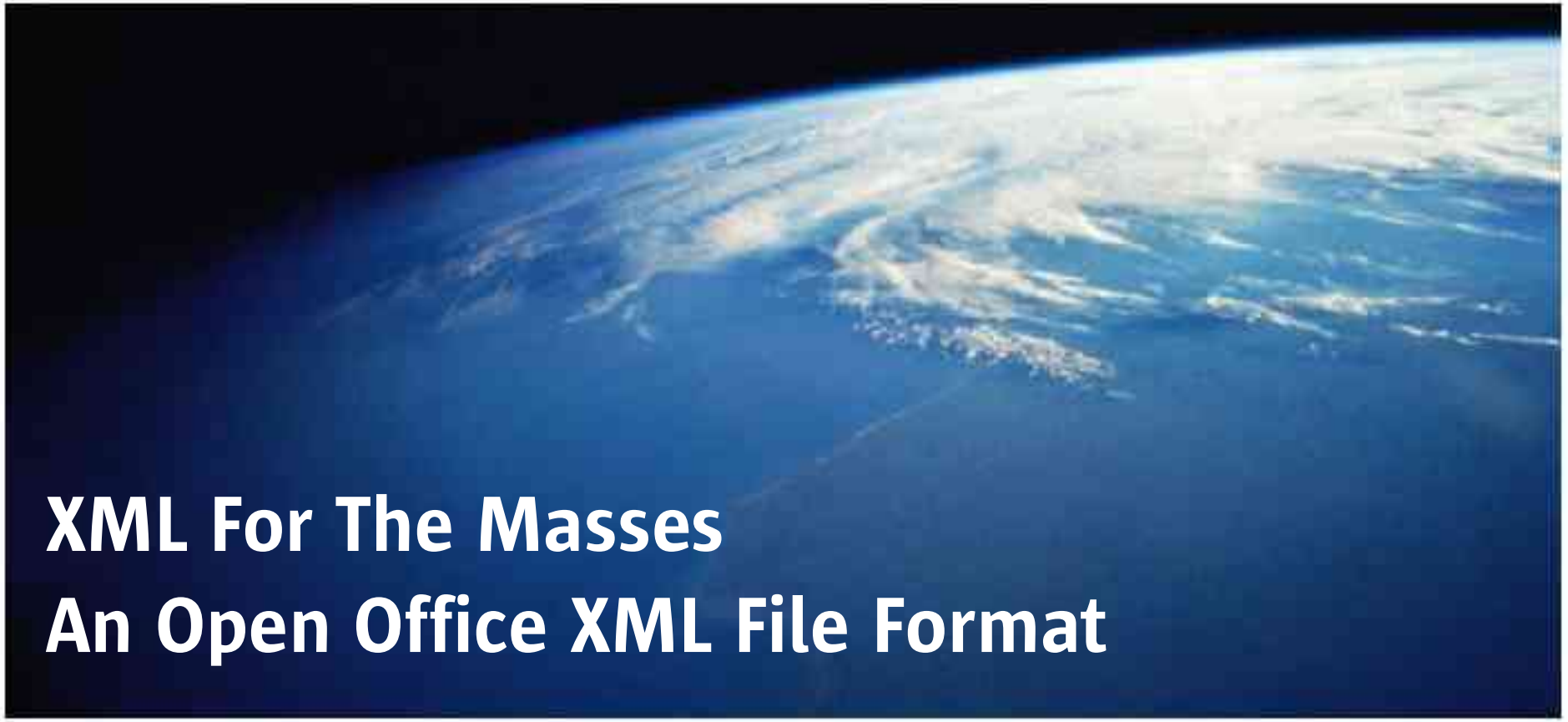


XML For The Masses
An Open Office XML File Format

Daniel Vogelheim

Software Engineer

Sun Microsystems

OASIS MEMBER

Sun microsystems
We make the net work.

What's Wrong With Your Office

The Case for an Open Office Format

- Voice of the Customer
  - office productivity applications
  - need additional processing, integration
    - archiving and indexing
    - content checking
    - database, work-flow integration
    - operation on many files
  - unstructured work-flow, many document types
- Requirement
  - preserve functionality, but open file format

XML Office File Format

Matching Customer's Requirements

- Solution: XML Office File Format
  - XML, for easy integration, processing
  - define office vocabulary
  - requirements for office file format
    - full featured, cover full office productivity space
    - easy to process, easy to generate
    - not vendor or application specific
- OpenOffice.org XML File Format
  - tailored to these requirements

Fixed vs Custom Vocabularies

Why the World Needs an XML Office Format

- Custom Vocabularies
  - requires tooling, extensive preparations
  - work well in specialized, structured work-flows
- Fixed Office Vocabulary
  - allows traditional usage patterns
  - mass-market compatible
  - add XML processing as needed
  - tools can operate on semantic units
  - transform into custom vocabularies

Standardizing The Format

Securing The User's Investment

- A File Format is an Investment
  - must be open, documented
  - must be stable, controlled evolution
- Win-Win with Widespread Adoption
  - more support, more tools
  - need user and industry acceptance
- Develop Standard Format at OASIS
  - based on OpenOffice.org XML Format
    - format has proven useful in real life

A Closer Look: The XML Office File Format

- Format Requirements
  - 1st class XML
  - easy transforms
- Format Details
  - content vs layout
  - binary data
- Format Applications
  - XML processing chains

Designing a File Format

- Meeting the Requirements
  - existing formats? Not sufficient.
  - embrace and extend? No!
  - XML-ify existing structures? No!
- Reviewed Design Process
  - examine existing formats
    - use MathML, XLink, Dublin Core
    - reuse from XHTML, SVG, XSL-FO, CSS
  - specify, review, finally implement

First Class XML

Map all Structured Content to XML

- Fully Compliant: XML, namespaces
- Use XML for Structured Content
  - no information in physical representation
  - no information in comments
  - no information in 'special' names or values
  - no 'sub-formats' for values
- Values Make Sense
  - values vs. presentation
  - process what you need

2002-12-10T11-45

December 12th, 2002

37541.49

Easy Transformations

Making it Simple to Access Office Files

- Consistent Design
  - common format across all applications
  - one concept, one representation
- Reuse of Vocabularies
  - HTML, SVG, DC, MathML, XLink, XSL
- Examples
  - all text in <text:p>, <text:h>
  - extract plain text: 2 XSLT rules
    - add more as you go along (+4 for footnotes)

Styles – Content vs Layout

“Markup reflects a theory of text.”

C. M. Sperberg-McQueen

- Office World
  - document = content + layout
  - layout part of user input
- Semantic Markup
  - document = content
  - layout separate, external (CSS, XSLT)
- We Keep Both, Separately

Styles – Content vs Layout

- Styles Separate From Content
  - easy to change layout
  - easy to process
  - transparent to user
- Style Section(s)
  - convert all formatting into styles
  - separate container elements
  - built-in stylesheet

```
<style name="Emphasis">
 <properties
      font-weight="bold"/>
</style>

<p style-name="Emphasis">
  text text text
</p>
```

Packages

Efficient Handling of XML and Binary Data

- XML File Format Concerns
  - file size
  - embedded images, objects
- XML Package
  - ZIP format, XML-based manifest
  - XML streams + binary streams
    - images, OLE objects, printer setup data
  - used by OpenOffice.org, Gnome, KOffice
- 'pure' XML: embedded BASE64

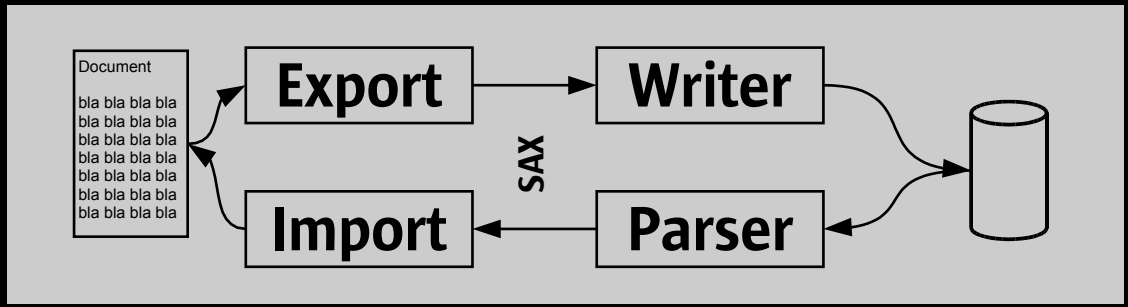
XML Components

Using XML APIs in OpenOffice.org

- SAX: Simple API for XML
  - event-handling style
  - streaming of XML data
  - efficient processing, large documents
- Use XML as Document API
  - read/create in-memory document
  - apply transformations

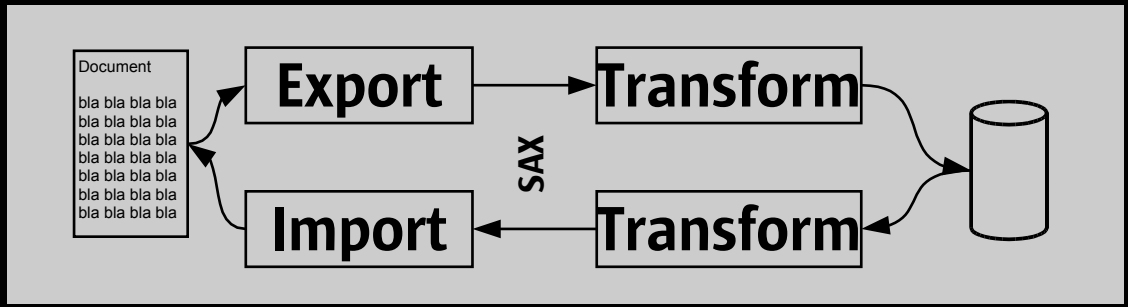
XML Filters & SAX Chains

- File Format Translation
  - during load, save
  - on in-memory document
  - on disk, batch mode, after the fact
- shipping technology
  - DocBook, LaTex, 3 filters in StarOffice



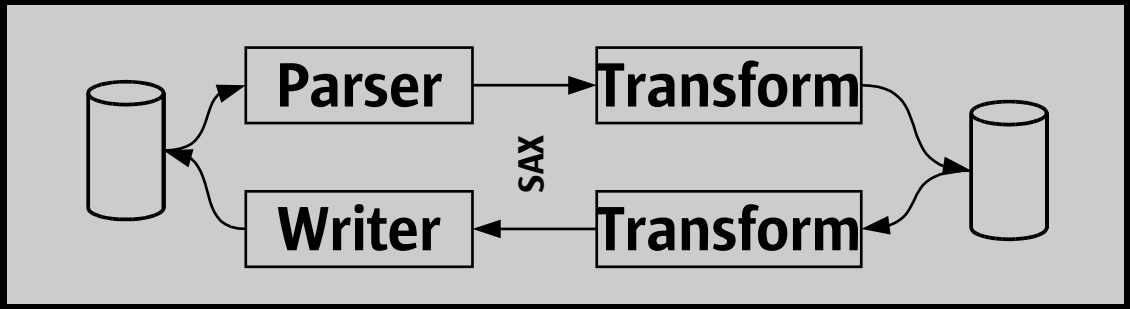
XML Filters & SAX Chains

- File Format Translation
  - during load, save
  - on in-memory document
  - on disk, batch mode, after the fact
- shipping technology
  - DocBook, LaTex, 3 filters in StarOffice



XML Filters & SAX Chains

- File Format Translation
  - during load, save
  - on in-memory document
  - on disk, batch mode, after the fact
- shipping technology
  - DocBook, LaTex, 3 filters in StarOffice



Conclusion

- Office XML File Format
  - fully supports office users & applications
  - enables
    - processing of office documents
    - integration into custom infrastructure
  - fixed vocabulary is key
- Standardization at OASIS
  - open standard, open development
  - secures your investment

Daniel Vogelheim

Daniel.Vogelheim@sun.com

OASIS MEMBER

Sun microsystems

We make the net work.