

BI Publisher-Praxis: Komplexe Reports und Apex-Dokugenerator

**Frank Effenberger, Janine Lehmann
Robotron Datenbank-Software GmbH
Dresden**

Schlüsselworte

Apex, BI Publisher, RTF Template, Publisher, XML.

Einleitung

Unternehmen mit hoher Managementaktivität haben stets einen Bedarf an standardisierten und teil- bzw. vollautomatisiert erstellten Berichten, um so eine Vergleichbarkeit und Kontrolle ihres Unternehmens bei möglichst wenig Zeitaufwand zu realisieren. Die öffentliche Verwaltung hat dabei besonders hohe Ansprüche an die Berichtserstellung.

Die Praxis der vergangenen Jahre zeigte, dass die Weboberfläche des Oracle Business Intelligence Publishers 11g (**BIP**) unzureichend ist, um die gewünschten, komplexen Berichtsszenarien abzubilden. Aus diesem Grund ist es notwendig, das BIP-Addin für Microsoft Word zu verwenden und detaillierte Templates im Rich-Text-Format (**RTF**) zu erstellen.

Dieser Vortrag stellt die Herausforderungen bei der Erstellung von komplexen Reports mit dem BIP-Addin für Microsoft Word dar und zeigt konkrete Lösungen aus der Erfahrungen in drei Großprojekten der öffentlichen Verwaltung. Der Fokus liegt dabei auf einer praxisorientierten Darstellung der korrekten Verwendung von XSL-FO und XSLT in RTF-Templates, um dynamische Inhaltverzeichnisse und Layoutanforderungen an Tabellen unter Beachtung komplexer Bedingungen umzusetzen.

Weiterhin wird die Möglichkeit der automatischen Generierung eines Anwenderhandbuches für eine Apex-Anwendung vorgestellt, in der auf Basis der Hilfetexte eine Kopplung mit dem BIP erreicht wurde. Durch die Verbindung des BIP mit Apex über Webservices ist es so möglich, bei Änderungen an der Anwendung sofort eine valide Dokumentation zu erstellen, die während der Entwicklung und nach der Installation beim Kunden nachhaltige Anwendung findet.

Komplexe Reports am Fallbeispiel von Großprojekten der öffentlichen Verwaltung

Aufgrund der stetig steigenden Datenmengen in Unternehmen und den damit verbundenen Anforderungen an die Datenauswertung ist das Reporting in Bezug auf Effizienz und Qualität eine entscheidende Komponente. Die Verwendung des BI Publishers im Rahmen von spezifischen Berichtsanforderungen in der öffentlichen Verwaltung bietet eine hohe Flexibilität für die Erstellung bei gleichzeitig sinkendem Aufwand. Dies gilt es zu untersuchen.

Ein besonderes Hauptaugenmerk liegt auf der Formular- und Dokumentenerstellung. Hier besteht die Besonderheit in der Einheitlichkeit, Übersichtlichkeit und Exaktheit von Berichten. Sie werden beispielsweise als Vorlagen oder zur Weitergabe für unterschiedliche Interessengruppen verwendet. Bei Dokumenten ist zusätzlich noch auf eine allgemeingültige Struktur von Berichten hinsichtlich Inhaltsverzeichnis, Seitenangaben, Verweisen oder Diagrammen zu achten. Der wesentliche Punkt liegt aber auf den Daten für die zu erstellenden umfangreichen Dokumente oder detaillierten Formulare. Damit sind Design und Datenkomplexität die wesentlichen Herausforderungen für ein Berichtstool.

Der BI Publisher dient zur Umsetzung dieser Anforderungen. Aus den Erfahrungen laufender Großprojekte lassen sich Vorgehensweisen und Handlungsstrategien ableiten. Zunächst erfolgt die umfangreiche Datenmodellierung, die als Grundlage für viele unterschiedliche Berichte dient. Somit kann ein Datenmodell für mehrere Layouts verwendet werden. Es sollten bereits die wesentlichen Berechnungen und Zusammenhänge im Datenmodell abgebildet werden, um die Komplexität bei der Layouterstellung zu mindern. Für die Erstellung komplexer RTF-Templates können zusätzlich Word-Mittel, Funktionen und Variablen genutzt werden. Aufgrund dessen ist das BIP-Addin dem Layout-Editor vorzuziehen.

Als hilfreiches Mittel bei der RTF-Erstellung dienen hierbei SQL- und XSL-Funktionen, die in BI Publisher Tags eingebunden werden. Die Syntax zur Interpretation durch den BI Publisher lautet „xdofx“ bzw. „xdoxslt“. Dadurch wird die Berichtserstellung für Endanwender ohne weitreichende XML-Kenntnisse erleichtert. Bei der späteren Ausführung des Berichtes übersetzt der BI Publisher intern die Tags der Datenfelder in die standardmäßigen SQL- bzw. XSL-Codes. XSLT lässt sich jedoch auch ohne Tags anwenden. Dies ist beispielsweise über XSL-Templates, die in RTF-Templates referenziert werden, zu realisieren. Des Weiteren lassen sich Erweiterungen der Darstellungsmöglichkeiten von Diagrammen in Berichten oder die Verwendung von Variablen nur mit XSLT umsetzen. Über den BI Publisher gibt es im Anschluss vielfältige Ausgabeformate und Weiterleitungsmöglichkeiten für die Berichte. Damit lassen sich diese in umfangreichem Maße präsentieren.

Apex-Dokugenerator

In einem Software-Projekt, welches auch nach der initialen Produktübergabe weiterhin gewartet, entwickelt und damit vom Auftragnehmer unterstützt wird, finden in Absprache mit dem Auftraggeber regelmäßig Änderungen statt. Dadurch sind damals erstellte Anwenderhandbücher und Software-Dokumentationen meist schnell veraltet und haben nur noch eine bedingte Gültigkeit.

In APEX-Projekten gibt es die Möglichkeit, Hilfetexte zu den jeweiligen Seiteninhalten zu generieren und zusätzlich in einer Datenbank abzulegen. Die Idee des Apex-Dokugenerators liegt darin, diese Hilfetexte zu nutzen, um eine aktuelle und valide Anwenderdokumentation für den Endbenutzer bereitzustellen. Zeitgleich kann dieses Tool auch während der verschiedenen Entwicklungsphasen und Meilensteine vor der ersten Übergabe des Produktes genutzt werden, um intern die Verständlichkeit und Einheitlichkeit der Hilfetexte zu kontrollieren. Weiterhin findet dadurch eine enge Kopplung der Anwenderdokumentation an die Entwicklung statt und Änderungen müssen nur an einer zentralen Stelle anstatt an zwei Orten (Hilfetext und das Dokument) gepflegt werden. Dies ist vor allem bei umfangreichen Softwareprojekten von Vorteil.

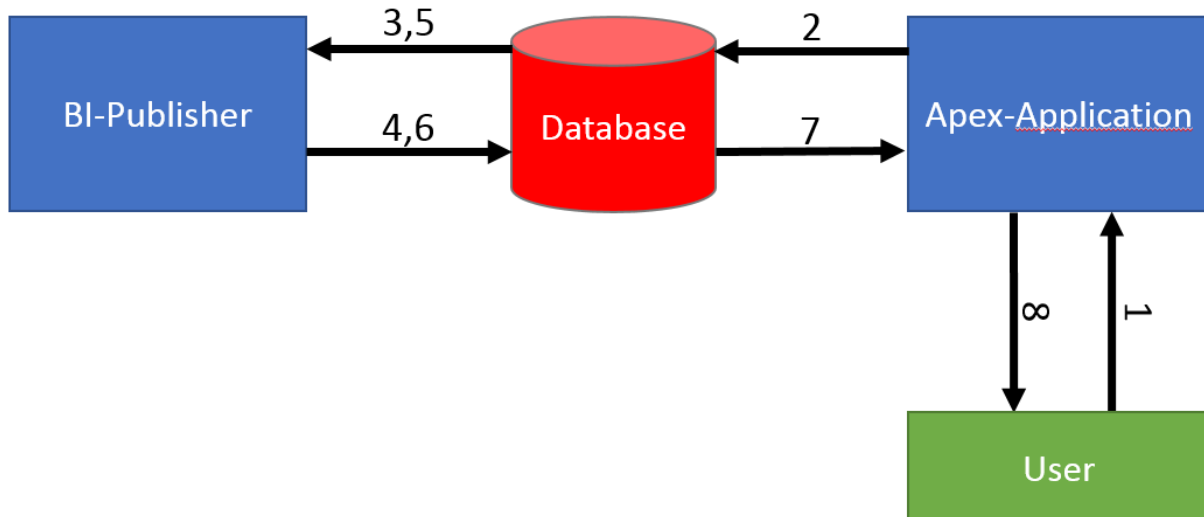


Abbildung 1: Architekturmuster für den Apex-Dokugenerator.

Abbildung 1 zeigt die Grobarchitektur der Anwendung, bestehend aus einer Oracle Datenbank mit installiertem Apex und dem BI Publisher. Die angegebenen Zahlen stellen die Schritte dar, die eine aktuell gültige Erstellung einer Anwendungsdokumentation erklären.

Schritt 1 stellt eine Dokumentenanfrage des Nutzers dar, welche in der Regel durch einen Button in der APEX-Anwendung realisiert wird. Die Betätigung des Buttons veranlasst die APEX-Anwendung dazu, eine Prozedur mit integrierter Funktion aufzurufen (Schritt 2). Der nächste Schritt beschreibt den

innerhalb der Funktion angestoßenen Webservice-Request an den BI Publisher, mit dem die aktuelle Anwenderdokumentation als PDF bereitgestellt werden soll.

Der BI Publisher führt im vierten Schritt eine SQL-Query gegen die Datenbank aus und bekommt im fünften Schritt die aktuellen Hilfetexte in einer zuvor bereitgestellten Datenstruktur als Ergebnis der Abfrage zurück. Das PDF wird mit den erhaltenen Daten nun finalisiert und als Webservice-Response im sechsten Schritt zurück an die Datenbankfunktion gegeben. Innerhalb der Prozedur mit integrierter Funktion wird nun ein Downloadlink für das PDF bereitgestellt. Der User hat nun die Möglichkeit, die aktuelle Anwenderdokumentation herunterzuladen und auf seinem Arbeitsgerät zu speichern.

Aus Sicht der Datenmodellierung gibt es drei Tabellen und zwei Views, welche die Datengrundlage für den BI Publisher bilden (siehe **Abbildung 2**). Die Tabelle Kapitel stellt eine hierarchische Definition der verschiedenen Kapitel der Anwenderdokumentation dar. Im BI Publisher stellt die Tabelle KAPITEL_CBP einen View auf die Tabelle Kapitel mit einem Connect-By-Prior-Befehl (**CBP**) dar. Da der BI Publisher den CBP nicht alleine ausführen kann, musste dieser Umweg über die View gegangen werden.

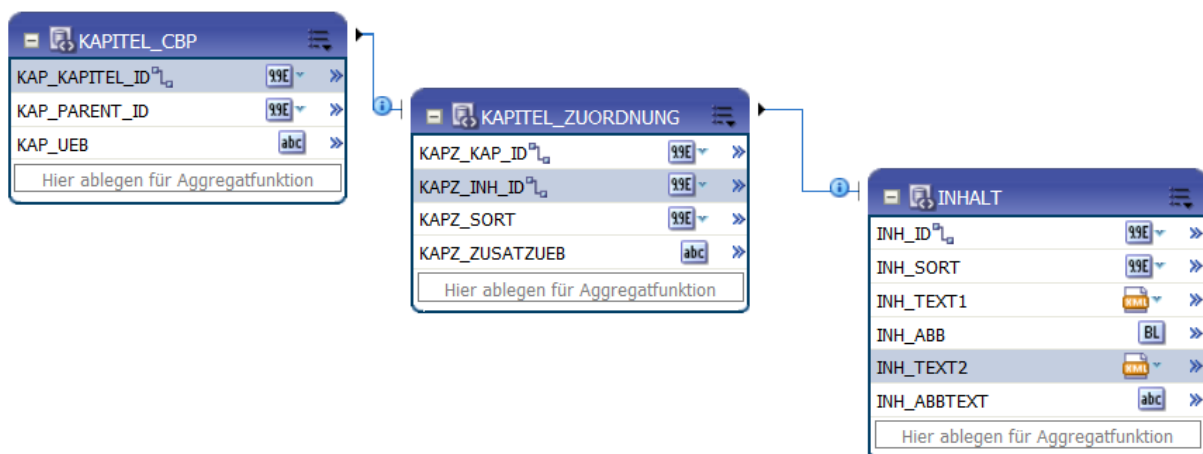


Abbildung 2: Zuordnung der Tabellenstruktur im BI Publisher.

Jedes Kapitel hat eine individuelle Zuordnung zu verschiedenen Inhalten. Diese Beziehung wird über die Tabellen KAPITEL_ZUORDNUNG und INHALT dargestellt. Dieses generische Vorgehen erlaubt es, dynamisch an jeder Position der Kapitel dynamische Inhalte in Form von Grafiken oder Texten zu hinterlegen. Dabei wird auch HTML- und CSS berücksichtigt, welche z. B. in der Apex-Anwendung verwendet werden, um ein einheitliches Layout bereitzustellen. Auf Seiten des BI Publishers besteht die Tabelle INHALT aus einer View, in der Inhalte der Texte mit einer CSS der Apex-Anwendung verknüpft werden, damit Tabelleninhalte wie in der Anwendung dargestellt werden.

Auf Seiten des RTF-Templates ist es aufgrund der dynamischen Erstellung der Anwenderdokumentation nötig, das Deckblatt individuell je nach Kunde zu gestalten bzw. in dem Template anzupassen. Die Funktionalität des Inhaltsverzeichnisses muss mithilfe des BIP-Addins nachgebaut werden, ebenso wie automatisierte Referenzen in den Kopf- und Fußzeilen.

Kontaktadresse:

Frank Effenberger
Robotron Datenbank-Software GmbH
Stuttgarter Straße 29
01189 Dresden
Telefon: +49 351 25859-2512
Fax: +49 351 25859-3699
E-Mail frank.effenberger@robotron.de
Internet: www.robotron.de

Janine Lehmann
Robotron Datenbank-Software GmbH
Stuttgarter Straße 29
01189 Dresden
Telefon: +49 351 25859-2432
Fax: +49 351 25859-3699
E-Mail janine.lehmann@robotron.de
Internet: www.robotron.de