

JasperReports

Posted on **Juli 1, 2013**



W.B.S.S ist eine Software Entwicklungs-Firma, die individuelle, komplexe und qualitativ hochstehende webbasierte Software-Lösungen anbietet.

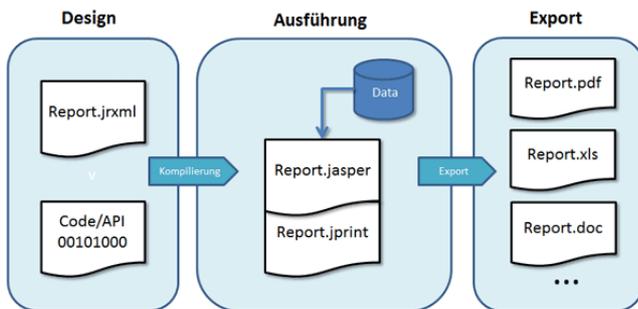
Guido Amrein, Software Architect

Abstract

JasperReports ist eine Open Source Java Klassenbibliothek welche ein umfangreiches Reporting in Java (Web-) Applikationen ermöglicht. Ursprünglich als einfache und kostengünstige Möglichkeit für das embedded Reporting in Java Applikationen gedacht, bietet heute Jaspersoft, die Firma hinter JasperReports, weitergehende Produkte an wie ein Report Designer, ein standalone Report-Server oder eine BI-Plattform. Das Herzstück dieser Produktpalette bildet aber immer die JasperReports Klassenbibliothek. Dank einer grossen und aktiven Community ist der Support, welcher auch kommerziell von Jaspersoft bezogen werden kann, gewährleistet.

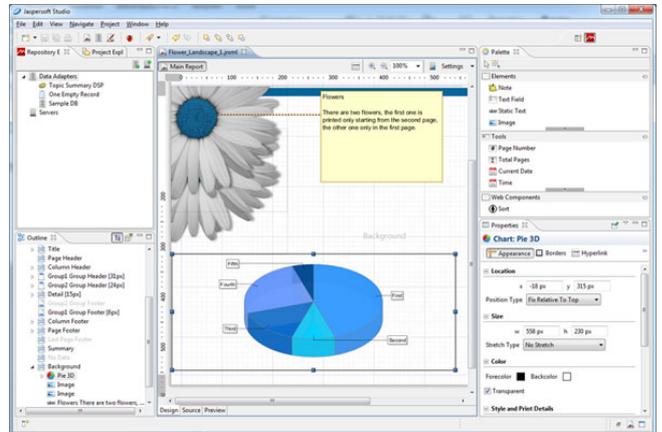
Report Lebenszyklus

Der Lebenszyklus eines Reports kann in die Phasen Design, Ausführung und Export eingeteilt werden.



DESIGN

Die Report Templates werden im Design Programm [iReport](#) oder im Eclipse-basierenden dem [Jaspersoft Studio](#) erstellt. Diese können aus anspruchsvollen Layouts mit Charts, Bildern, Subreports, Kreuztabellen und vielem mehr bestehen.

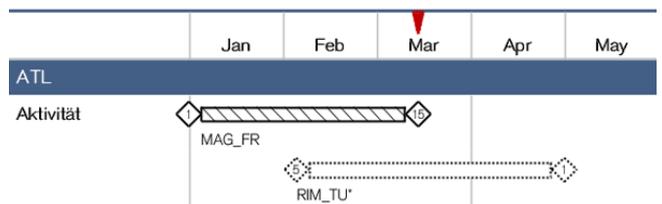


Das Resultat der Design-Phase wird in einer XML-Datei mit der Endung „jrxml“ gespeichert. Diese Datei wird kompiliert und steht danach für die Ausführung zur Verfügung. Daneben kann das Design auch über das JasperReports API dynamisch im Code erfolgen. Beispielsweise erstellt der nachfolgende Code ein Rechteck:

```

JRDesignRectangle rect = new
JRDesignRectangle();
rect.setX(0);
rect.setY(0);
rect.setWidth(100);
rect.setHeight(25);
rect.setStyleNameReference("Group");
  
```

Mit diesem Vorgehen lassen sich praktisch beliebige Kundenanforderungen umsetzen. Stellvertretend dafür das Beispiel eines Timeline-Reports, welcher erst zur Laufzeit des Programms dynamisch mittels der JasperReports API generiert wird:



Wenn die Benutzung der Low-Level API zu kompliziert erscheint, für den gibt es [Dynamic Jasper](#), ein Open Source Framework, welches die wesentlichen Funktionen zur dynamischen Reporterstellung mittels JasperReports vereinfacht.

AUSFÜHRUNG

In der Ausführungsphase wird der kompilierte Report mit den Daten gefüllt.

Es werden folgende Datenzugriffstechnologien und Formate unterstützt:

- JDBC/SQL
- JPA, Hibernate/EJBQL
- Java Beans (POJO)
- BigData (Hadoop, MongoDB)
- XML, XLS, CSV

Das Ergebnis dieses Prozesses ist eine Jasper Print-Datei die für die verschiedenen Exportformate bereitsteht.

EXPORT

Die Berichts-Exportmöglichkeiten sind vielfältig und können über 3rd-Party-Produkte noch erweitert werden.

- Standard Ausgabeformate: PDF, XLS, RTF, XML, HTML, CSV, TXT
- Mittels 3rd-Party-Produkte ([Aspose](#)): DOCX, PPT

Integration

Die Integration in eine Java (Web-)Applikation erfolgt in wenigen Schritten:

1. Die Library jasperreports-x.x.x.jar dem Klassenpfad hinzufügen
2. Integrationscode schreiben:

```
// 1. Retrieve template
InputStream jd = new FileInputStream(new
File("Report.jrxml"));
// 2. Compile design to JasperReport
JasperReport jr =
JasperCompileManager.compileReport(jd);
// 3. Create the JasperPrint object
JasperPrint jp =
JasperFillManager.fillReport(jr, params,
dataSource);
// 4. Create an output byte stream where
data will be written
ByteArrayOutputStream baos = new
ByteArrayOutputStream();
// 5. Export report
JRPdfExporter exporter = new JRPdfExporter
();
exporter.setParameter
(JRExporterParameter.JASPER_PRINT, jp);
exporter.setParameter
(JRExporterParameter.OUTPUT_STREAM, baos);
exporter.exportReport();
```

Daneben besteht die Möglichkeit alle Reports in dem auch als Open Source erhältlichen [JasperReports-Server](#) zu

verwalten. Dieses eigenständige Produkt bietet folgende Möglichkeiten:

- Deploy der Reports direkt aus iReport/Jaspersoft Studio
- Definition der Datenzugriffsschicht
- Kompilierung und Export
- Zugriff der Reports über Url-Links (REST)
- Rollenbasierte Zugriffssteuerung
- BI Funktionalität (kommerzielle Version)

Vorteile von JasperReports auf einen Blick

- Weit verbreitet, grosse Community
- Open Source, kommerzieller Support wenn benötigt
- Mächtige Design-Werkzeuge für komplex Layouts
- Programmierbares, flexibles Design mittels API
- Umfangreiche Datenzugriffs- und Exportmöglichkeiten
- Flexible Architektur: vom Embedded System bis zur BI-Plattform