

# **BI Publisher Reports in der Oracle E-Business Suite Projekt-Erfahrungsbericht**

**Frank Schoepke  
PRIMUS DELPHI GROUP  
Norderstedt**

**Schlüsselworte: BI Publisher, EBS, Erfahrungsbericht**

## **Einleitung:**

Es soll ein Überblick über die Programmierung und Einbindung von BI-Publisher Reports in die Oracle E-Business Suite R12 anhand eines Projekterfahrungsberichtes gegeben werden.  
Es sollen die Möglichkeiten und Grenzen sowie mögliche Lösungsansätze aufgezeigt werden.

Themengebiete:

Überblick Einbindung von BI Publisher Reports in EBS R12

Kopf-/Fußzeilen Formatierung

Last Page Funktion

Lösungsansatz für eine Mehrsprachigkeit für nicht installierte Sprachen

Formatierung von Zahlen in der BI Publisher Ausgabe

BI Publisher java API's

## **1. Die Projektanforderungen**

Die Anforderung war, Dokumente die an Kunden bzw. Lieferanten verschickt werden, im Firmen-Layout auszugeben und auch die Möglichkeit zu schaffen, diese Dokumente per E-Mail an den Kunden zu senden. Unser Kunde ist ein international tätiges Unternehmen mit vielen Standorten weltweit. In der E-Business Suite sind nur 2 Sprachen (US-Englisch und Französisch) installiert, aber an den Endkunden sollen die Dokumente in Landessprache erstellt werden.

## **2. Einbinden von BI Publisher Reports in die EBS R12**

Die einfachste Methode, eine mit Oracle Reports erstellte Auswertung als BI Publisher Report einzubinden ist, dessen Ausgabeformat in XML zu ändern. Nach dieser Änderung erzeugt Oracle Reports die Ausgabe im XML-Format. Mit Hilfe des BI Publisher Desktops (dies ist ein Add-In für Microsoft Word) und einer XML-Ausgabe des Hintergrund-Programmes kann danach sehr einfach ein erstes Template für den BI Publisher erstellt werden.

Mit der XML Publisher - Administrator Zuständigkeit muss bei den anschließenden Einrichtungsschritten beachtet werden, dass der Code der Datendefinition dem Code des Hintergrund-Programmes entspricht. Nachdem das Template definiert und hochgeladen wurde, wird die Ausgabe des Oracle Reports durch den Ausgabenachverarbeitungs-Hintergrundmanager formatiert und als Ausgabe des Prozesses angezeigt. Zur Fehleranalyse kann das von Oracle Reports erzeugte XML über Diagnose -> XML Anzeigen aufgerufen werden.

Der Vorteil dieser Methode ist, dass Oracle Reports nach der Änderung des Ausgabeformats, die Prozessausgabe automatisch in ein XML-Format ausgibt, das vom BI Publisher gelesen werden kann. Sofern keine zusätzlichen Daten für den Ausdruck benötigt werden, muss nur noch das Layout mit Word definiert werden, wobei der Teufel im Detail steckt.

### 3. Einbinden von Kunden-Logos in die Kopf-/Fußzeile

Bei der Erstellung von Word-Templates unterliegen die Kopf- und Fußzeilen gewissen Einschränkungen. In die Kopf-/Fußzeilen können keine Formularfelder eingefügt werden. Das bedeutet, dass hier keine Datenfelder des XML direkt ausgegeben werden können. Dies kann aber einfach umgangen werden, indem man die auszugebenden Daten in Templates (so etwas wie Unterprogramme innerhalb eines BI Publisher Templates) kapselt und in den Kopf-/Fußzeilen mit dem Befehl `<?call-template:template_name?>` das Template aufruft.

### 4. Einbinden eines Endkunden-Logos in einen Ausdruck

Für den Ausdruck eines Angebots galt die Anforderung, das Firmen-Logo des Kunden im Kopf mit anzudrucken. Über eine OAF-Personalisierung wurde die Möglichkeit geschaffen, eine Datei (jpg) an den Kundenstamm anzuhängen.

In einen BI Publisher Report kann ein extern abgelegtes Bild mit dem Befehl `<fo:external-graphic src="{URL_TO_FILE}"/>` eingebunden werden. Da die Dateien nicht im Dateisystem abgelegt werden sollten, muss das Bild aus der Datenbank geladen werden. Hierfür stellt Oracle eine Funktion zur Verfügung, die die Url zum Dateianhang ermittelt.

Beispiel:

```
SELECT pk1_value party_number,
       fnd_gfm.construct_download_URL(
           fnd_web_config.gfm_agent,
           fnddoc.media_id) url_customer_logo
FROM   fnd_attached_documents fndattdoc
       ,fnd_documents_vl fnddoc
WHERE  1=1
AND    fndattdoc.entity_name = 'XXCUS_AR_CUSTOMER_IMAGE'
AND    fnddoc.document_id    = fndattdoc.document_id
AND    fnddoc.category_id IN (SELECT category_id
                              FROM   fnd_document_categories_vl
                              WHERE  user_name = 'XXCUS_LOGO'
                              )
```

Dieses Select liefert eine URL, die zur Laufzeit der Session zum gespeicherten Bild zeigt. Die Spalte `pk1_value` beinhaltet den Key zum Dateianhang, der in der OAF-Personalisierung angegeben wurde (in unserem Fall die `PARTY_NUMBER`).

### 5. Einbinden von Endsummen (LastPage Funktion)

Um auf der letzten Seite unten Endsummen anzudrucken, gibt es die BI-Publisher Funktion `<?start@last-page-first:body?><?end body?>`. Dieser Funktion soll laut Beschreibung ein

„Abschnittswechsel (nächste Seite)“ vorangehen. Alles was auf dieser Seite vor dem Formularfeld mit der last-page-Funktion steht, wird oben auf der Seite angedruckt, alles was unter diesem Formularfeld steht, wird unten auf der letzten Seite angedruckt. Der Zusatz „-first“ bewirkt, dass dieser Abschnitt auch mit ausgegeben wird wenn die Ausgabe nur aus einer Seite besteht.

Man sollte nun glauben, dass danach nichts mehr ausgegeben werden kann, aber funktioniert doch. Im o.g. Projekt sollten nach den Endsummen noch zusätzliche Informationen (zum Beispiel Lieferbedingungen) angedruckt werden. Dies kann dadurch erreicht werden, dass vor der last-page-Funktion ein „Abschnittswechsel (fortlaufend)“ eingefügt wird und vor den zusätzlichen Seiten ein „Abschnittswechsel (nächste Seite)“.

## 6. Mehrsprachigkeit für nicht installierte Sprachen

Da im Kundensystem nicht alle Sprachen installiert werden sollen und es zum Teil auch größere Unterschiede in den Dokumenten der einzelnen Länder gibt, wurde eine spezielle Funktionalität für die Dokumente der einzelnen Länder implementiert.

Beim Hochladen der Templates werden die Sprache und die Region des jeweiligen Landes angegeben. Beim Starten der Reports können diese Templates ausgewählt werden. Damit der User nicht bei jedem Ausdruck das richtige Template auswählen muss, wurde eine Forms-Personalisierung implementiert. Über eine Profiloption wird gesteuert, welches Template (Kombination aus Sprache und Region) herangezogen werden soll. Bei unserem Kunden wird diese Profiloption auf Organisationsebene gepflegt, da es je Land eine Organisation gibt. Da nicht alle Dokumente manuell über die Maske gestartet werden, sind einige Dokumente (wie z.B. der Lieferschein) als PL/SQL-Prozedur in das System eingebunden, die den eigentlichen Report mit dem gewünschten Template starten.

## 7. Zahlenformatierung in BI Publisher Reports

Um Formularfelder mit Zahlen in Abhängigkeit der Profiloptionen „ICX: Numeric characters“ bzw. „XDO: Numeric characters“ korrekt darzustellen, sind einige Dinge in der BI Publisher Programmierung zu beachten.

Zunächst müssen im XML für den BI Publisher alle Zahlen im Format Dezimal-Punkt ohne Tausender- Trennzeichen ausgegeben werden. Sofern für die XML-Erstellung ein bestehendes Oracle Reports-Programm benutzt wurde, sind für die meisten Beträge zwei Datenfelder im XML vorhanden. Die Felder, deren Namen mit \_DSP enden, sollten im Template **nicht** verwendet werden, da diese Felder in Oracle Reports schon unter Beachtung der o.g. Profiloptionen formatiert wurden.

Im BI Publisher Template müssen die betreffenden Formularfelder als Typ=Zahl mit dem Zahlenformat „#,##0.00“ (also im amerikanischen Zahlenformat) eingerichtet werden. Eine Besonderheit stellen noch die BI Publisher-Funktionen für Überträge (show-carry-forward) und Vorträge (show-brought-forward) dar. Diese beiden Funktionen sind sehr speziell. Für die richtige Formatierung von Zahlen muss im Formularfeld das Format mit dem Zusatz `number-separators="{$_XDOFSEPARATORS}"` angegeben werden.

Beispiel:

```
<xdofo:inline-total name="NET">
  <xdofo:show-carry-forward name="NET"
    format="999G999D99" number-separators="{$_XDOFSEPARATORS}"/>
</xdofo:inline-total>
```

Voraussetzung für das oben geschilderte Vorgehen ist der Patch 9105394 (für R 12.1). Dieser Patch bewirkt, dass der Hintergrundprozess des BI-Publisher Reports immer mit NLS\_LANG = AMERICAN\_AMERICA und NLS\_NUMERIC\_CHARACTERS = '.,' läuft. Die Formatierung macht dann ausschließlich der nachgeschaltete Ausgabenachverarbeitungs-Hintergrundmanager des BI Publishers.

## **8. BI Publisher Java APIs**

Mit dem BI Publisher werden von Oracle einige Java APIs zur Verfügung gestellt, mit denen spezielle Anforderungen umgesetzt werden können. Eine Anforderung im Projekt war, dass mit dem Ausdruck eines Angebotes noch zusätzliche PDF-Dokumente, die vom Anwender als Anhang am Angebot gespeichert werden, gedruckt werden sollen.

Mit den PDF Document Merger APIs besteht die Möglichkeit, mehrere PDF-Dokumente zu einem Dokument zusammenzufügen, ggf. noch ein Wasserzeichen (Grafik oder Text) einzufügen und im zusammen gefügten Dokument die Seite Nummerierung zu aktualisieren. Für die Umsetzung solcher Anforderungen sind Programmierkenntnisse in Java erforderlich.

### **Kontaktadresse:**

**Frank Schoepke**  
**PRIMUS DELPHI GROUP**  
Am Hange 27  
D-22844 Norderstedt

Telefon: +49 (0) 40-284186 55  
Fax: +49 (0) 40-284186 90  
E-Mail: [f.schoepke@primus-delphi-group.com](mailto:f.schoepke@primus-delphi-group.com)  
Internet: [www.primus-delphi-group.com](http://www.primus-delphi-group.com)