

# Open Source Reports - Berichte erzeugen mit Jasper und iReports

**Dominic Weiser**  
**esentri consulting GmbH**  
**Ettlingen**

## **Schlüsselworte**

JasperReports, iReport, Open Source, EJB, JDBC, Hibernate, Datenbank, PDF, Word, XML, CSV, WYSIWYG

## **Einleitung**

Reporting in elektronischer Form ist nach wie vor ein großes Thema in vielen Unternehmen. In den meisten Fällen werden Berichte dazu verwendet, bestimmte Kennzahlen der Arbeit in eine ansprechende kompakte Form zu bringen. Am Markt sind eine Vielzahl an Technologien verfügbar die dazu verwendet werden Berichte zu erzeugen. Eine dieser Technologien ist JasperReports. Diese JasperReports lassen sich mit Hilfe des grafischen Modellierungswerkzeugs iReport erstellen.

Neben dem grafischen Editor, in welchem sich via Drag&Drop Textbausteine zu einem Bericht zusammensetzen lassen, gibt es die Möglichkeit Reports direkt als XML zu definieren und anschließend durch die Vorschau zu validieren. Die Palette der zur Verfügung gestellten Ausgaben reicht von der tabellarischen Darstellung bis zu Diagrammen. Auf der anderen Seite können eine Vielzahl an Technologien wie XML-Dateien oder Datenbanken als Datenquelle angebunden werden.

## **JasperReports**

Bei JasperReports handelt es sich um ein Java Open Source Projekt welches durch Teodor Danciu im Juni 2001 ins Leben gerufen wurde. Seit 2004 wird JasperReports vornehmlich von Jasper Soft weiterentwickelt. Ziel von JasperReports ist es aus Java Programmen Berichte in verschiedene Dateiformate zu exportieren. Zu diesen Dateiformaten zählen unter anderem: PDF, Microsoft Word/Excel, HTML und TXT.

Neben der Vielzahl an Exportformaten gibt es eine genau so große Fülle an unterstützten Datenquellen. Neben Datenbanken, welche mittels JDBC oder Hibernate eingebunden werden können EJBs, XML, CSV und weitere Dateiformate verwendet werden. Dabei übernimmt JasperReports die Transformation und Aufbereitung der Daten aus der Quelle bis hin zur fertigen Datei im gewünschten Ausgabeformat. Der spätere Bericht muss im Jasper eigenen XML-Format JXML erstellt werden. Anschließend wird die JXML Datei zu einer .jasper Datei kompiliert, welche wiederum als Grundlage für den Reportgenerator dient. Dazu wird ermöglicht dass sich alle Berichte zu jederzeit leicht anpassen und austauschen lassen. Das XML Format trägt zu einer leserlichen und leicht verständlichen Form bei, welche das Arbeiten im Team begünstigt.

In der Seitenaufteilung basiert Jasper intuitiv auf Seitenbasis. Dabei kann die Größe der Seite vom Anwender selber definiert werden. Standardmäßig sind die DIN-A Formate und US Letter enthalten. Daneben wird eine Seite in unterschiedliche Bereiche eingeteilt (siehe Unten). Innerhalb dieser Bereiche können mithilfe verschiedener Komponenten wie Textbausteine, Bereiche, Charts usw. benutzerdefinierte Berichte erzeugt werden. Dabei erfolgt die Positionierung der Komponenten wahlweise absolut oder relativ. Der Grundaufbau ist zeilenorientiert, wodurch sich vornehmlich Texte und Tabellen erzeugen lassen.

## iReports

Bei iReport handelt es sich um ein Open Source Werkzeug zum erstellen von JasperReports. Die Hauptentwicklung wird durch JasperSoft durchgeführt. iReport unterstützt den Entwickler von JasperReports über die gesamte Entwicklungszeit. Diese Unterstützung beginnt beim Einbinden der Datenquelle über den grafischen Editor zum erstellen des Reports bis hin zu einer aktuellen Vorschau.

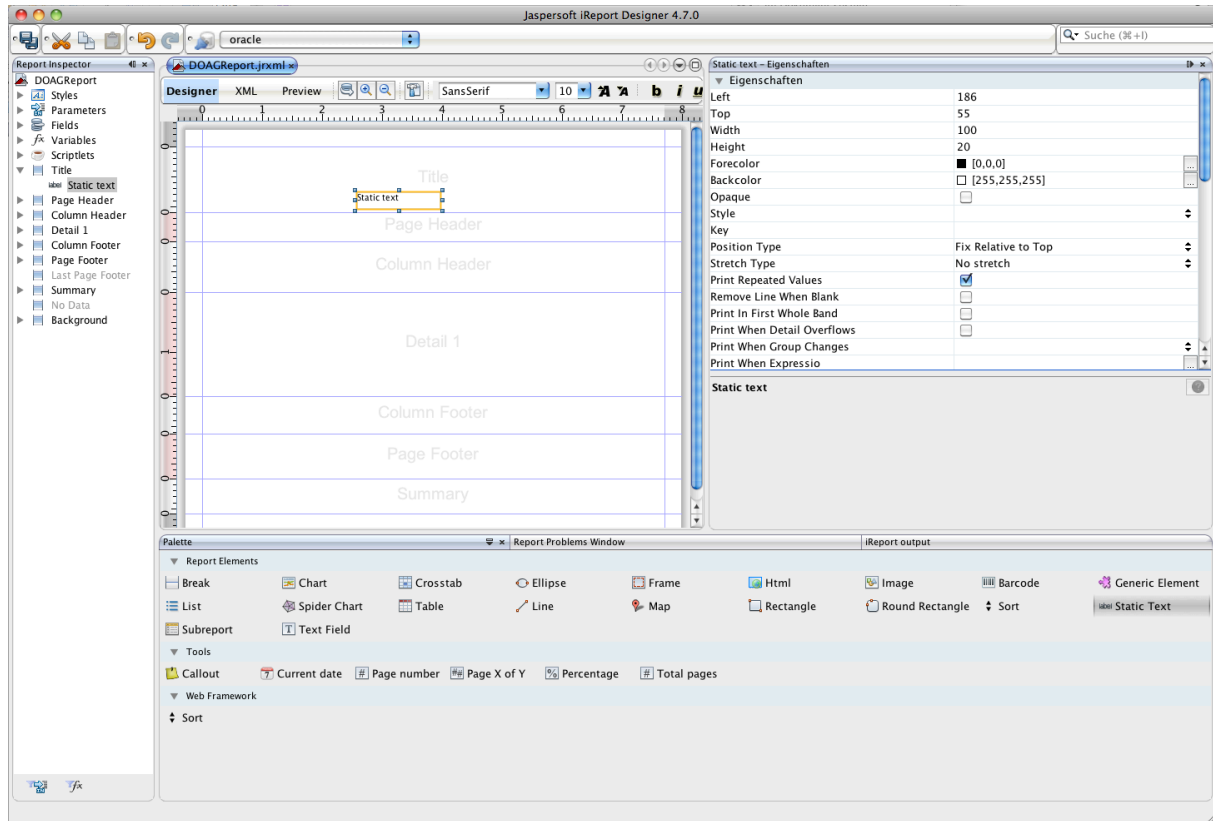


Abb. 1: Oberfläche iReport

Wie von den meisten IDEs bekannt unterteilt sich die Oberfläche iReport in verschiedene Bereiche. In der Kopfzeile werden unterschiedliche Bedienelemente dargestellt. Dazu zählen Speichern, Ausschneiden, Kopieren, Einfügen, Vor- und Rückgängig machen sowie die Wahl der Datenquelle (siehe unten). Darunter wird im Hauptbereich der aktuelle Stand des Templates dargestellt. Hierzu lassen sich verschiedene Perspektiven darstellen:

- **Designer** – Dabei handelt es sich um den grafischen Editor für die JXML Datei. Die Komponenten lassen sich im WYSIWYG Stil via Drag & Drop aus der Palette hinzufügen (siehe Unten)
- **XML** – Hier wird der Quelltext des Reports dargestellt.
- **Preview** – Hier kann die Vorschau dargestellt werden. Die eingebundene Datenquelle (siehe Unten) wird dazu verwendet das vorher erstellte Template mit Daten zu befüllen.

Um den Hauptbereich können unterschiedliche Ansichten platziert werden. Dazu zählen unter anderem:

- **Report inspector** – Hier wird die Struktur des Reports dargestellt die alle Inhalte wie Variablen, Felder und Komponenten tabellarisch enthält.
- **Eigenschaften** – Hier werden die Eigenschaften der aktuell selektierten Komponente angezeigt. Über diese Ansicht lässt sich die Komponente in Aussehen und Inhalt manipulieren. Diese Eigenschaften können alternativ direkt in den Quellcode eingetragen werden.
- **Palette** – Hier werden die von iReport und JasperReports unterstützten Komponenten dargestellt. Diese Komponenten dienen als Quelle für die Drag & Drop Funktionalität.

### Einbinden der Datenquelle

Die Grundlage von jedem Report sind die Daten, aus welchen der Report erstellt. Im Workflow ist dies daher der erste Schritt beim erstellen eines neuen Report. Dazu muss die gewünschte Datenquelle über den Headerbereich hinzugefügt und ausgewählt werden. Leider sind nicht für alle Verbindungen (speziell Datenbanken) die Treiber vorhanden. Diese können über Einstellungen -> iReport -> Classpath -> Add JAR / Add Folder eingebunden werden.

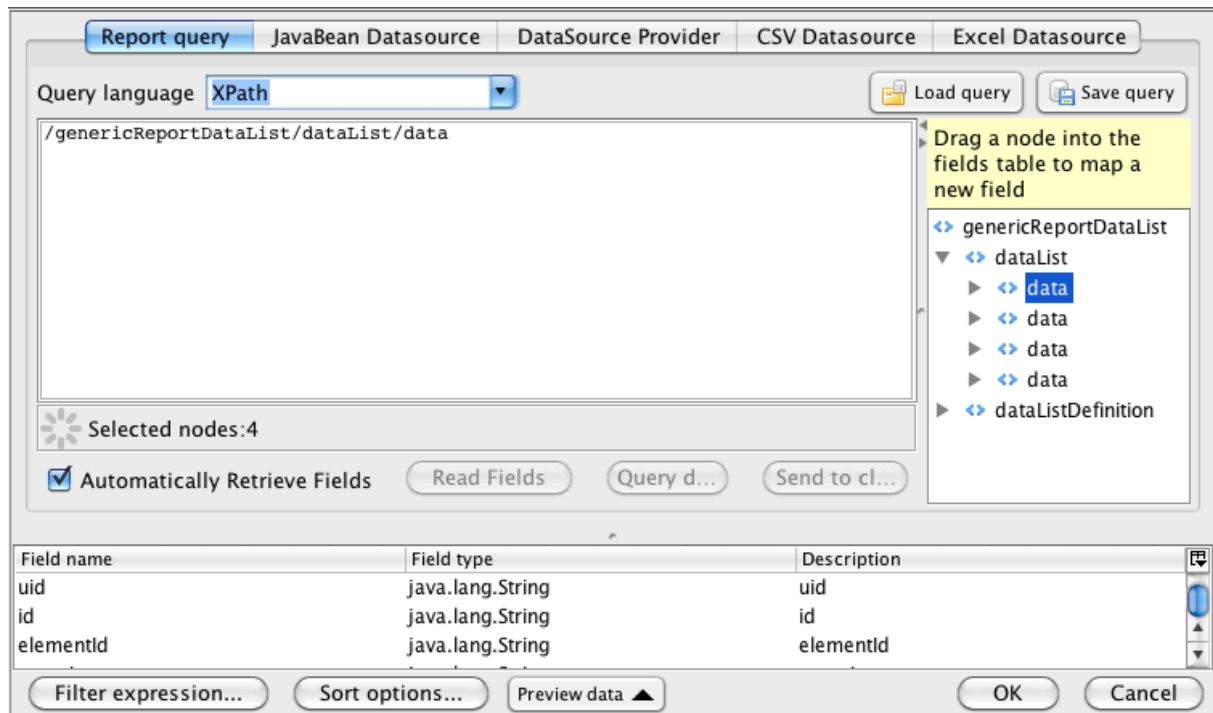


Abb. 2: Report query

Nachdem die Datenquelle eingetragen und selektiert wurde, können anschließend die Daten ausgewählt werden. Dies geschieht über die üblichen Querysprachen wie SQL oder XPath. Im Beispiel in Abb. 2 wurde eine XML Datenquelle ausgewählt worin die Daten mittels XPath bestimmt werden. In der Quelldatei gibt es mehrere „data“ Einträge. Für jeden Datensatz wird später eine neue Zeile, Spalte oder Absatz erzeugt wird (siehe Unten).

## Erstellen des Reports

Alle im voran gegangenen Schritt geladenen Datensätze werden jetzt in der Ansicht „Report inspector“ unter „Fields“ angezeigt. Von hier könne die Einträge via Drag & Drop direkt dem Report hinzugefügt werden. Dabei gibt es im Report unterschiedliche Bereiche welchen die Informationen hinzugefügt werden können:

- **Title** – Dieser Bereich wird pro Bericht nur ein mal auf der ersten Seite dargestellt.
- **Page Header** – Dieser Bereich wird immer zu Beginn einer neuen Seite angezeigt.
- **Column Header** – Wird immer zu Beginn einer Tabelle dargestellt. Zusätzlich wird dieser Bereich auf jeder Seite wiederholt.
- **Detail 1** – Für jeden Datensatz wird dieser Bereich wiederholt. In dem Beispiel von oben würden daher vier „Zeilen“ erzeugt werden.
- **Column Footer** – Wird am Ende der Tabelle dargestellt und zusätzlich auf jeder Seite wiederholt.
- **Page Footer** – Wird am Ende jeder Seite dargestellt.
- **Summary** – Wird am ende des Dokuments angezeigt.

Neben den standardmäßig eingestellten Werten zur Sichtbarkeit einzelner Bereiche und Komponenten verfügen diese über eine „Print When Expression“. Hier kann mittels Java-Code konfiguriert werden, wann das Objekt dargestellt wird. Dabei ist darauf zu achten, dass nur ein begrenzter Funktionsumfang zur Verfügung steht. Ein häufiger Anwendungsfall für eine „Print When Expression“ ist im Zusammenspiel zwischen Title und Page Header zu finden. Der Titel ist über dem Page Header positioniert. Häufig werden in der Kopfzeile Informationen wie Kapitelname oder Datum dargestellt. Der Konflikt entsteht auf der ersten Seite, hier muss die Kopfzeile in den Titel miteingebaut werden. Wird dies nicht gemacht würde durch der Page Header (die Kopfzeile) unterhalb des Titels angezeigt werden. Aus diesem Grund muss man der „Print When Expression“ der Kopfzeile sagen, dass sie nicht auf der ersten Seite angezeigt werden soll. Dazu gibt es in JasperReports Variablen welche zur Laufzeit berechnet werden. Eine dieser Variablen ist die Seitennummer, welche dazu verwendet werden kann die erste Seite ausfindig zu machen. Die „Print When Expression“ lautet dementsprechend:

```
$V{PAGE_NUMBER} != 1
```

## Fazit

JasperReports ermöglicht eine kostengünstige und in Zusammenarbeit mit iReport eine intuitive alternative zu etablierten Berichtswerkzeugen wie BIRT oder Oracle Reports. Durch den WYSIWYG Editor in iReport werden die Einstiegshürden in die Thematik größtenteils genommen. Dennoch besitzt JasperReports genügend Tiefe um komplexe Berichte zu erzeugen. Leider wird der durchweg positive Eindruck ein wenig gedämpft. Gerade im Bereich der Kompatibilität zwischen verschiedenen Versionen treten häufig ungewollte Nebeneffekte auf. Daher muss darauf geachtet werden, dass alle Entwickler im Team die gleiche Version von iReport im Einsatz haben. Dennoch bleibt zu sagen, dass iReport auch bei Berichten über tausend Seiten und mehrschichtigen Subreports sehr stabil läuft.

## Kontaktadresse:

Dominic Weiser  
esentri consulting GmbH  
Pforzheimer Straße 132  
76275 Ettlingen

Telefon: +49 (0) 7243-354 90 0  
E-Mail: dominic.weiser@esentri.com  
Internet: www.esentri.com