

Reporting mit BIRT und Ivy

Titel: Reporting mit BIRT und Ivy
Anleitung zur Einrichtung und Gebrauch des Ivy BIRT Plugin
Thema / Projekt: Ivy-BIRT Extension
Autor: mde - Soreco International
Datum: 14. April 2011
DOC: Version 1.3

Änderungsnachweis

Wann	Was	Wer	Version
17.02.2010	Erste Grundversion der Beschreibung	Markus Demolsky	1.0
18.11.2010	Update auf neue Version 4.1.2 des BIRT Plugin - Neue Prozess Elemente in der Palette. Create Report with Database und Create Report with XML Input	Markus Demolsky	1.1
05.03.2011	Update auf neue Version 4.1.3 des BIRT Plugin - Neues Prozess Elemente in der Palette. CreateDynamicReport	Markus Demolsky	1.2
14.04.2011	Ergänzungen für Xpert.ivy 4.2	Markus Demolsky	1.3

Einführung	3
Voraussetzungen	4
BIRT Designer	4
BIRT Runtime	4
Ivy-BIRT Plugin.....	5
BIRT Runtime	5
BIRT Plugin im Ivy Designer	6
Installation des Plugin	6
BIRT Prozess Palette	8
BIRT und Ivy Datenbanken	13
BIRT Plugin am Ivy Server.....	15
Installation des Plugin	15
BIRT Runtime JARs einrichten	15
Ivy Extensions erweitern	15
BIRT Properties einspielen	16
Beispielprojekt.....	17

Einführung

In Prozessapplikationen müssen früher oder später Auswertungen von Prozessdaten gemacht werden. Diese Auswertungen sind in den meisten Fällen projektspezifisch und müssen somit für jedes Projekt individuell angepasst werden. Mit Ivy können die Entwickler selbst entscheiden, welche Report Methode für sie am besten geeignet ist. Dieses Dokument stellt die BIRT Variante mit Ivy vor.

BIRT ist ein Eclipse-basierte Open-Source-Reporting-System für Web-Anwendungen, vor allem auf Java und J2EE. IRT besteht aus zwei Hauptkomponenten: einen Bericht Designer auf Eclipse basiert, und einer Runtime-Komponente, die Sie hinzufügen können, um Ihre Anwendung Server. BIRT bietet auch ein Diagramm-Modul, mit der Sie Grafiken zu Ihrer eigenen Anwendung.

Dieses Dokument ist keine Einführung in BIRT sondern setzt voraus, dass der Entwickler bereits BIRT Kenntnisse hat.

Weiterführende Literatur zu BIRT:

- <http://www.eclipse.org/birt/>
- <http://www.birt-exchange.org/#newest>
- <http://birtworld.blogspot.com/>

Gute Bücher zu BIRT:

- Practical Data Analysis and Reporting with BIRT
- BIRT: A field Guide to Reporting

Voraussetzungen

DAS IVY BIRT PLUGIN UNTERSTÜTZT DIE BIRT VERSION 2.5

BIRT Designer für Xpert.ivy 4.1.x

Der BIRT Designer muss in der aktuellen Ivy Version (4.1.x) noch als eigenständige Eclipse Instanz betrieben werden. Der Grund dafür liegt in der Eclipse Plattform, da Ivy auf Eclipse 3.2 und BIRT auf Eclipse 3.4+ aufsetzt. Daher ist es nicht möglich den Report Designer als eigenständige Perspektive in den Ivy Designer zu integrieren.

Der BIRT Designer kann unter <http://download.eclipse.org/birt/downloads/> runtergeladen werden. Es muss nur der RCP Designer von der Seite runtergeladen werden. Die Installationsanleitung befindet sich ebenfalls auf der Homepage:
<http://www.eclipse.org/birt/phoenix/build/>

Mit dem Designer können die Berichte erstellt werden und bietet auch eine integrierte Vorschau des Berichtes an.

BIRT Designer für Xpert.ivy 4.2

In Xpert.ivy 4.2 ist der Report Designer bereits als eigene Perspektive inkludiert. Daher braucht auch kein externer BIRT Designer installiert werden.

BIRT Runtime

Damit die Berichte zur Laufzeit auf dem Ivy Server verarbeitet werden können, muss eine BIRT Runtime installiert werden.

Die Runtime kann ebenfalls unter <http://download.eclipse.org/birt/downloads/> runtergeladen werden. Es gibt einen Punkt „Runtime“, welcher nur die Runtime Komponenten beinhaltet.

WICHTIG! Die BIRT Versionen des Designer und der BIRT Runtime müssen übereinstimmen, da es sonst zu Problemen bei der Verarbeitung kommen kann.

Ivy-BIRT Plugin

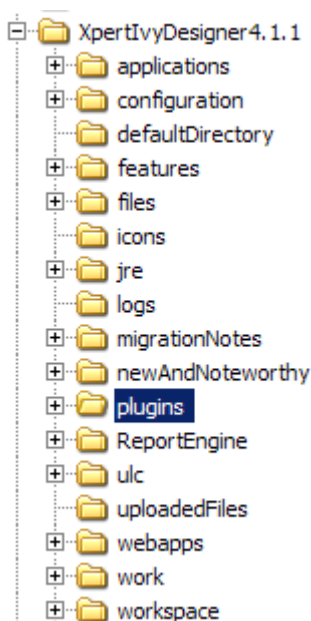
Damit BIRT in Ivy Projekten verwendet werden kann, muss das Plugin für BIRT erworben werden. Das Plugin wird als einzelne JAR-Datei ausgeliefert. Die Einrichtung des Plugins wird im nächsten Abschnitt beschrieben.

BIRT Runtime für Xpert.ivy 4.1.x

Die BIRT Runtime muss nur für Xpert.ivy 4.1.x Versionen eingerichtet werden. Wenn mit Xpert.ivy 4.2 gearbeitet wird, kann dieser Abschnitt übersprungen werden.

Nachdem die BIRT Runtime von der BIRT Download Seite runtergeladen (ZIP-Datei) muss die ZIP Datei entpackt werden. Darunter befindet sich ein Verzeichnis „ReportEngine“. Für die Ausführung benötigt man nur den ReportEngine Teil. Dieses Verzeichnis kann auf einen beliebigen Ort des Rechners kopiert werden. Dies kann sinnvoll sein, wenn unterschiedliche Entwicklungsumgebungen von Ivy die Runtime verwenden.

Es besteht jedoch auch die Möglichkeit die Runtime direkt in den Designer zu kopieren, wie folgende Darstellung zeigt:



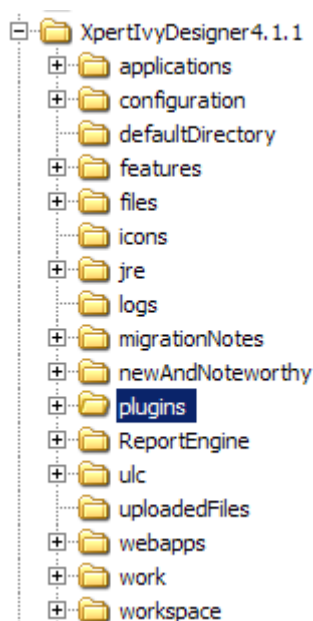
BIRT Plugin im Ivy Designer

Installation des Plugin

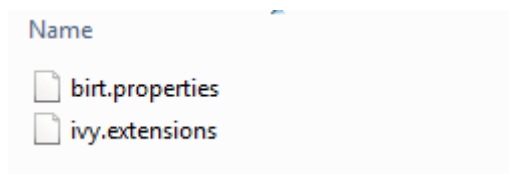
Wie bei den Voraussetzungen erwähnt, wird das BIRT Plugin als eine JAR-Datei ausgeliefert.

`ch.ivyteam.ivy.extension.birt_<version>.jar`

Vor der Installation, muss der Designer in welchem das Plugin installiert werden soll, geschlossen sein. Dieses Plugin muss in das plugins Verzeichnis des Designers kopiert werden, wie in der unten angeführten Abbildung dargestellt. Wird eine neue Version des Plugins ausgeliefert, muss die bestehende Version `ch.ivyteam.ivy.extension.birt_xxxx` gelöscht werden.



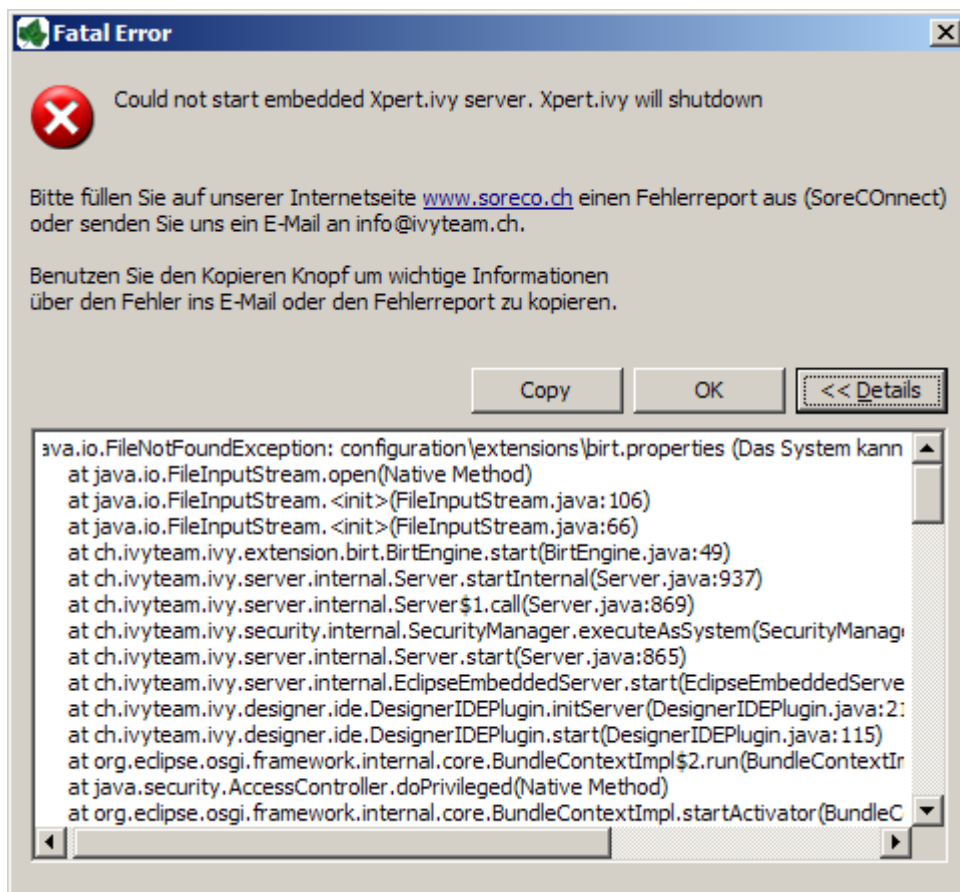
Das Plugin muss nun auch wissen wo die ReportEngine liegt. In der oben angeführten Abbildung wurde die Report Engine direkt in das Verzeichnis vom Designer kopiert. Einstellungen für das Plugin werden aus der Konfigurationsdatei `/configuration/extensions/birt.properties` geladen. Diese Datei muss erstellt werden.



Property	Angabe erforderlich	Beschreibung, Beispiel
engineHome	JA	Pfad auf den ReportEngine Ordner, die Laufzeitumgebung von BIRT. Wird die Report Engine in den Designer kopiert, kann die Angabe relativ erfolgen, also

		engineHome=ReportEngine
designRepository	NEIN	<p>Globales Design Repository für Berichte. Zur Designzeit werden im BIRT Designer die Berichte erstellt. Das Ergebnis ist eine Report Designdatei mit der Endung .rptdesign. Wird ein globales Design Repository für alle Berichte verwendet, kann über diese Konfiguration der Pfad zum Design Repository angegeben werden.</p> <p>HINWEIS! Die Angabe des Design Repository muss absolut sein.</p> <p>Wird diese Konfiguration nicht definiert, muss bei jedem Bericht der absolute Pfad angegeben werden.</p>

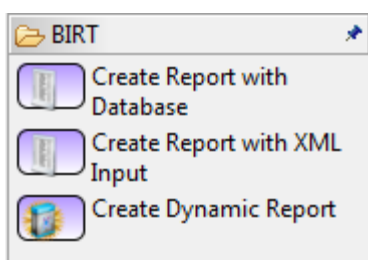
Fehlt die Datei, erscheint beim Start des Designers sofort der Fehler:



Weitere Schritte für die Benutzung des Ivy-BIRT Plugins brauchen für den Designer nicht mehr gemacht werden.

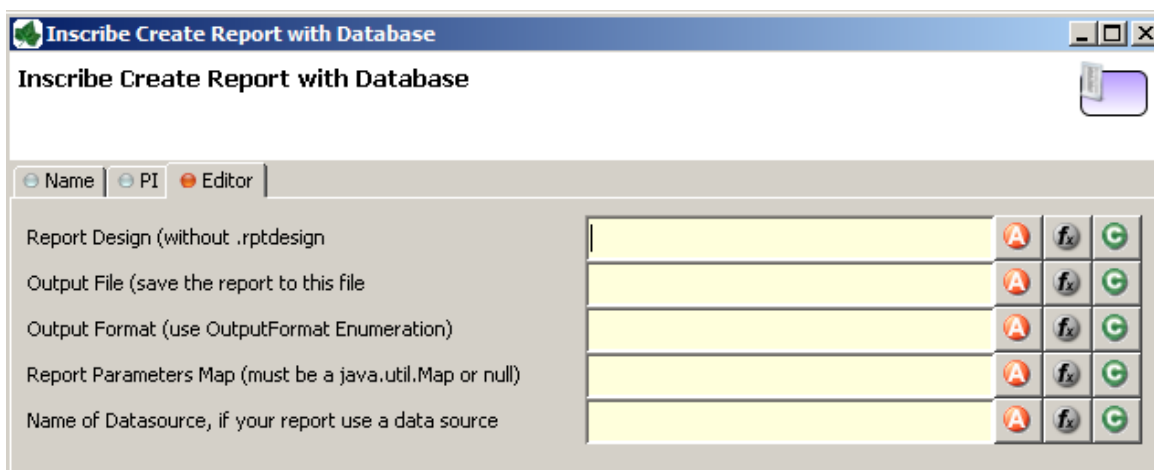
BIRT Prozess Palette

Das Plugin bietet dem Prozess Designer eine Prozesspalette „BIRT“ an, wie in der Abbildung ersichtlich.



Create Report with Database

Für die Erstellung von Berichten mit Anbindung an eine Datenbank. Siehe auch nächsten Abschnitt (BIRT und Ivy Datenbanken)



Eigenschaft	Angabe erforderlich	Beschreibung
Report Design	JA	Das Report Design für die Generierung. Wurde bei den Konfigurationen ein Design Repository angegeben, dann muss hier die relative Pfadangabe zum Bericht gemacht werden. Wird der Bericht zum Beispiel in ein Unterverzeichnis meineKategorie im Report Repository gespeichert, dann muss hier angegeben werden: „meineKategorie/meinDesign“. Wurde kein Design Repository angegeben, muss hier die absolute Angabe zum Report Design gemacht werden
Output File	JA	Ist das File Objekt (Ivy File) in dem der Bericht gespeichert werden soll. Dies können sowohl persistente

		als auch temporäre Dateien sein. Der Name der Datei muss jedoch in einem vorigen Schritt gemacht werden, da die Namensvergabe und/oder die Bestimmung ob temporär oder persistent nicht in einem Skript Editor gemacht werden kann.
Output Format	JA	<p>Zur Zeit unterstützt das Ivy BIRT Plugin folgende Formate:</p> <ul style="list-style-type: none"> • PDF • HTML • DOC (Word) <p>Die Angabe des Output Format ist eine Enumeration vom Typ <code>ch.ivyteam.ivy.extension.birt.OutputFormat</code> Das bietet dem Entwickler die Möglichkeit, dass das Output Format in einem vorigen Prozess Schritt berechnet werden kann, d.h. abhängig vom Prozessverlauf kann entweder ein DOC, HTML oder PDF erstellt werden.</p>
Report Parameter Map	NEIN	Werden im Bericht Parameter definiert, müssen diese in Form einer Map (Key, Value) übergeben werden. Die Befüllung der Parameter muss in einem der vorigen Schritte im Prozess gemacht werden. Falls keine Parameter übergeben werden, muss null übergeben werden.
Datasource	NEIN	<p>Die zu verwendete Datenbank aus dem Datenbankverzeichnis von Ivy, falls der Bericht auf die Datenbank zugreifen soll.</p> <p>Bei Verwendung der Datenbank müssen ein paar Punkte beachtet werden → Siehe Parameterdefinition für JDBC Verwendung.</p>

Create Report with XML Input

Für die Erstellung von Berichten mit einem XML Input.

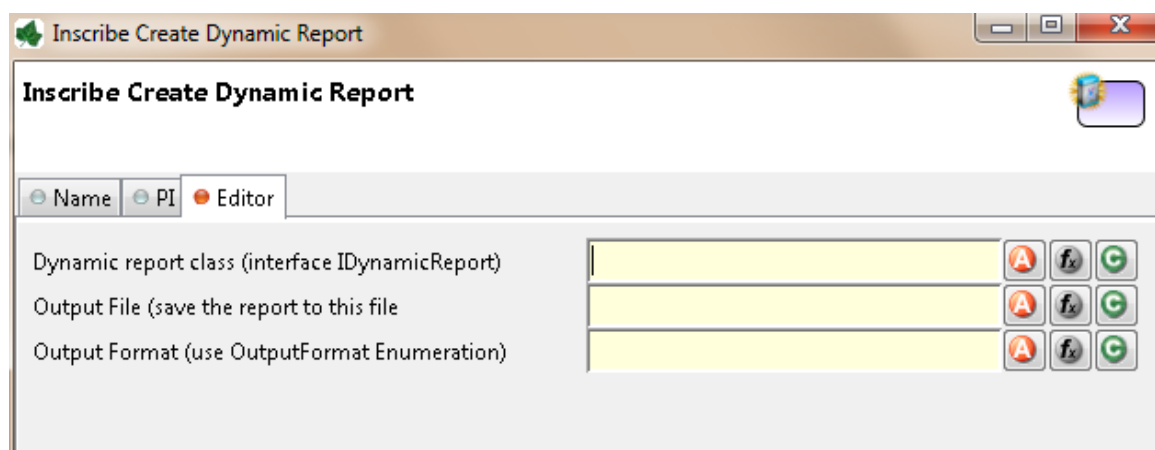
Eigenschaft	Angabe erforderlich	Beschreibung
Report Design	JA	Siehe oben
Output File	JA	Siehe oben
Output Format	JA	Siehe oben
Report Parameter Map	NEIN	Siehe oben
XML Input String	NEIN	Wird der Bericht auf Basis einer XML Datei gemacht, welche als Input dient, so muss diese XML Datei als String übergeben werden. Beispiel (reportXmlContent ist ein String und beinhaltet das XML): <pre>new java.io.ByteArrayInputStream(in.reportXmlContent.getBytes("UTF-8"))</pre>

Create Dynamic Report

Dynamische Berichte können über die Java API erstellt werden. Diese Variante benötigt kein Report Design, da der gesamte Bericht in einer Java Klasse erstellt wird.

Bei der Erstellung von dynamischen Bericht muss der Entwickler das Report Object Model von BIRT sehr gut kennen, da der Bericht nicht wie üblich über den Report Designer erstellt wird, sondern ausschließlich über die API. Eine ausführliche Dokumentation befindet sich unter:

<http://help.eclipse.org/helios/topic/org.eclipse.birt.doc.isv/model/api/index.html>



Eigenschaft	Angabe erforderlich	Beschreibung
Dynamic Report Class	JA	Java Klasse die den dynamischen Bericht erstellt. Diese Klasse muss das Interface <code>ch.ivyteam.ivy.extension.birt.dynreport.IDynamicReport</code> implementieren.
Output File	JA	Siehe oben
Output Format	JA	Siehe oben

Die Birt Extension stellt eine abstrakte Implementierung für einen dynamischen Bericht (AbstractDynamicReport) zur Verfügung, welche die Erstellung von dynamischen Berichten sehr vereinfacht. Wenn diese Klasse erweitert wird, muss lediglich die Methode `createDynamicReport` implementiert werden, wo das gesamte Report Design umgesetzt werden kann. Die abstrakte Klasse übernimmt Basisarbeiten und stellt Hilfsmethoden zur Verfügung:

- `getReportDesign` → Zugriff auf das Report Design
- `getDesignFactory` → Bietet Zugriff auf die Design Factory für die Erstellung von Report Elementen (Labels, Texten, Grids, Tabellen, etc.)
- `addCmsImageToReport` → Kann in Ivy befindliche CMS Images den Bericht hinzufügen

Das erste Codebeispiel SimpleDynamicReport erstellt einen Bericht mit einem Label.

Beispiel 1

```
public class SimpleDynamicReport extends AbstractDynamicReport {

    @Override
    public ReportDesignHandle createDynamicReport() {
        //Put your report elements to the report design
        try {
            LabelHandle myLabel = getDesignFactory().newLabel("myLabel");
            myLabel.setText("Hello, I am here");
            getReportDesign().getBody().add(myLabel);
        } catch (Exception e) {
            e.printStackTrace();
        }
        return getReportDesign();
    }
}
```

Das zweite Beispiel erweitert das erste Beispiel und zeigt im Bericht noch ein Image aus dem Ivy CMS an. Die Images müssen zuerst dem Bericht hinzugefügt werden, damit diese anschließend im Bericht verwendet werden können.

Beispiel 2

```
public class SimpleDynamicReport extends AbstractDynamicReport {

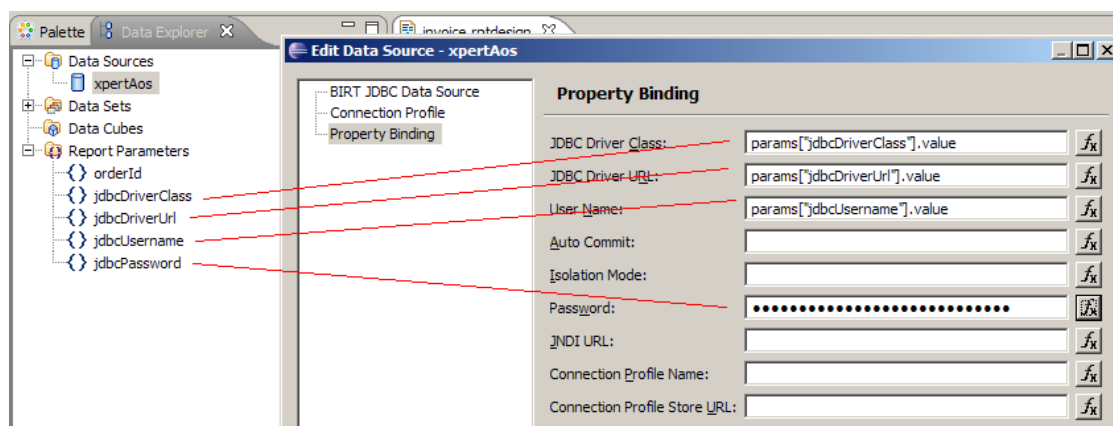
    @Override
    public ReportDesignHandle createDynamicReport() {
        //Put your report elements to the report design
        try {
            LabelHandle myLabel = getDesignFactory().newLabel("myLabel");
            myLabel.setText("Hello, I am here");
            getReportDesign().getBody().add(myLabel);
            addCmsImageToReport("/images/ok", "myImage");

            ImageHandle myImage = getDesignFactory().newImage("myUniqueImageName");
            myImage.setImageName("myImage")
        } catch (Exception e) {
            e.printStackTrace();
        }
        return getReportDesign();
    }
}
```

BIRT und Ivy Datenbanken

In den meisten Fällen greifen die BIRT Berichte auf Datenbanken zu, welche auch in Ivy verwendet werden. In Ivy werden die externen Datenbanken im Datenbank Repository definiert. Damit diese Datenbankinformationen vom Prozess an den Bericht übergeben werden können, müssen bestimmte Richtlinien eingehalten werden.

Die einzige Möglichkeit die BIRT bietet, die Datenbankverbindung zur Laufzeit zu ändern ist über Parameterübergabe.



Diese Parameter werden vom Ivy-BIRT Plugin bei ausgewählter Datenquelle automatisch mitgegeben. Daher müssen beim Design des Berichts diese Parameter auch definiert werden. Dabei ist der Name des Parameters WICHTIG!

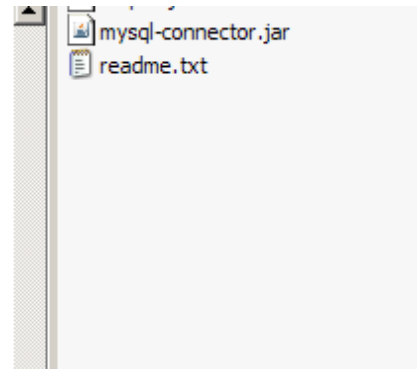
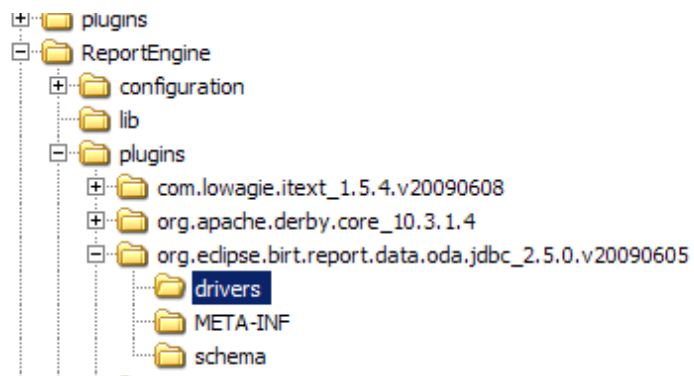
Parameter	Angabe erforderlich	Beschreibung
jdbcDriverClass	JA	Der zu verwendende Datenbanktreiber
jdbcDriverUrl	JA	Die URL zu der Datenbank
jdbcUsername	JA	Username
jdbcPassword	JA	Passwort

Die Befüllung dieser Parameter erfolgt durch das Plugin automatisch und natürlich auch abhängig vom aktuell gesetzten Environment der Applikation.

Weiteres benötigt die BIRT Runtime auch den Treiber für die Datenbank. Alle Datenbanktreiber, welche die BIRT Runtime unterstützen soll müssen in das Verzeichnis

ReportEngine\plugins\org.eclipse.birt.report.data.oda.jdbc_2.5.0.v20090605\drivers

kopiert werden. Dies kann der gleiche Treiber, welcher bereits im BIRT Designer für den Zugriff auf die Datenbank verwendet wurde sein.



BIRT Plugin am Ivy Server

Installation des Plugin























Wie bei den Voraussetzungen erwähnt, wird das BIRT Plugin als eine JAR-Datei ausgeliefert.

ch.ivyteam.ivy.extension.birt_4.1.2.jar

Dieses Plugin muss in das **lib/shared** Verzeichnis des Servers kopiert werden.

BIRT Runtime JARs einrichten

Zusätzlich zum Plugin müssen noch die JAR-Dateien der Report Engine in das **lib/shared** des Ivy Server kopiert werden. Die JAR-Dateien der Report Engine befinden sich im Ordner ReportEngine/lib.

Name ^	Größe	Typ	Geändert am
 chartengineapi.jar	1.442 KB	Executable Jar File	17.06.2009 06:59
 com.ibm.icu_4.0.1.v20090415.jar	5.652 KB	Executable Jar File	17.06.2009 06:59
 commons-cli-1.0.jar	30 KB	Executable Jar File	17.06.2009 06:59
 coreapi.jar	261 KB	Executable Jar File	17.06.2009 06:59
 crosstabcoreapi.jar	98 KB	Executable Jar File	17.06.2009 06:59
 dataadapterapi.jar	38 KB	Executable Jar File	17.06.2009 06:59
 dataaggregationapi.jar	2 KB	Executable Jar File	17.06.2009 06:59
 dataextraction.jar	5 KB	Executable Jar File	17.06.2009 06:59
 dteapi.jar	132 KB	Executable Jar File	17.06.2009 06:59
 emitterconfig.jar	10 KB	Executable Jar File	17.06.2009 06:59
 engineapi.jar	111 KB	Executable Jar File	17.06.2009 06:59
 flute.jar	70 KB	Executable Jar File	17.06.2009 06:59
 js.jar	740 KB	Executable Jar File	17.06.2009 06:59
 modelapi.jar	2.119 KB	Executable Jar File	17.06.2009 06:59
 modelodaapi.jar	10 KB	Executable Jar File	17.06.2009 06:59
 odadesignapi.jar	57 KB	Executable Jar File	17.06.2009 06:59
 org.apache.commons.codec_1.3.0.v20080530-1600.jar	53 KB	Executable Jar File	17.06.2009 06:59
 org.eclipse.emf.common_2.5.0.v200906080927.jar	199 KB	Executable Jar File	17.06.2009 06:59
 org.eclipse.emf.ecore.xmi_2.5.0.v200906080927.jar	213 KB	Executable Jar File	17.06.2009 06:59
 org.eclipse.emf.ecore_2.5.0.v200906080927.jar	1.031 KB	Executable Jar File	17.06.2009 06:59
 org.w3c.css.sac_1.3.0.v200805290154.jar	29 KB	Executable Jar File	17.06.2009 06:59
 scriptapi.jar	58 KB	Executable Jar File	17.06.2009 06:59

Ivy Extensions erweitern

In der Serverumgebung muss dem Ivy Server die Extension explizit bekannt gegeben werden, damit der Server die neuen Prozesselemente auch verarbeiten kann. Alle Ivy Erweiterungen

werden in der Konfigurationsdatei **configuration/extensions/ivy.extension** definiert. Die erweiterte ivy.extensions Datei wird zusammen mit dem Ivy BIRT Plugin ausgeliefert.

Die bestehende Datei kann jedoch mit folgenden Einträgen für das Ivy-BIRT Plugin erweitert werden.

```
<extensionPoint interface="ch.ivyteam.ivy.server.IServerExtension"
bundle="ch.ivyteam.ivy.extension.birt">
    <extension bundle="ch.ivyteam.ivy.extension.birt"
class="ch.ivyteam.ivy.extension.birt.BirtEngine"/>
</extensionPoint>

<extensionPoint interface="ch.ivyteam.ivy.java.IIvyProjectClassPathExtension"
bundle="ch.ivyteam.ivy.extension.birt">
    <extension bundle="ch.ivyteam.ivy.extension.birt"
class="ch.ivyteam.ivy.extension.birt.BirtProjectClassPathExtension"/>
</extensionPoint>

<extensionPoint
interface="ch.ivyteam.ivy.designer.process.ui.IIvyProcessPaletteExtension"
bundle="ch.ivyteam.ivy.extension.birt">
    <extension bundle="ch.ivyteam.ivy.extension.birt"
class="ch.ivyteam.ivy.extension.birt.process.element.BirtProcessPalette"/>
</extensionPoint>

<extensionPoint
interface="ch.ivyteam.ivy.process.element.IExtensibleStandardProcessElementExtension"
bundle="ch.ivyteam.ivy.extension.birt">
    <extension bundle="ch.ivyteam.ivy.extension.birt"
class="ch.ivyteam.ivy.extension.birt.process.element.BirtProcessElements"/>
</extensionPoint>

<extensionPoint
interface="ch.ivyteam.ivy.designer.process.ui.IProcessElementUiInformationExtension"
bundle="ch.ivyteam.ivy.extension.birt">
    <extension bundle="ch.ivyteam.ivy.extension.birt"
class="ch.ivyteam.ivy.extension.birt.process.element.BirtProcessElementUiInformation"/>
</extensionPoint>
```

BIRT Properties einspielen

Wie beim Designer muß die Datei birt.properties in das Verzeichnis configuration/extensions kopiert werden.

Beispielprojekt

Mit der Auslieferung des Ivy BIRT Plugin wird ein Ivy Beispielprojekt, bestehend aus:

- Bericht Design
Ist ein Bericht mit zwei Parameter, welche vom Prozess übergeben werden können. Das Bericht Design befindet sich im Ordner `/reports/Testreport.rptdesign` im Projekt des Beispielprojekt.
- Prozess für die Berichterstellung
Der Bericht erstellt in einem Script Step die Datei und befüllt die Parameter. Im zweiten Schritt wird der Bericht über das neue Prozesselement erstellt. Bei der Erstellung des Berichts erhält der Entwickler auch DEBUG Meldungen, welche vom Plugin vorgenommen werden.

