



Real-Time compatible Change-Tracking

Svante.Schubert@gmail.com
Freelancer

Reason for ODF Standard

- Blueprint for ODF applications
- Specifying the Model (Shock frozen document)
- Specification for output tests of ODF application
- Independent place to settle disagreements (OASIS list)
- Base for Office Interoperability

Standard Usage

- 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>

Standards in a Nutshell

- ECMA / OASIS
 - OASIS / ECMA standardisation 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 organisation of national standardization groups (National Bodies)
 - Government may influence National Bodies
 - ISO "de jure"/ may dictate usage by law

Change-Tracking Basics

- Basic User Scenario
 - User A receives a document for review
 - User A make changes and sent back
 - User B gets changes annotated
 - User B may accept/reject changes of User A

Change-Tracking Basics

- Status Quo: ODF change tracking
 - Before/After Status in the XML
 - No ODF specification what can be changed
 - XML Grammar (RelaxNG) only describes all allowed ODF documents
 - No template styles changes are trackable (e.g. Heading 1)
 - No other file changes are being tracked (e.g. like GIT su



Change-Tracking Basics

- Real-Time Application Scenario
 - User A has a browser Office
 - User B has LibreOffice
 - User A & B like to collaborate (sent & merge changes)
 - User A went offline & wants to save changes in document

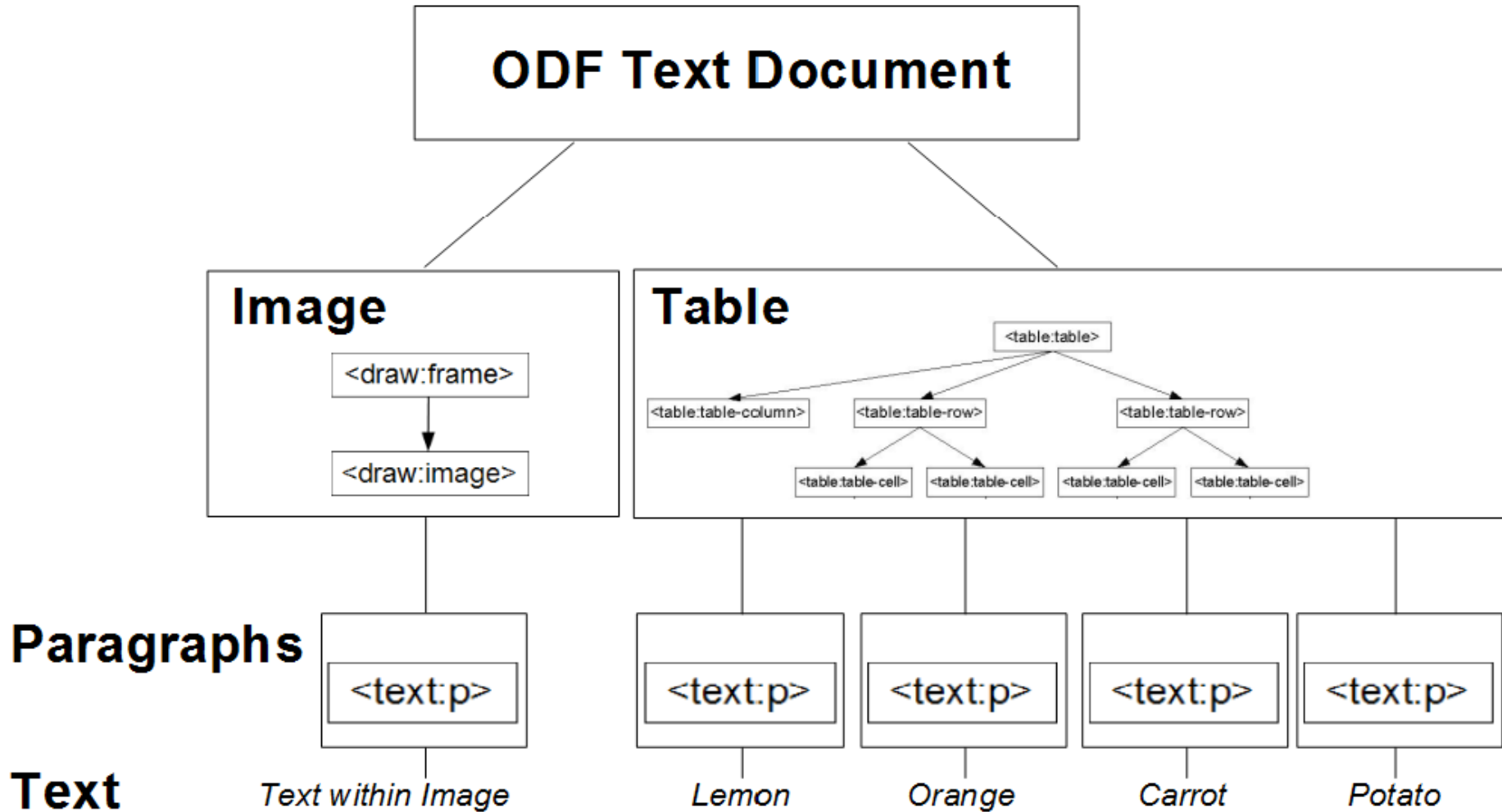
Dependencies of Changes

**Change Tracking
Changes**

**Undo & Redo
Changes**

**Collaboration
Changes**

Semantic groups of ODF XML (Components)



Design Basics: Meta Operations

- The six Operation Types:
 - **Add & Delete**
 - **Move** (special case of Add & Delete needed for efficiency)
 - **Replace** (special case of Add & Delete needed for efficiency)
 - **Split & Merge**

Serializing Operations

Starting markup:

```
<text:h>Some boring, but important text!</text:h>
```

Ending markup:

```
<text:h>Some important text!</text:h>
```

Done changes:

```
<do>  
  <del type="text" s="/1/5" e="/1/16" />  
</do>
```

Undo changes:

```
<undo>  
  <add type="text" s="/1/5"> boring, but</add>  
</undo>
```

Serializing Example: Adding/Removing a List Level

Starting markup:

```
<text:list>  
  <text:list-item><text:p>Line 1</text:p></text:list-item>  
  <text:list-item><text:p>Line 2</text:p></text:list-item>  
  <text:list-item><text:p>Line 3</text:p></text:list-item>  
</text:list>
```

Ending markup:

```
<text:p>Line 1</text:p>  
<text:p>Line 2</text:p>  
<text:p>Line 3</text:p>
```

Changes (only FYI - not being saved):

```
<do>  
  <del type="list-level" s="/1" e="/3">  
</do>
```

Undo changes (undo.xml):

```
<undo>  
  <add type="unordered-list" s="/1" e="/3" />  
</undo>
```

The Operation Queue/Stack

Status - 1

```
<do>  
  <add type="paragraph" s="/5">My work!</add>  
</do>
```

Status - 2

```
<do>  
  <add type="paragraph" s="/2">Colleagues work!</add>  
  <add type="paragraph" s="/5">My work!</add>  
</do>
```

Status - 2

```
<do>  
  <add type="paragraph" s="/2">Colleagues work!</add>  
  <add type="paragraph" s="/5">My work!</add>  
</do>
```

Status - 3

```
<do>  
  <add type="paragraph" s="/6">My work!</add>  
  <add type="paragraph" s="/2">Colleagues work!</add>  
</do>
```

↑
Time
(latest on top)

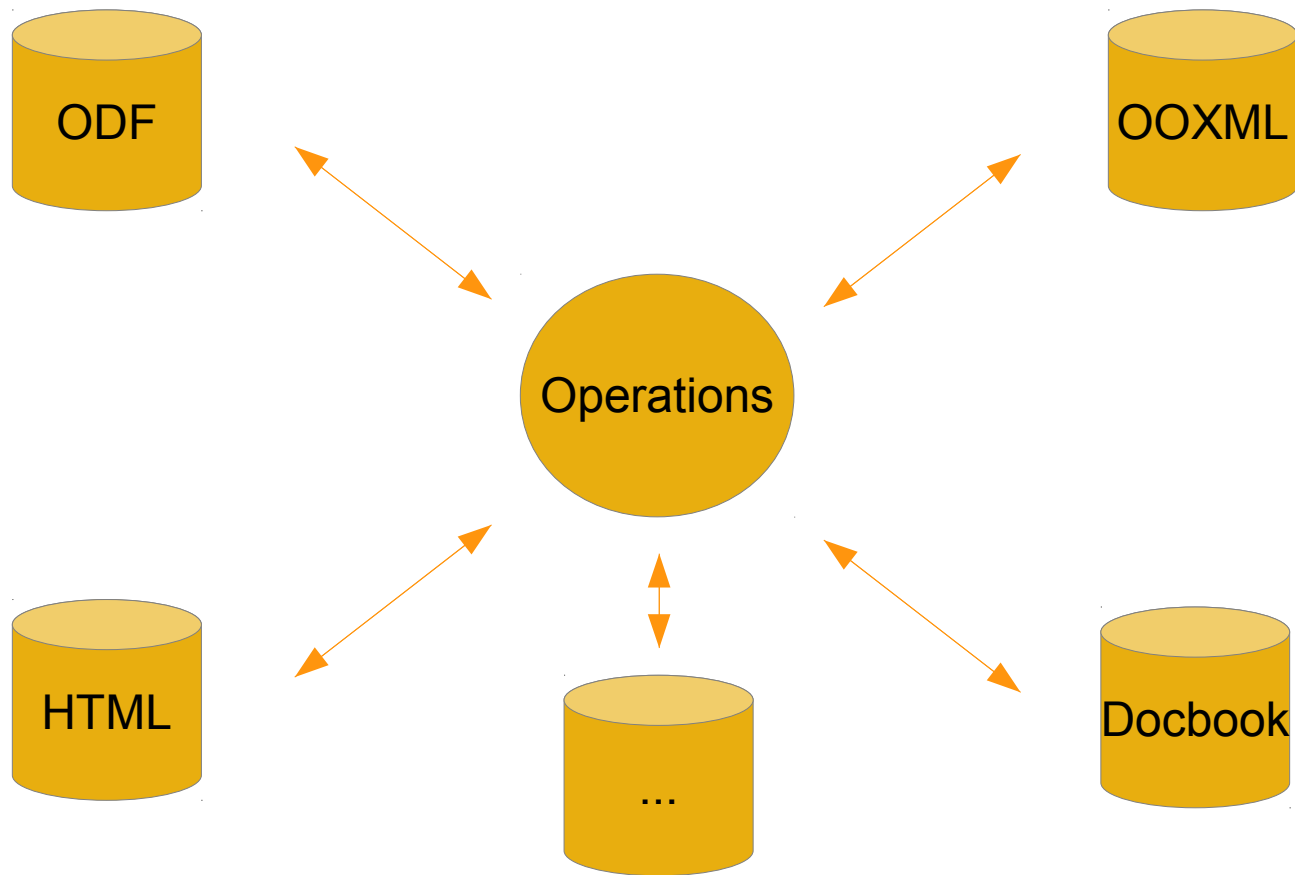
↓
Moving
operation
through time

Basic MCT Design - Relative reference to changes

- Advantages: Changes outside ODF Content
 - Merging changes of huge documents does not require the docs
 - Merging changes independent of document size
 - Allow to apply changes on a read-only document (signed or write protected as in the web)

The Lingua Franca of Operations

- Decouple Transformation Complexity:



The Holy Grail: Full Document Interoperability

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

