



# Real-time compatible ODF change-tracking

Svante.Schubert@gmail.com  
Freelancer

# CT Interoperability

## ODF with Microsoft Offices (MSO)

- All MSO throw away all ODF changes
  - MS claims ODF underspecified for change-tracking (template styles)
  - No subset implemented in MSO, instead everything ignored
  - No business document ping-pong possible between LO/AO and MSO
  - Problem Solution: Fix of specification for ODF 1.3

# Standard Importance

- Microsoft ODF promise to EU

“Microsoft’s Primary PC Productivity Applications shall support the ODF Standard ... for ten years from the effective date of this Undertaking, within 9 months of final publication by ISO of a new ODF Standard Microsoft shall support that version..”

*<http://www.microsoft.com/en-us/news/press/2009/dec09/12-16statement.aspx>*

# Standard in a Nutshell

- OASIS / ECMA
  - OASIS / ECMA standardization have company members
  - OASIS / ECMA lead by industry for global usage
  - OASIS / ECMA comparable easy to evolve a standard
- ISO (International Organization for Standardization)
  - Slogan after II. World War: "World Peace through World Trade"
  - ISO is an international organization of national standardization groups (National Bodies)
  - Government may influence National Bodies
  - ISO “de jure”/ may dictate usage by law

## Summary: Reason of a Standard

- Blueprint for ODF applications
- Base for Office Interoperability
- Independent place to settle disagreements (OASIS list)
- Leverage to get ODF implementation in MSO

# CT Interoperability: ODF and OOXML

- File Formats ODF and OOXML similar by design
  - Shock frozen state of document
  - Persistent bucket for information
  - ZIP with XML files, images, etc.
- Problem of ODF change-tracking
  - Not caused by the difference of OOXML and ODF
  - Not caused by OOXML feature superset
  - Caused by underspecification in ODF (Style & Table changes)

# CT Interoperability: ODF and OOXML

- CT Design of ODF & OOXML: “Floppy Disc paradigm”
  - Changes stored within the changed component (e.g. paragraph)
  - Before and After state is stored
  - No definition of changes

# Importance of Changes

- Real Time Collaboration (Internet)
  - No documents can/should be exchanged, only changes
  - Goal: Send as little as possible
  - Requires a common model
  - Requires to reference of change (1 millionth paragraph change)
- Changes required for Testing
  1. Load input document
  2. Do changes on document
  3. Check expected output document



# Dependencies of Changes

**Change Tracking  
Changes**

**Undo & Redo  
Changes**

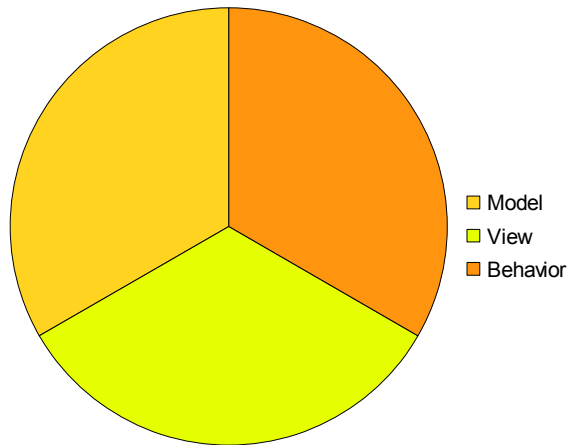
**Collaboration  
Changes**

# CT Interoperability: ODF and OOXML

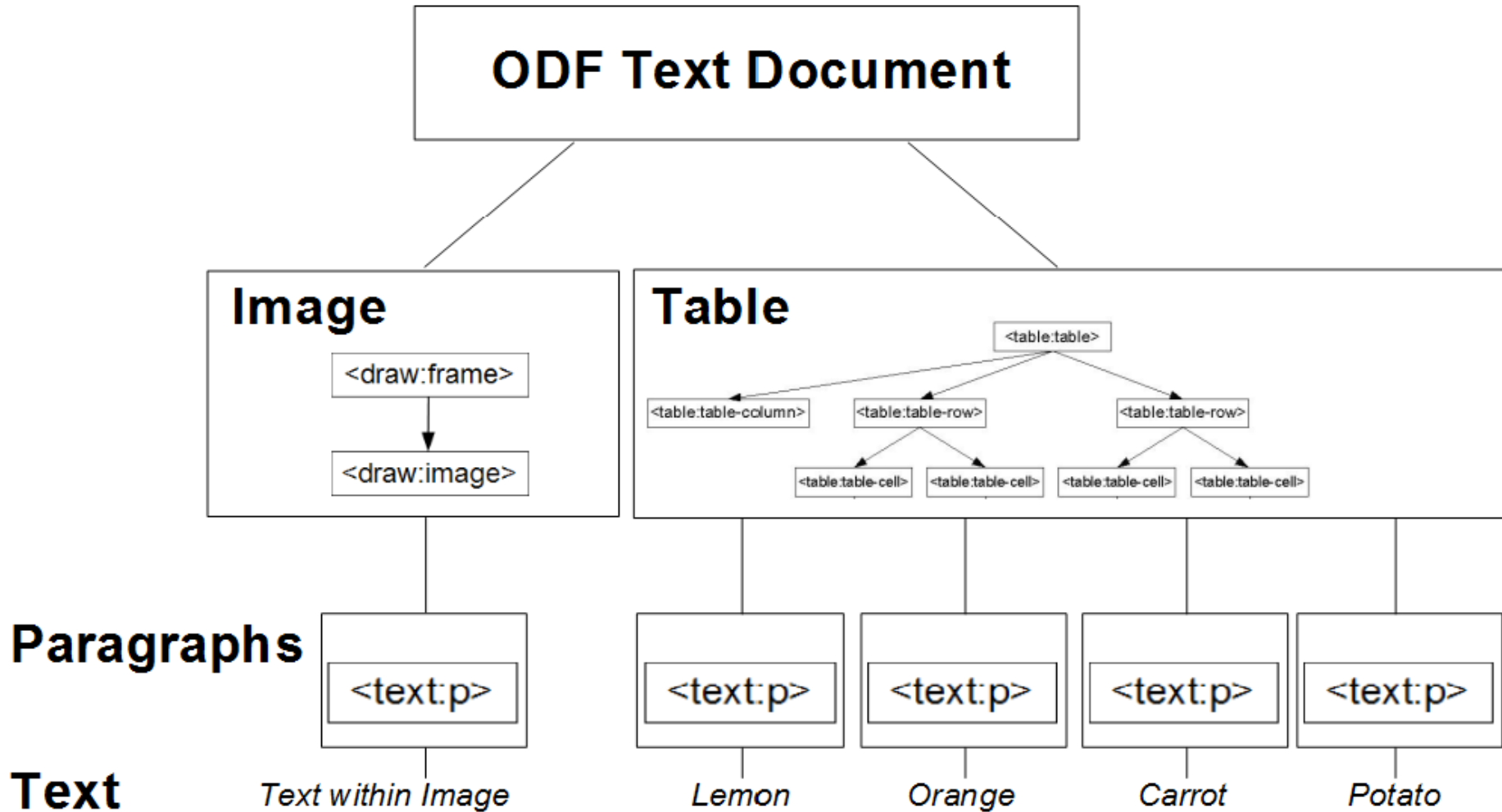
- Design Idea: Move complexity into ODF specification
  - Define Change (Convention over Configuration)
  - Definition of Changes require a common model
  - First step: Abstraction on logical units (components)

# The Holy Grail: Full Document Interoperability

- Document Interoperability consists of:
  - **Model:** text, styles, metadata, signatures, etc.
  - **View:** layout of glyphs, lines, frames, pages, etc.
  - **Behavior:** allowed changes, even macros



# Semantic groups of ODF XML (Components)



# Initial 7 Components of Change Tracking

- Paragraph (Heading, List)
- Text (including Styles, e.g. bold or hyperlink)
- Page
- Table
- Image
- Shape
- Chart

# Design Basics: Meta Operations

- The seven Operation Types:
  - **Add & Delete**
  - **Move** (special case of Add & Delete needed for efficiency)
  - **Replace** (special case of Add & Delete needed for efficiency)
  - **Split & Merge** (text container only)
  - **Format** (special case of adding – only properties)

# Operational Transformation (OT) within Operation “Card Stack”

## Status - 1

```
<do>  
  <add type="paragraph" s="/3">Yours!</add>  
</do>
```

## Status - 2

```
<do>  
  <add type="paragraph" s="/2">Mine!</add>  
  <add type="paragraph" s="/3">Yours!</add>  
</do>
```

## Status - 2

```
<do>  
  <add type="paragraph" s="/2">Mine!</add>  
  <add type="paragraph" s="/3">Yours!</add>  
</do>
```

## Status - 3

```
<do>  
  <add type="paragraph" s="/4">Yours!</add>  
  <add type="paragraph" s="/2">Mine!</add>  
</do>
```



Time  
(latest on top)



“Moving  
operation  
through time“

# Operations Summary

- A different way to describe a document
  - A “list of changes” instead of “before/after state”
- Ease the pain of merge problems
  - Overlapping changes in document
  - Simultaneous change of users  
(Shell game of changes - Where is my change?)
- Enable new possibilities/features
  - Merging changes of huge documents does not require the docs
  - Merging changes independent of document size
  - Accept/Reject Change, now even Delay
  - Commenting a read-only file by keeping changes outside