportで作成した請求書をJSPで表示する。

それでは実際にWebアプリに組み込んでいきましょう。

◆ JasperReportsからJarファイル取得

「Tomcatのコンテキストルート/ROOT」以下のWEB-INFに
libディレクトリを作成して以下のJarファイルをコピーしてください。

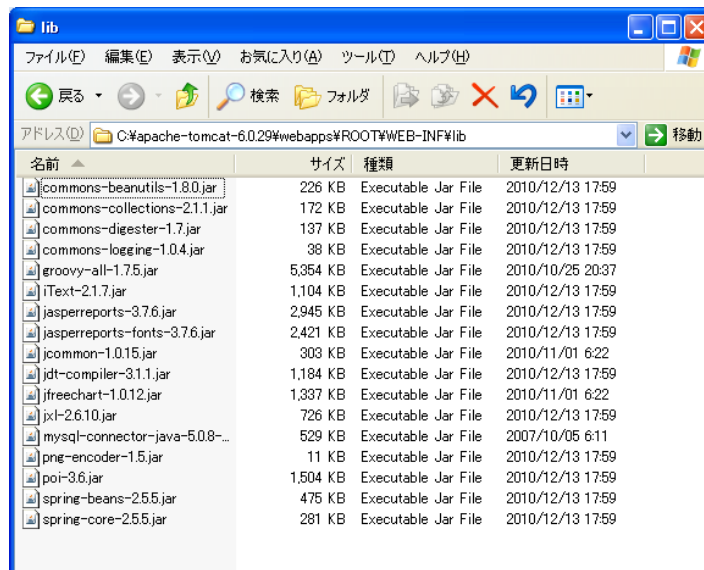
commons-beanutils-1.8.0.jar
commons-collections-2.1.1.jar
commons-digester-1.7.jar
commons-logging-1.0.4.jar
groovy-all-1.7.5.jar
iText-2.1.7.jar
jasperreports-3.7.6.jar
jasperreports-fonts-3.7.6.jar
jcommon-1.0.15.jar
jdt-compiler-3.1.1.jar
jfreechart-1.0.12.jar
jxl-2.6.10.jar
png-encoder-1.5.jar
poi-3.6.jar
spring-beans-2.5.5.jar
spring-core-2.5.5.jar

Ⅱ. レポートのWebアプリへの組み込み例

◆DB接続用Jarファイルの取得

同様にDB接続用にmysql.jarを以下のURLからダウンロードしてLibフォルダにコピーしてください

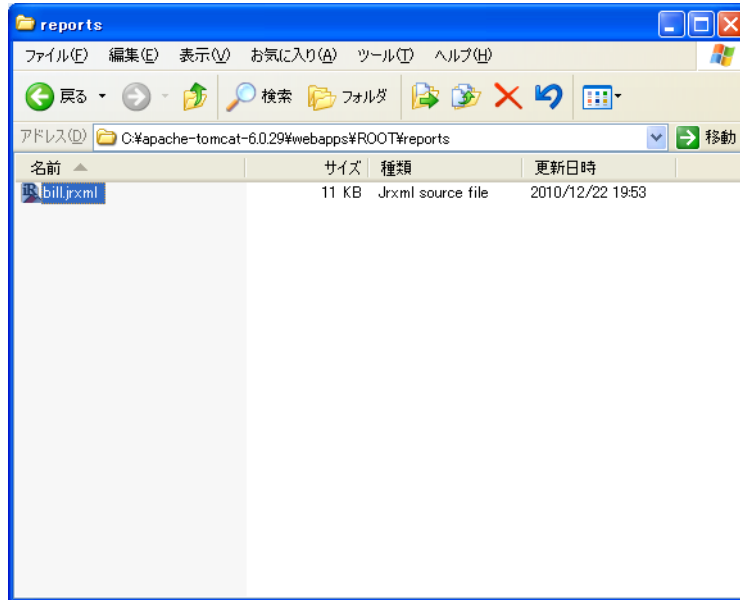
URL: <http://dev.mysql.com/downloads/connector/j/5.0.html>
 mysql-connector-java-5.0.8-bin.jar



Ⅱ. レポートのWebアプリへの組み込み例

◆ iReportで作成したjrxmlファイルのコピー

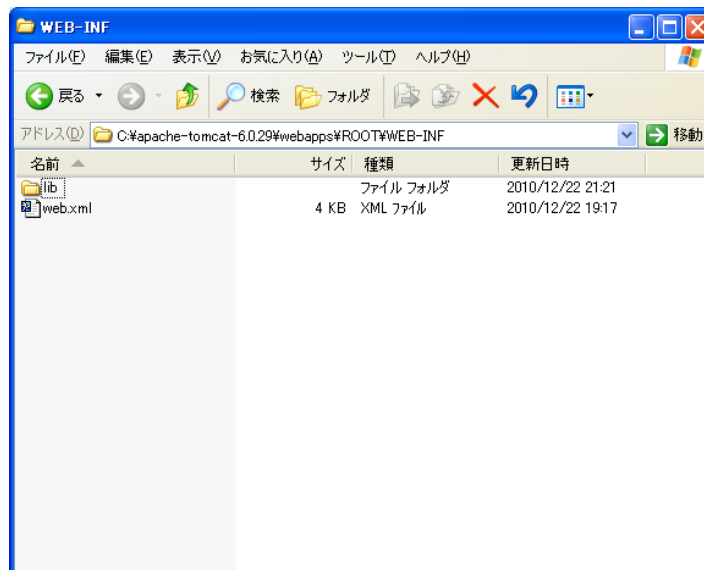
前回作成したjrxmlファイルを「Tomcatのコンテキストルート/ROOT」以下にreportsフォルダを作成して、そこにコピーしてください。



Ⅱ. レポートのWebアプリへの組み込み例

◆ web.xmlのコピー

JasperReportsのサーブレットがすでに記述されているファイルが
「JasperReportsインストールDir¥demo¥samples¥webapp¥WEB-INF」
以下にあるのでそれを
「Tomcatのコンテキストルート/ROOT/WEB-INF」
にコピーしてください。



Ⅱ. レポートのWebアプリへの組み込み例

◆ JSPファイルの作成

「Tomcatのコンテキストルート/ROOT」以下にjspフォルダを作成し、そこに以下のファイルを作成してください。

```
<%@ page errorPage="error.jsp" %>
<%@ page import="net.sf.jasperreports.engine.*" %>
<%@ page import="net.sf.jasperreports.engine.util.*" %>
<%@ page import="net.sf.jasperreports.engine.export.*" %>
<%@ page import="net.sf.jasperreports.j2ee.servlets.*" %>
<%@ page import="java.util.*" %>
<%@ page import="java.io.*" %>
<%@ page import="java.sql.*" %>

<%
    // JRXML ファイルの指定
    File reportFile = new
File(application.getRealPath("/reports/bill.jrxml"));
    if (!reportFile.exists())
        throw new JRRuntimeException("File WebappReport.jasper
not found. The report design must be compiled first.");

    // JRXML ファイルのコンパイル
    JasperReport jasperReport =
JasperCompileManager.compileReport(reportFile.getPath());

    // データソースの設定
    Class.forName("com.mysql.jdbc.Driver");
    Connection con = null;
    con = DriverManager.getConnection(
        "jdbc:mysql://localhost/report", "ユーザー名", "パスワード");

    // パラメータ設定
    Map parameters = new HashMap();
    parameters.put("ReportTitle", "Address Report");
    parameters.put("BaseDir", reportFile.getParentFile());

    JasperPrint jasperPrint =
        JasperFillManager.fillReport(
            jasperReport,
            parameters,
            con);

    JRHtmlExporter exporter = new JRHtmlExporter();

    StringBuffer sbuffer = new StringBuffer();

    session.setAttribute(ImageServlet.DEFAULT_JASPER_PRINT_SESSI
ON_ATTRIBUTE, jasperPrint);

    exporter.setParameter(JRExporterParameter.JASPER_PRINT,
jasperPrint);
    exporter.setParameter(JRExporterParameter.OUTPUT_STREAM,
response.getOutputStream());
    exporter.setParameter(JRHtmlExporterParameter.IMAGES_URI,
"..../servlets/image?image=");

    exporter.exportReport();

%>
```

Ⅱ. レポートのWebアプリへの組み込み例

◆ 動作確認

それではTomcatを起動して確認しましょう。
 以下のような画面が表示されるはずです。



前回iReportで作成した帳票がWebアプリケーションから読み込んで表示することができました。

Ⅱ. レポートのWebアプリへの組み込み例

◆ まとめ

最後に処理の実装のポイントを簡単に記載しておきます。

① 帳票ファイルの取得

```
// JRXML ファイルの指定  
File reportFile = new File(application.getRealPath("/reports/bill.jrxml"));
```

コンテキストルートを起点にしてパスを記述してください。

② 帳票ファイルのコンパイル

```
// JRXML ファイルのコンパイル  
JasperReport jasperReport = JasperCompileManager.compileReport(reportFile.getPath());
```

JRXMLファイルを指定する場合はコンパイルが必要になります。
すでにJasperファイルを作成している場合はコンパイルは不要です。
動的なデータのバインドが不要な帳票の場合はJprintファイルを用意することもできます。
そうすることで処理時間を短縮することができます。

③ 出力設定

```
exporter.setParameter(JRExporterParameter.OUTPUT_STREAM, response.getOutputStream());
```

出力する際はバイトストリームを直接出力するようにします。
でないとエンコードを二重にかけてしまうので、
日本語が表示されなくなってしまう。

以上でJasperReportsでのWebアプリ開発の解説は終了です。

- OpenStandiaは、「攻めのIT」を支援します。
- オープンソースのことなら、なんでもご相談ください！

オープンソースまるごと



お問い合わせは、NRIオープンソースソリューション推進室へ



ossc@nri.co.jp



<http://openstandia.jp/>