

Generating Microsoft Word Documents from Oracle Database





Agenda

- * Office Open XML (OpenXML) Document Structure
- * Document Generation Tool
- * Good Things
- * View from the Inside - how everything works
- * Dilemmas
- * Currently Supported Functionalities
- * Document Generation Examples
- * Conclusion

- * 30 Years in IT industry
- * 20 Years in Business Analysis, Database Design and Software Development, IT Consulting
- * 13 Years Experience with Oracle Products (PL/SQL, Designer, Forms, Apex, Reports)
- * OCP Advanced PL/SQL, Forms and Java SE developer
- * www.zt-tech.eu
- * <https://www.linkedin.com/in/zoran-tica/>



Introduction



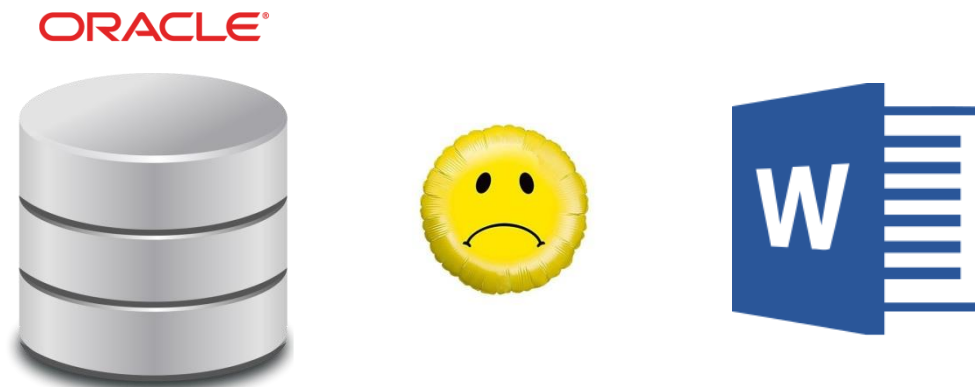
Introduction

The need to create various documents from various applications using various tools:

- * invoices
- * financial reports
- * address books
- * inventories
- * letters
- * surveys
- * and much more

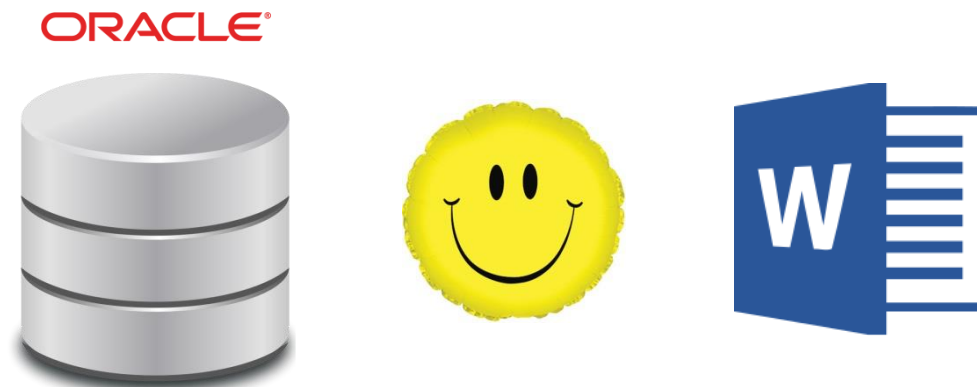
Introduction

- * Oracle officially doesn't support Microsoft Office documents generation from within database.
- * Usage of Oracle tools as BI Publisher (architecture and configuration problems, price etc.)



Introduction

- * Solution in this presentation offers possibilities to generate Microsoft Word documents from within database

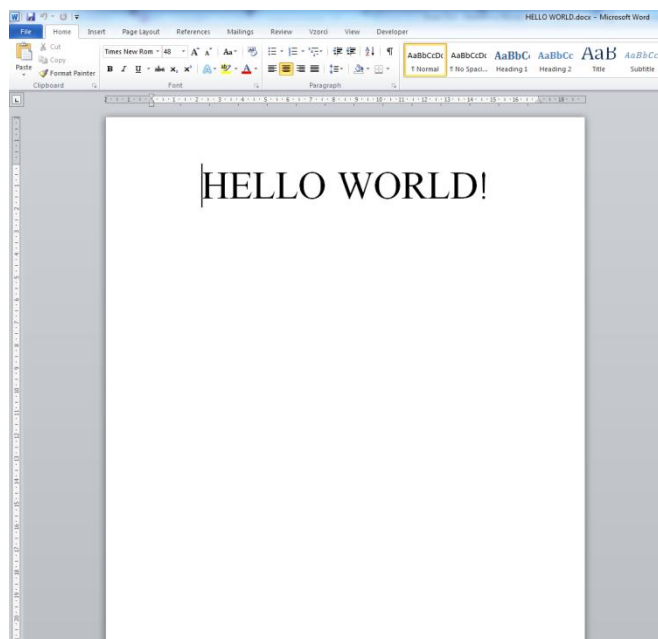


OpenXML Document Structure

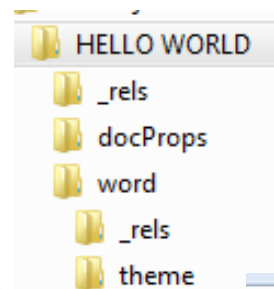


OpenXML Document Structure

Binary document; compressed (ZIP) folder structure and XML documents



folder structure



word folder with documents

Name	Type	Size
_rels	File folder	
theme	File folder	
document.xml	XML File	3 KB
fontTable.xml	XML File	2 KB
settings.xml	XML File	3 KB
styles.xml	XML File	15 KB
stylesWithEffects.xml	XML File	16 KB
webSettings.xml	XML File	1 KB

OpenXML Document Structure

Usage of some XML Documents within DOCX File

XML document	Usage
word/document.xml	Document content
word/styles.xml	Various styles used in document
docProps/core.xml	Document metadata (author, creation date...)
word/_rels/document.xml.rels	In-document relations
[Content_Types].xml	Content types used in document
word/header1.xml	Page header content
Word/numbering.xml	Numbering lists definition

OpenXML Document Structure

XML Documents Source

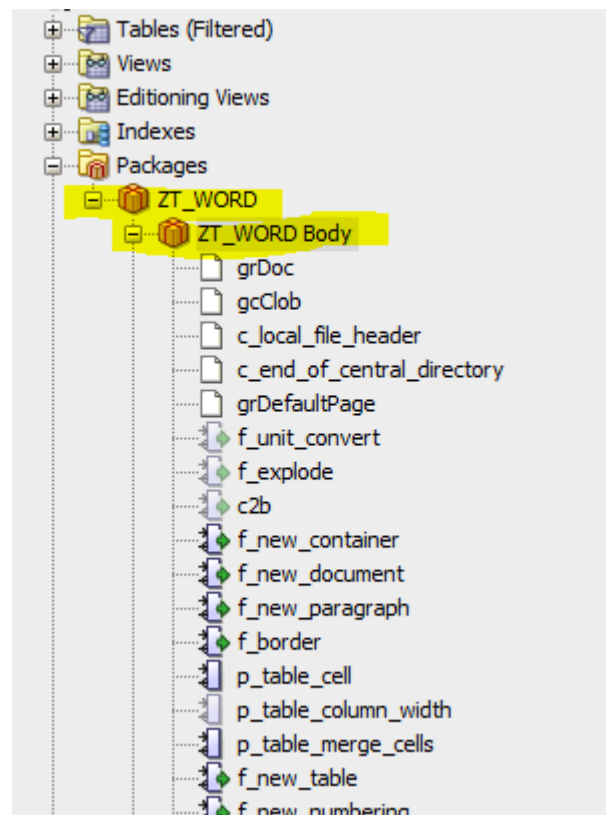
```
document.xml
1  <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2  <w:document xmlns:wpc="http://schemas.microsoft.com/office/word/2010/wordprocessingCanvas" xmlns:mc="http://
3  <w:body>
4      <w:p w:rsidR="00257B9B" w:rsidRPr="008D4D83" w:rsidRDefault="008D4D83" w:rsidP="008D4D83">
5          <w:pPr>
6              <w:jc w:val="center"/>
7              <w:rPr>
8                  <w:rFonts w:ascii="Times New Roman" w:hAnsi="Times New Roman" w:cs="Times New Roman"/>
9                  <w:sz w:val="96"/>
10                 <w:szCs w:val="96"/>
11             </w:rPr>
12         </w:pPr>
13         <w:r w:rsidRPr="008D4D83">
14             <w:rPr>
15                 <w:rFonts w:ascii="Times New Roman" w:hAnsi="Times New Roman" w:cs="Times New Roman"/>
16                 <w:sz w:val="96"/>
17                 <w:szCs w:val="96"/>
18             </w:rPr>
19             <w:t>HELLO WORLD!</w:t>
20         </w:r>
21     </w:p>
```

Document Generation Tool



Document Generation Tool

Only one PL/SQL package



Good Things



Good Things

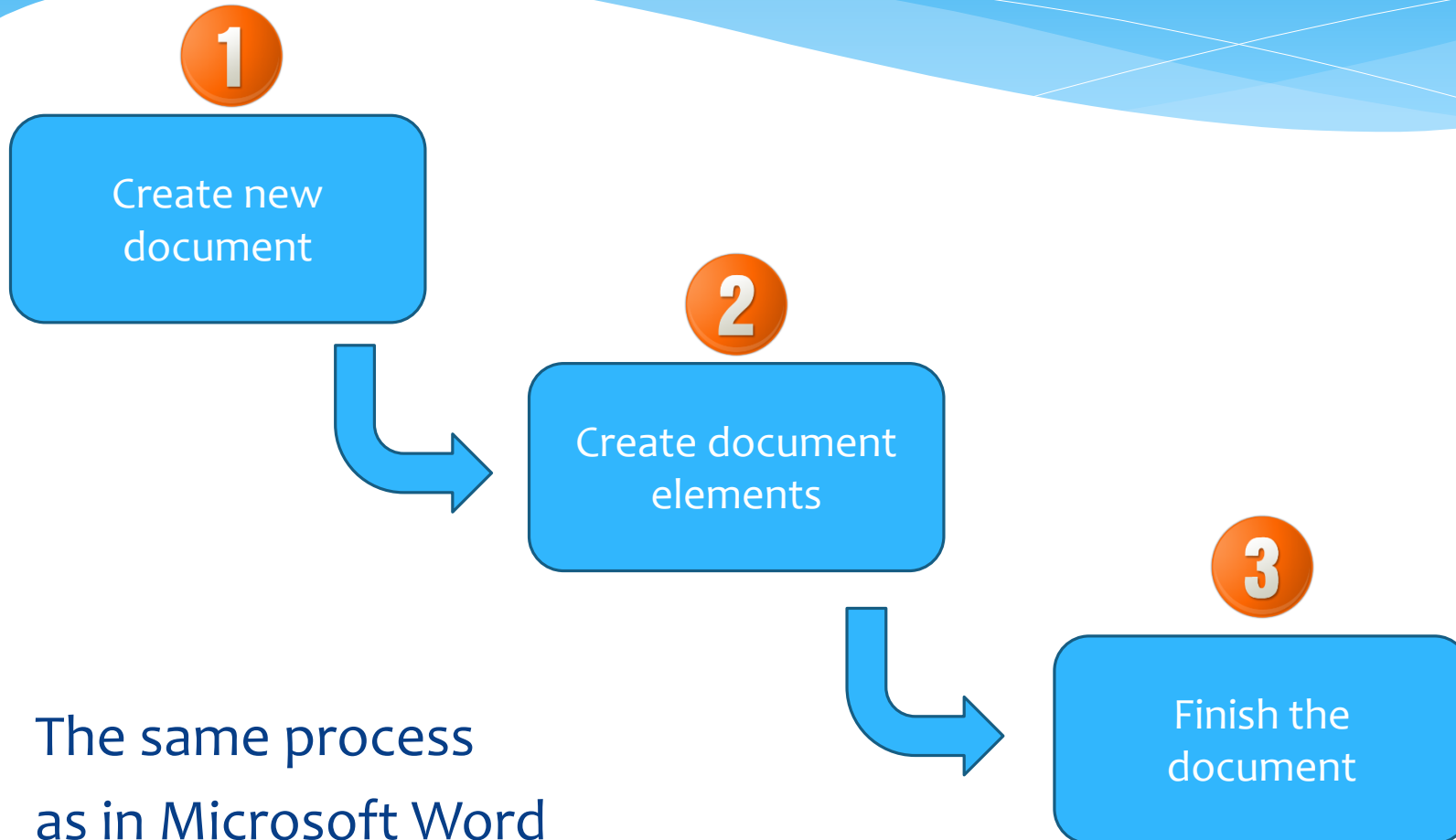


- * Everything is contained inside database
- * No need for additional tools, additional programming languages or architectural changes
- * PL/SQL developer can write code to create documents – no need for additional skills

How everything works




How everything works



The same process
as in Microsoft Word

How everything works

New document and elements creation with usage of package functions

↓  `FUNCTION f_new_document(
 p_author varchar2 default null,
 p_default_page r_page default null,
 p_unit varchar2 default 'cm') RETURN pls_integer;`

 2

↓ `FUNCTION f_new_table(
 p_doc_id number,
 p_rows pls_integer,
 p_columns pls_integer,
 p_table_width pls_integer default null,
 p_columns_width varchar2 default null) RETURN pls_integer;`

↓ `FUNCTION f_make_document(
 p_doc_id number) RETURN blob;`

 3

How everything works

Documents and elements
are stored in record or collection variables

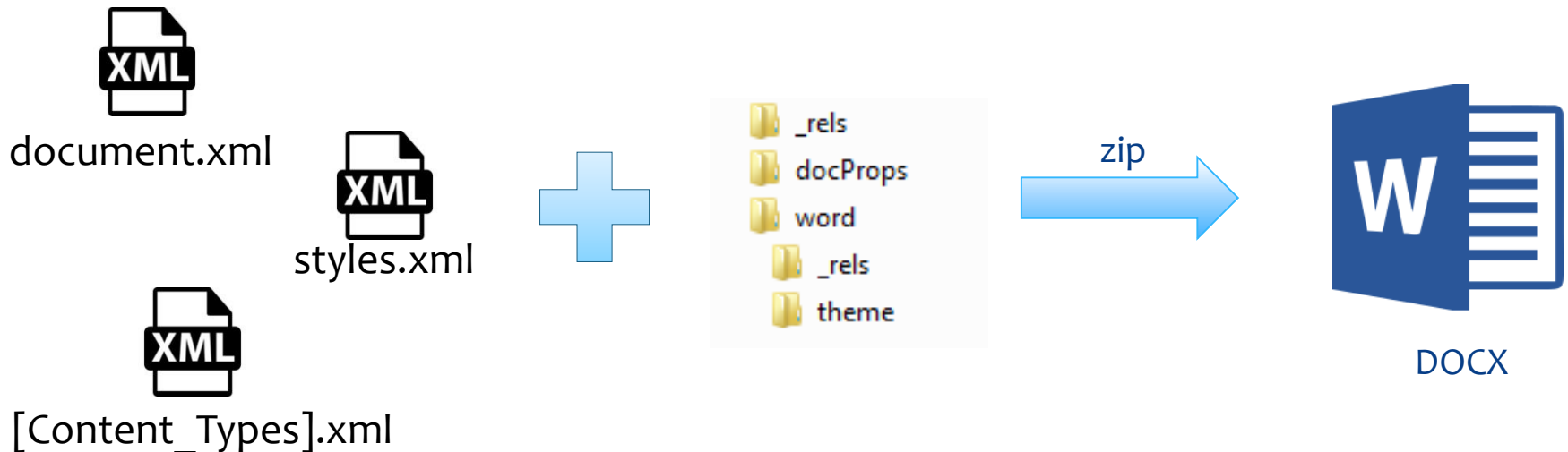
```
TYPE r_font IS RECORD (  
    from_paragraph boolean,  
    font_name varchar2(50),  
    font_size number,  
    bold boolean,  
    italic boolean,  
    underline boolean,  
    color varchar2(6)  
);  
  
TYPE r_text IS RECORD (  
    text varchar2(32000),  
    font r_font  
);  
TYPE t_text IS TABLE OF r_text;
```

```
TYPE r_table_cell IS RECORD (  
    alignment_v varchar2(20),  
    border_top r_border,  
    border_bottom r_border,  
    border_left r_border,  
    border_right r_border,  
    merge_v varchar2(50),  
    merge_h pls_integer,  
    background_color varchar2(6),  
    paragraph r_paragraph  
);  
TYPE t_table_cells IS TABLE OF r_table_cell INDEX BY varchar2(10);  
  
TYPE r_table IS RECORD (  
    width pls_integer,  
    rows_num pls_integer,  
    columns_num pls_integer,  
    column_width t_table_number,  
    cells t_table_cells  
);
```

How everything works

Finishing process

first creates XML documents and then compresses them within appropriate folder structure into DOCX document



Dilemmas



Dilemmas

To start develop this PL/SQL package at all?

Definitely YES!



Dilemmas

Store object data in variables or create XML documents
“on the fly”?

Store object data in variables.
Easier and more flexible solution.



Dilemmas

How to ZIP XML documents into DOCX file? Java, PL/SQL or something else?

PL/SQL ZIP solution
created by Anton Scheffer.



Dilemmas

How to create XML documents? XML DB or PL/SQL string functions?

PL/SQL string functions
(easier and more flexible solution).

CLOB

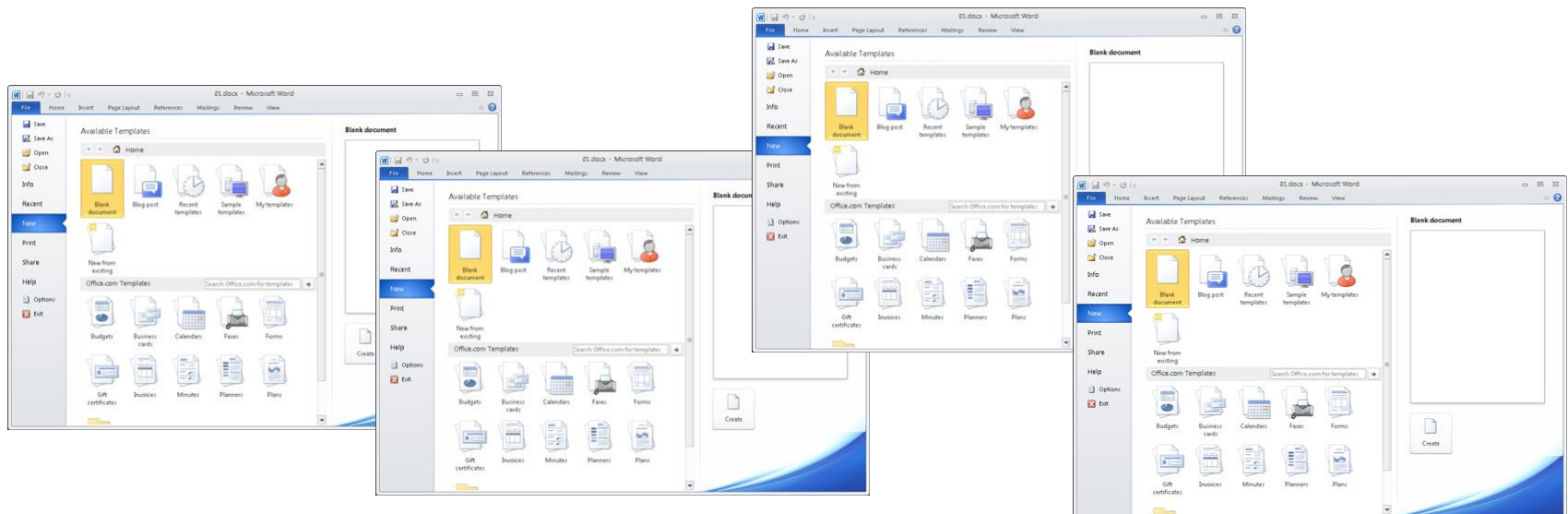
Supported Functionalities



Supported Functionalities

Multiple document generation

create and manipulate multiple documents from the same code



Supported Functionalities

Text functionalities:

- * add paragraphs
- * add text
- * set font
- * set size
- * set color
- * set justification
- * set style

Table functionalities:

- * create table
- * add text to cells
- * set column width
- * set cell background color
- * merge cells
- * cell and table borders

Supported Functionalities

Page functionalities:

- * set page size
- * set page orientation
- * set page margins
- * set page header and footer
- * add page break
- * add section break (next page, even, odd)

Numbered and bulleted lists:

- * create numbered list
- * create bulleted list

Examples



Examples

- * Plain text with different text and paragraph settings including styles
- * Table with header rows and merged cells
- * Section break in document with new page settings and header/footer in new section
- * Add bulleted and numbered list to document
- * Combined all together

Conclusion



Conclusion

- * Utility to generate Microsoft Word DOCX documents from within Oracle database
- * Easy to use and implement in every project with Oracle database
- * Powerful enough to create complex documents
- * Package with currently supported functionalities is free to download and use
- * Download location <http://www.zt-tech.eu> (download section/page)





THE END

Thank you for listening!

