robotron®

Erfahrungsbericht - Oracle ADF 11g im produktiven Einsatz

Martin Kunze, 17.11.2011

Agenda

robotron®

- Robotron
- Grundkonzepte ADF vs. Forms
- Verwendeter ADF–Technologiestack
- PL/SQL-Wiederverwendung in ADF
  - Connection-Pooling, Transaktionskontrolle, PL/SQL-Variablen
- DB-Sperrverhalten
- Erfahrungsbericht (Pro und Contra)
- Fazit

Robotron Datenbank-Software GmbH

robotron®

- Hauptstandort: Dresden
- Niederlassungen: Schweiz, Österreich, Prag, Russland
- ~ 280 Mitarbeiter

- (Haupt)-Technologien
  - Oracle DB und Oracle Forms
  - Java
- (Neben)-Technologien
  - Eclipse RCP, GWT/Spring/Hibernate, .NET, eXForms, …
- Produkte
  - Software für den Energiemarkt (EDM), …

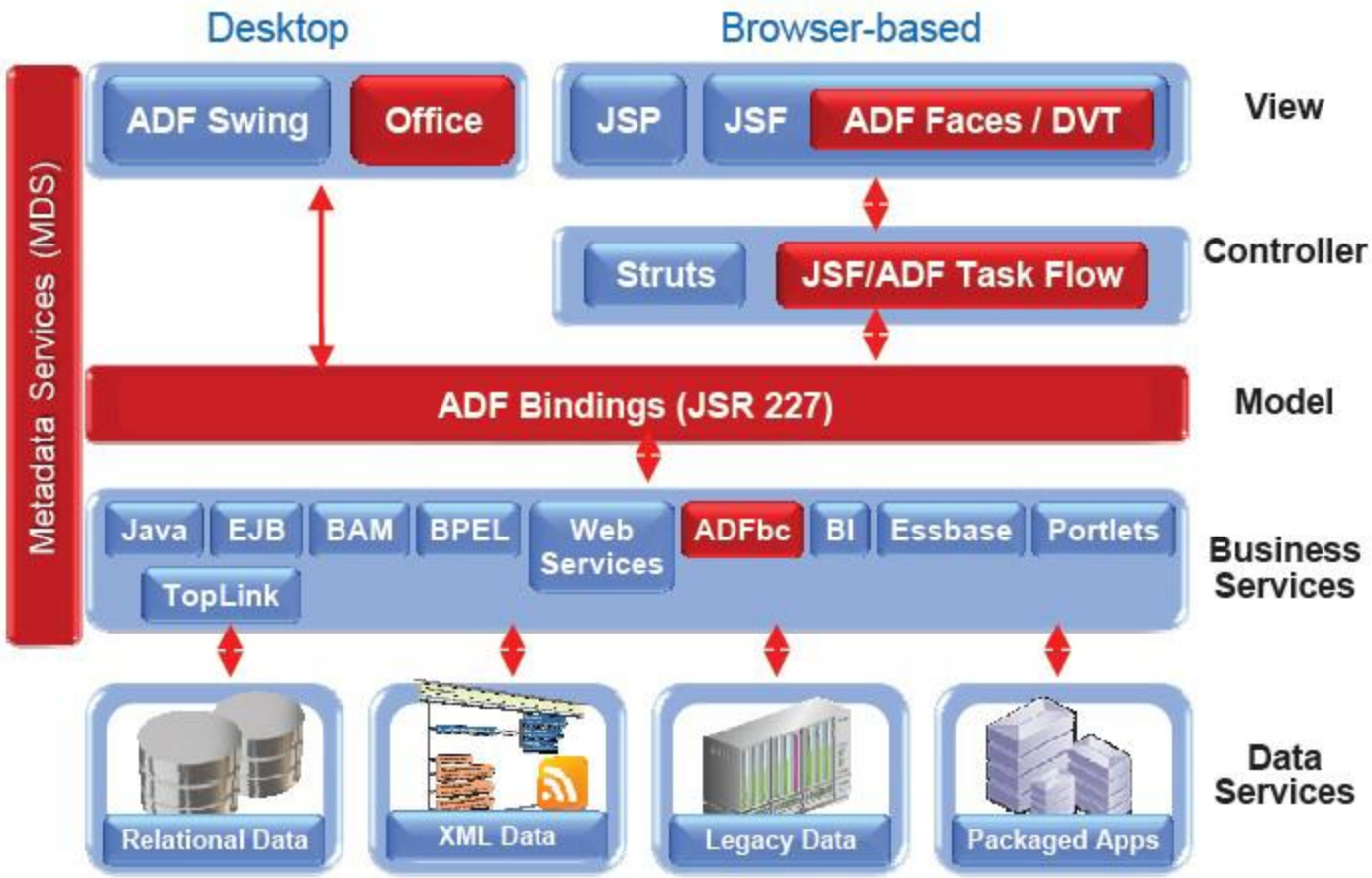
Grundkonzepte

robotron®

- ADF ist Technologienachfolger von Oracle Forms
- Unterschiedliche Architektur zwischen Forms und ADF
  - Forms
    - Client Server - Architektur
    - Kein DB-Connection-Pooling (1:1 Verbindung Client und DB)
    - Security meist über DB realisiert
  - ADF
    - Web - Architektur
    - Exzessiver Gebrauch von DB-Connection-Pooling
    - Dynamic JDBC Credentials nur bedingt zu empfehlen
    - Security in der Mittelschicht realisiert

Verwendeter ADF-Technologiestack

robotron®



PL/SQL

robotron®

- Restriktionen
  - Mittelschichttransaktionen sind keiner dedizierten Datenbanktransaktion zugeordnet
    - Aktuelle Nutzer kann der DB in den Sessionkontext übergeben werden („prepareSession“-Methode AM)
  - Transaktionssteuerung in der Mittelschicht
    - Vorsicht bei COMMIT/ROLLBACK in DB-Prozeduren
    - DB-Prozeduren haben keine Kenntnis über Änderung in der Mittelschicht („postChanges“)
  - Keine sinnvolle Verwendung für PL/SQL-Variablen
  - Unterschiedliche Security-Konzepte

PL/SQL - Nutzung

robotron®

- Direkte Nutzung von PL/SQL mittels JDBC
- View Object auf Basis von PL/SQL
  - Nested Table auf Basis von SQL-Objekttyp in VO:
  - SELECT * FROM TABLE(PKG.fnc(:param1))
- Entity Object auf Basis von PL/SQL
  - doDML-Methode der oracle.jbo.server.EntityImpl überschreiben
  - Direkter Aufruf von PL/SQL mittels JDBC
- „indirekte“ Verwendung in DB-Views / Triggern

Sperrverhalten

robotron®

- optimistisch
  - für Web-Anwendung empfohlen
  - Vor Speicherung werden betroffene Spalten auf Abweichungen zum Initialstand überprüft
  - Information falls anderer Nutzer Datensatz geändert hat
  - Risiko: erneutes Speichern kann andere Änderung überschreiben
- pessimistisch
  - lockt, wenn Datensatz vom Nutzer editiert wird
  - Default: Sperrung des Datensatzes per “FOR UPDATE”
  - Sperre mittels DB realisiert, Sperre kann auch von anderen Anwendungen vollzogen werden

Sperrverhalten

robotron®

- Empfehlung:
  - Optimistisches Locking i.V.m. Change-Indicator-Spalte
  - „kritische“ Daten direkt bei der Anzeige sperren und nicht erst beim Editieren

vgl: Immediate Row Level Lock Management for ADF 11g Transactional Applications by Andrejus Baranovskis

Erfahrungsbericht

robotron®

- Migration von Forms nach ADF (Teil 1)
  - Aufgrund der unterschiedlichen Architekturen existiert kein offizieller Migrationspfad
  - Migrationstools
    - scheitern an komplexen „Robotron-Masken“
    - erzeugter Code weicht oft von „normalen“ ADF-Code ab
    - Schichtenübergreifende Erzeugung von Code (ADF BC – ADF Taskflows und ADF Pages) ist schwer zu automatisieren
    - Abhängigkeit zu weiteren Toolhersteller

Erfahrungsbericht

robotron®

- Migration von Forms nach ADF (Teil 2)
  - Neuentwicklung inkl. Anpassung ans Web für kleine/mittlere Anwendung am sinnvollsten
  - Für umfangreiche Anwendung (z.B. EDM-System) ist ADF nicht immer die erste Wahl, wenn:
    - die bestehende Anwendung keine Web-Anwendung ist
    - eine 1:1-Beziehung zwischen Client und DB zwingend erforderlich ist
    - die „gesamte“ Logik in der Datenbank abgelegt ist und eine Anpassung wirtschaftlich nicht tragbar ist
    - auf Kundensystemen keinen Einfluss in Bezug zur Browserwahl ausgeübt werden kann

Erfahrungsbericht

robotron®

- Integration Forms mit ADF
  - Untersuchung von „OraFormsFaces“
    - Ist grundsätzlich für die Einbindung von Forms-Masken verwendbar
    - zusätzliche Lizenz- und Wartungsgebühren
    - Nutzung unterschiedliche DB-Verbindungen
    - Konsistente Anwendungentwicklung mit „Mischform“ nicht möglich
  - Kommunikation zwischen Forms und ADF möglichst über DB realisieren

Erfahrungsbericht - Risiken

robotron®

- Browserhölle existiert weiterhin
- Teilweise erhebliche Lizenzkosten bei Kombination von ADF und Forms (Weblogic Suite)
- Kombination von ADF und Forms auf einem Server ist bei Updates/Upgrades problematisch
- ADF-Updates bringen oft mehr Änderungen mit als Forms-Updates (Weiterbildung und Anpassung der Anwendung an neue Gegebenheiten)
- Entwicklungsproduktivität ist im Vergleich zu Forms geringer
- Einarbeitungsaufwand recht hoch (JSF-Lebenszyklus, …), Entwicklungsumgebung oft ungewohnt

Erfahrungsbericht - Chancen

robotron®

- Sprachfeatures von Java (z. B. Vererbung)
- breite Wizard-Unterstützung vom JDeveloper
- Wiederverwendbarkeit von ADF (Taskflows, Declarative Components, ADF BC Elemente als ADF Library, …)
- Zunehmend Nutzung modernere Browser bei Kunden (Desupport Win XP)
- Verbesserungen an ADF und JDeveloper

sorgen für eine ständig steigende Produktivität!

Erfahrungsbericht – Optimale ADF-Anwendung

robotron®

- Don‘t fight against the framework
- „optimale“ Datenmodell
  - Nur 1 ID-Spalte und 1 Change-Inidcator-Spalte je Tabelle
  - PL/SQL ist „webfähig“, keine PL/SQL-Variablen
  - DML-Operationen setzen direkt auf den DB-Tabellen auf
- „optimale“ Anwendung
  - Nutzung des ADF-Standards AM-/DB-Connection-Pooling, ADF-Security, ADF-Taskflows
  - Features der DB nutzen, komplexe Abfragen in Views auslagern, …
- „optimale“ Entwickler
  - Breite Kenntnisse in Java EE (JSF, Persistenzschicht), Java, ADF

Fazit

robotron®

- ADF/JDeveloper sind für die Entwicklung von Webanwendungen bestens geeignet.
- Für Neuentwicklungen kann ADF, unter Beachtung der Restriktionen bezüglich Webfähigkeit, bedenkenlos genutzt werden.
- Die „Transformation“ einer bestehende (sehr umfangreichen) Forms-Anwendungen nach ADF sollte jedoch gründlich evaluiert und hinterfragt werden!

Fragen?

robotron®