

Documenten in en uit Apex

Hoe om te gaan met documenten en
Oracle Apex

Dick Dral, Detora

Even voorstellen

- Oracle sinds 1988 (Ministerie van LNV)
- HTML sinds 1995
- Oracle Webtoolkit sinds 1998
- Apex sinds 2007
- Sinds 2008 zzp, www.detora.nl
- Blog dickdral.blogspot.nl
- Sinds kort lid van Smart4Apex, www.smart4apex.nl



Inhoud

- Waarom documenten?
- Opslag en levering van documenten
- Aanleveren van gegevens met werkbladen
- Aanmaken van tekstdocumenten
- Aanmaken van werkbladen

Document

- Wikipedia: Een **document** is een verzameling gegevens vastgelegd op een gegevensdrager
- I.t.t. database inhoud ongestructureerd
- Electronische documenten
 - Tekstdocumenten
 - Werkbladen
 - Presentaties
 - Afbeeldingen
 - Gegevens (XML, CSV etc.)

Gebruik documenten

- Juridisch: factuur, contract, archivering
- Communicatie: wachtwoordbrief, aanmaning, handleiding
- Gegevensoverdracht: pakbon, upload CSV
- Kennisoverdracht: presentaties, webpagina's
- Vastleggen situatie: (elektronisch) dossier

Opslag van documenten

- Laden van documenten
 - Via Apex (filebrowse item) => **demo**
 - SQL Loader
 - Bij grote aantallen in ZIP
- Opslag:
 - In database (opslagruimte, aparte documententabel)
 - Op filesystem of internet (plaatsonafhankelijk, consistentie)
- Beschikbaar stellen
 - Via link in Apex, geen directe links bij gevoelige informatie. Zorg voor goed mimetype

Ruimtebeslag in de database

Aantal objecten mogelijk in Oracle XE (max 4Gb)

Type	Omvang	Max aantal in XE
DB rij	200 b	20.000.000
Word	50 Kb	80.000
PDF	300 Kb	13.000
Afbeelding	200-1500 Kb	2.500-20.000

Opslag documenten in aparte tabel

Tabel Persoon

ID	Naam	Geb.datum	Paspoort
1	Bakker	03-02-1957	BLOB
2	De Boer	15-10-1968	BLOB

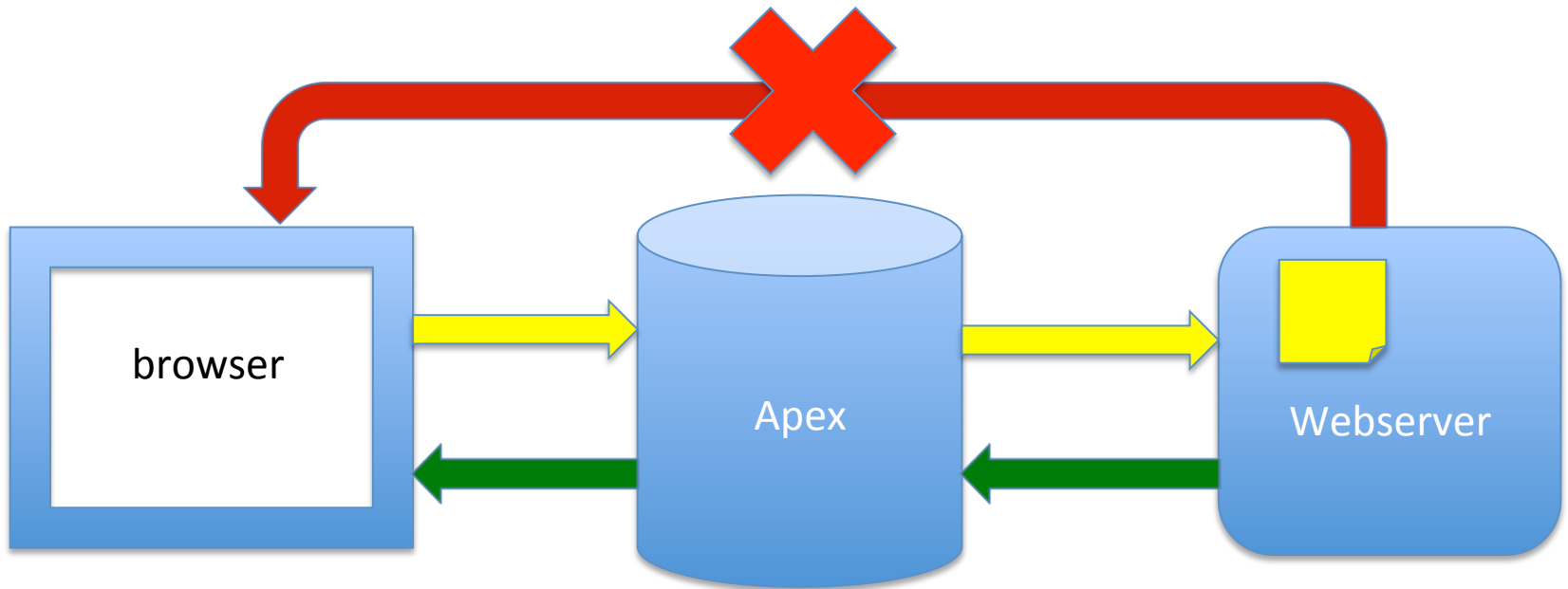
10.000 rijen * 250 Kb = 2,5 Gb

Tabel Persoon

Tabel Paspoort

10.000 rijen * 200 b = 2 Mb

Beveiliging documenten



Vermijd ongeoorloofde toegang
door aanpassen van URL

Gegevensaanlevering

- CSV uit Excel
 - Nadelen: onzekerheid over datum en numerieke formaten => **Demo**
- Native Excel (XLS, XLSX)
 - Niet direct ondersteund door Apex/Oracle
 - Mogelijk met Apache POI of AS_XLSX
- XML
 - Lastig aan te maken of wijzigen door personen

Native Excel XLSX

- XLSX => AS_READ_XLSX
 - Gemaakt door Anton Scheffer (AMIS)
 - PL/SQL package, dus in database
 - Uitgevoerd als pipelined function:

```
SELECT * FROM table( as_read_xlsx.read  
  ( as_read_xlsx.file2blob( 'direcyory',  
    'filename' ) ) );
```
- Download: technology.amis.nl/2013/01/19/read-a-excel-xlsx-with-plsql/

Native Excel XLS

- XLS => Apache POI
 - Open Source project
 - Java classes, dus Java programmeren
 - Servlet:
 - Input: XLS
 - Output: CSV
 - Deploy op applicatieserver
 - Aanroep met utl_http vanuit Oracle
- Informatie en download: poi.apache.org

Datatypes voor grote objecten

- LONG (verouderd)
- BLOB – Binaire opslag
 - Afbeeldingen, muziek, video
 - Binaire bronbestanden (DOC, XLS,PPT)
- CLOB – Tekst
 - Teksten (RTF, XML, DOCX)
 - Werkbladen (CSV, XLSX)
 - Presentaties (PPTX)
- Als kolom in tabel of als PL/SQL variabele

Bewerkingen op LOB's

- LONG : geen bewerkingen mogelijk
- CLOB/BLOB
 - Package DBMS_LOB
 - Conversie van BLOB naar CLOB en vv.
 - Lezen/schrijven van gedeelte van LOB (READ,WRITE,SUBSTR)
 - Zoeken binnen LOB (INSTR)
 - Lezen en schrijven van BFILE's

Nuttige functies op LOB's

- BLOB naar CLOB en vv.
 - DBMS_LOB
 - Character set / diakrieten
- BLOB/CLOB naar bestand en vv.
 - UTL_FILE, Oracle Directory
- BLOB/CLOB vanuit URL
 - UTL_HTTP
- BLOB/CLOB naar browser
 - HTTP, OWA_UTIL, mimetype

Character sets

- Character sets:
 - ASCII 7-bits
 - ISO 8859-1 8-bits
 - UTF-8 multibyte
- ASCII coding mostly equal, diacritics problem in Western languages
- Output = Mojibake (character soup)

Character sets 2

- Upload als BLOB (met bepaalde charset)
- BLOB naar CLOB => meestal conversie (CLOB in charset database)
 - UTL_RAW.CAST_TO_VARCHAR2 geen CS conversie
 - DBMS_LOB.CONVERTTOCLOB wel CS conversie, charset-id is argument
- Vaak geen info over charset in file. Wel:
 - Byte Order Mark (BOM)
 - In XML, HTML, Office documenten



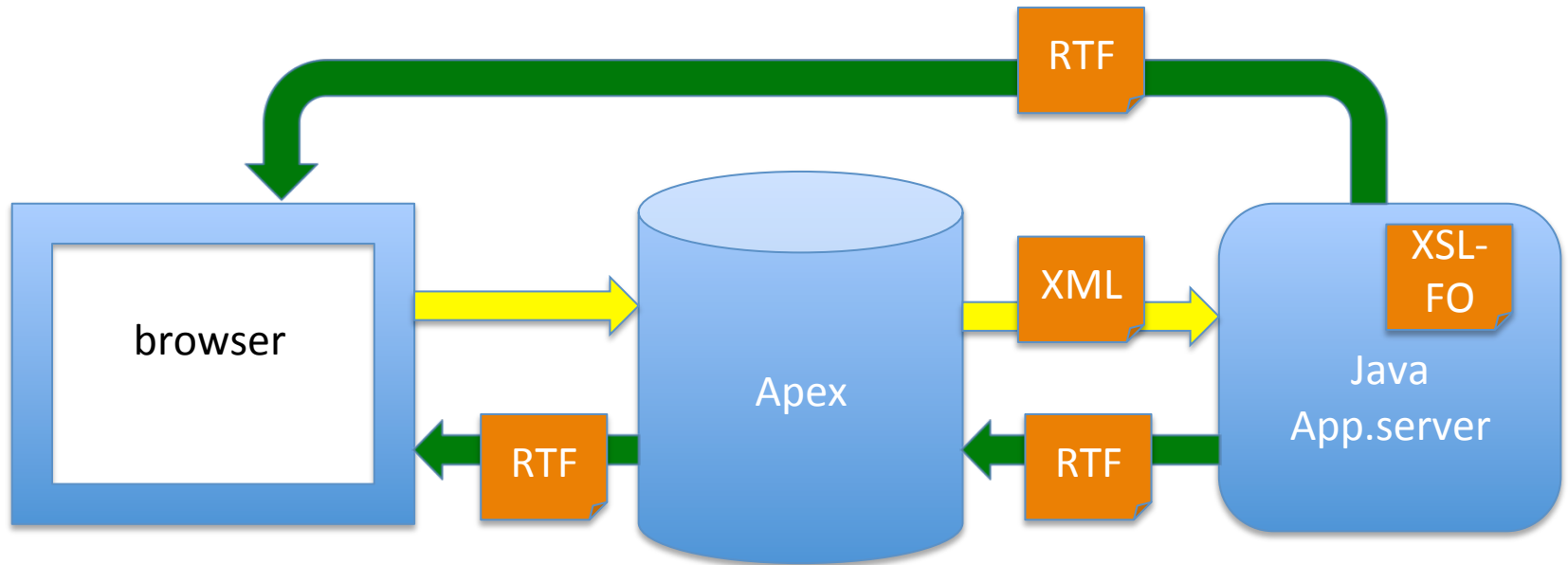
Aanmaken tekstdocumenten

- Tekstdocumenten => Word
- Formaten RTF, DOC, DOCX, PDF
- Aanpasbaar of direct afdrukbaar
 - Aanpasbaar => RTF,DOC, DOCX
 - Goed afdrukbaar of Liever niet aanpasbaar => PDF
- Architectuur
 - In database
 - Met aparte applicatieserver

BI Publisher

- Gebouwd rond XML-FO
- Veel soorten output
 - RTF, PDF, HTML, Excel, PowerPoint
- Installatie op aparte applicatieserver
 - Batch processing
 - Mailen van resultaten
- **demo**
- <http://www.oracle.com/us/solutions/business-analytics/business-intelligence/publisher/overview/index.html>

Architectuur BI Publisher



BI Publisher Servlet

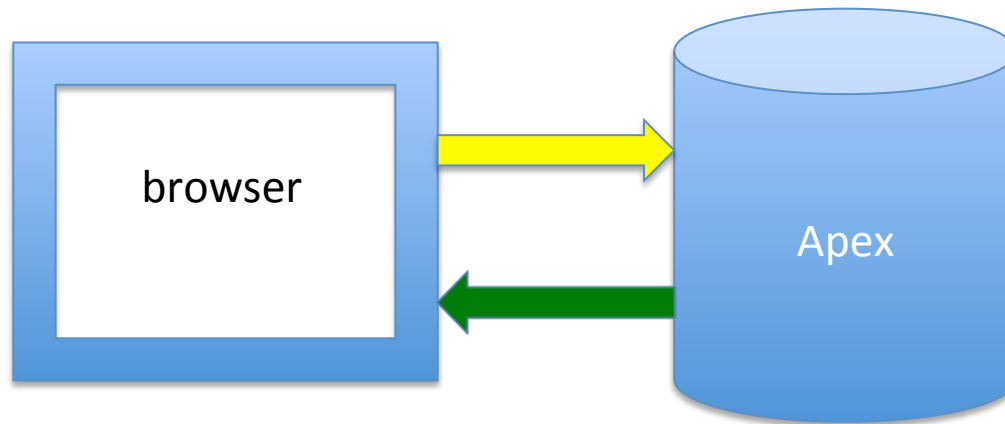
- Apache TomCat
- BI Publisher Java API
- Servlet:

```
FOPProcessor processor = new FOPProcessor();  
processor.setData(XMLDataFileName); // set XML input file  
processor.setTemplate(XSLTemplateFileName); // template  
processor.setOutput(RTFResultFileName); //set output file  
processor.setOutputFormat(FOPProcessor.FORMAT_RTTF);  
processor.generate();
```

SQLWord

- Manipulatie van RTF
- Syntax gebaseerd op PL/SQL Server Pages (PSP)
- Editen van Word document
- Compileert naar stored procedure
- Geen andere infrastructuurcomponenten
- Bruikbaar in de cloud => **demo**
- www.sqlword.nl

Architectuur SQLWord



Gevensverstrekking

- CSV
 - Makkelijk, standaard in Apex
 - Geen formattering en kolombreedtes
 - Één werkblad
- HTML
 - Wel opmaak, maar waarschuwing door Excel
 - Één werkblad
- XLS, XLSX
 - Niet standaard bij Apex/Oracle
 - Geen beperkingen

Download CSV

- Native Apex
 - Aankruisen gewenste velden
 - Geen LOV waarden
- Via eigen package
 - Op basis van query CSV output
 - Diakrieten => BOM

Download XLSX

- Native Excel output
 - Formattering van kolomkoppen
 - Breedte van kolommen goed
 - Geen character set probleem
 - Juiste datums en numbers

Download XLS

- Servlet met Apache POI
- XML input, eigen definitie
- XLS output

Downloaden vanuit Apex

- Bestand in CLOB of BLOB
- Zorg voor juiste mimetype en character set:

```
owa_util.mime_header (v_mimetype, FALSE,v_character set);  
htp.p ('Content-Disposition: attachment; filename="' || v_filename || '"');  
htp.p ('Content-length:' || dbms_lob.getlength(v_blob) );  
owa_util.http_header_close;  
wpg_docload.download_file(v_blob);
```

- Roep aan in Before Header process

Conclusie

- Houdt rekening met character sets
- Upload data liefst in Excel formaat
- Bij voorkeur simpele architectuur (geen aparte applicatieserver)

