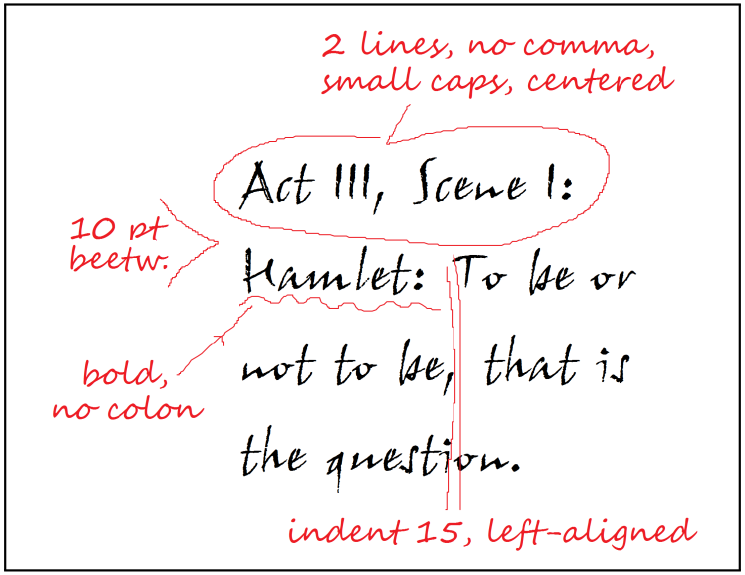
XML and Content Management

Lecture 1: From typesetting to XML

Maciej Ogrodniczuk

MIMUW, 4 October 2009

Text markup: prehistorical times



↓

ACT III
SCENE I

Hamlet

To be or not to be —
that is the question.

Text markup: present time

[Centering on]
[Small caps on]
Act III
[New line]
Scene I
[New line]
[New line]
[Small caps off]
[Centering off]
[Bold on]
Hamlet
[Bold off]
[Indent 10 pt]
To be or not to be —
that is the question.

→

ACT III
SCENE I

Hamlet
To be or not to be —
that is the question.

What is the document composed of?

[Act formatting]
Act III

[Scene formatting]
Scene I

[Person formatting]
Hamlet

[Line formatting]
To be
or not to be —
that is the
question.

+

Act formatting,
scene formatting:
[Centered text]
[Small caps]
[Separate paragraph]

Person formatting:
[Bold text]
[Separate paragraph]

Line formatting:
[Separate paragraph
Indented 10 pt]

=

ACT III
SCENE I

Hamlet

To be or not to be —
that is the question.

Another layout for the same text

[Act formatting]
Act III

[Scene formatting]
Scene I

[Person formatting]
Hamlet

[Line formatting]
To be
or not to be —
that is the
question.

+

Act formatting,
scene formatting:
[Centered, white on black/brown, in separate paragraph]

Person formatting:
[Bold, centered, in separate paragraph]

Line formatting:
[Decorative initial]

=

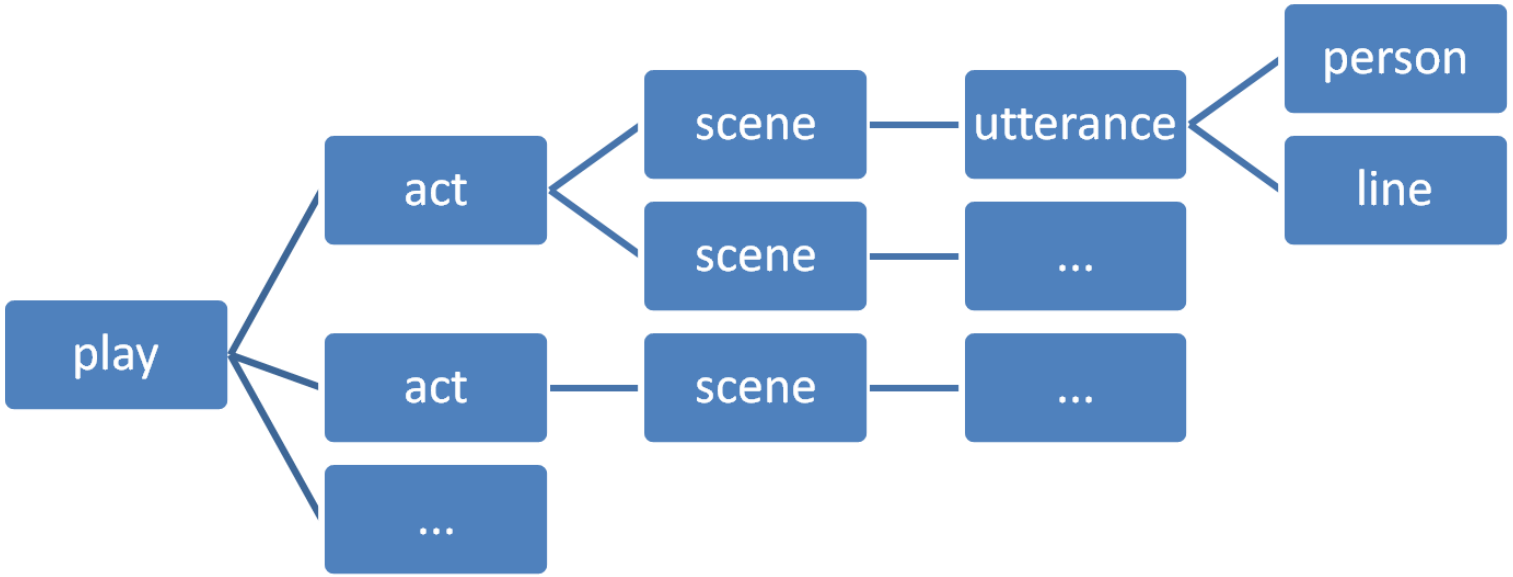
Act III

Scene I

Hamlet

To be or not to be —
that is the
question.

So, what is the document **really** composed of?



Is context-based visualisation a real-world case?

USTAWA

z dnia 2 lipca 2004 r.

o zmianie ustawy – Kodeks postępowania cywilnego oraz niektórych innych ustaw

Art. 1.

W ustawie z dnia 17 listopada 1964 r. – Kodeks postępowania cywilnego (Dz. U. Nr 43, poz. 296, z późn. zm.) wprowadza się następujące zmiany:

1) art. 5 otrzymuje brzmienie:

„Art. 5. W razie uzasadnionej potrzeby sąd może udzielić stronom i uczestnikom postępowania występującym w sprawie bez adwokata lub radcy prawnego niezbędnych pouczeń co do czynności procesowych.”;

...

1969: GML — IBM Script editor macro language

Generalized Markup Language (Goldfarb, Mosher, Lorie):

- tags describe document structure,
- they are replaced with formatting commands,
- there can be multiple "visualization profiles" for one document,
- the initial set of tags can be extended.

GML syntax: an example

```
:p.GML supported hierarchical containers, such as
:ol
:li.Ordered lists (like this one),
:li.Unordered lists, and
:li.Definition lists
:eol.
as well as simple structures.
:p.Markup minimization (later generalized and
formalized in SGML),
allowed the end-tags to be omitted for the "h1" and
"p" elements.
```

http://www.sgmlsource.com/history/roots.htm

Other markup systems

Ventura	`<B>Hamlet<D>`
TEX/LATEX	`\textbf{Hamlet}`
QuarkXPress	`<B>Hamlet<B>`
RTF	`{\b\f5\cf1 Hamlet}`
HTML	`<B>Hamlet</B>`
PostScript	`/Times-BoldR 900 ff(Hamlet)W`
Word 2007	`<w:r><w:rPr><w:b/></w:rPr>`
	`<w:t>Hamlet</w:t></w:r>`

The need of a standard

Problems:

- most applications use their own data formats,
- new version of applications often introduce changes in formats which causes backward compatibility problems (often: not possible to save in previous format),
- converters are available, but:
  - only for the most popular formats,
  - conversion may be lossy.

Our dream format should be:

- flexible and universal so that it could represent many document formats,
- uniform and easy-to-understand to keep its processing simple.

1986: SGML — Standard Generalized Markup Language

SGML — ISO standard 8879:1986.

3 most important features of SGML:

1. descriptivness,
2. document typing,
3. portability.

1994: HTML 2.0 defined as SGML application.

Shortcomings of SGML

Not so many flaws, but there are some basic ones...

1. obligatory definition of document structure,
2. complexity, too many options,
3. processing is difficult (= expensive).

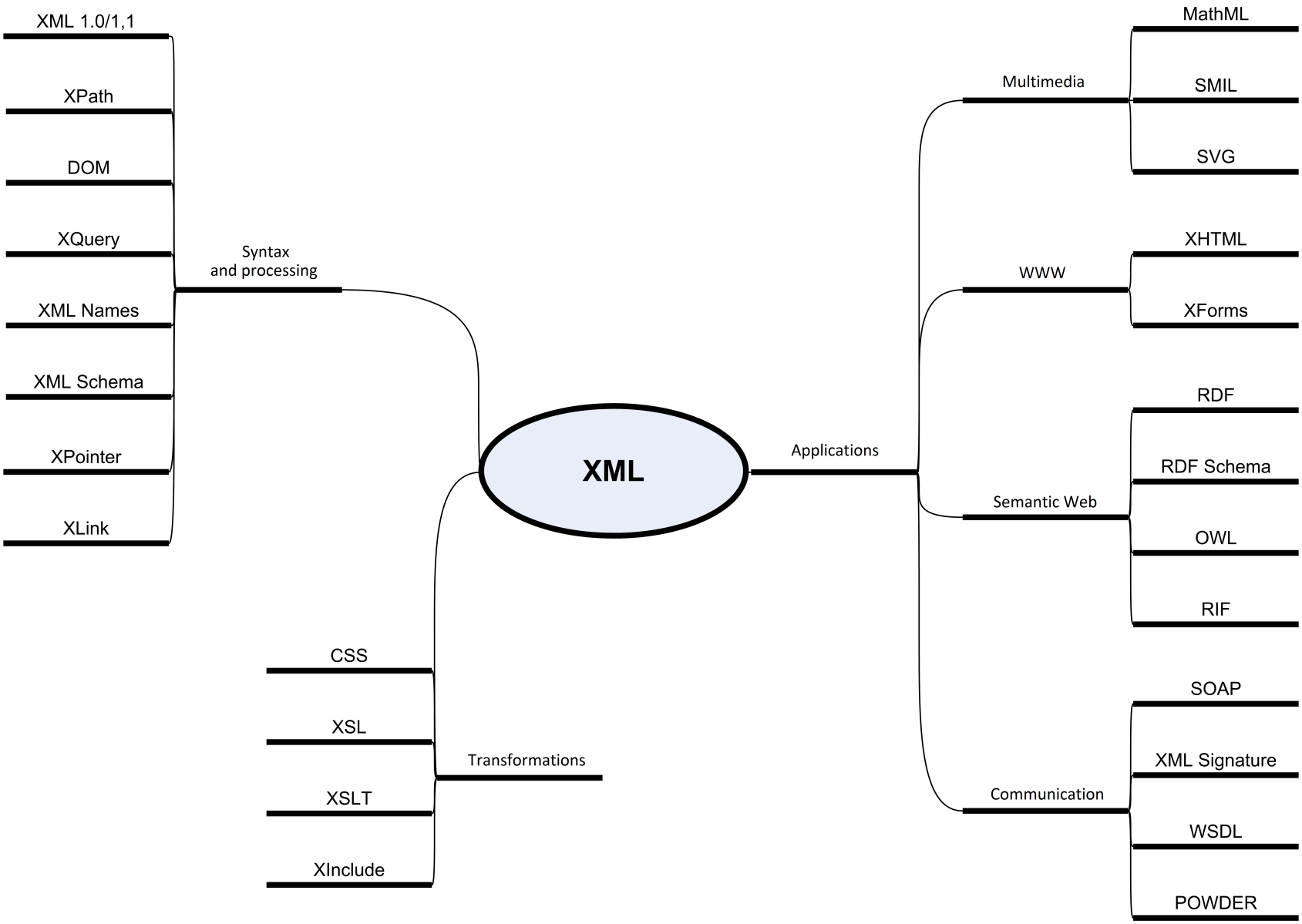
1998: XML — Extensible Markup Language

XML — W3C standard:

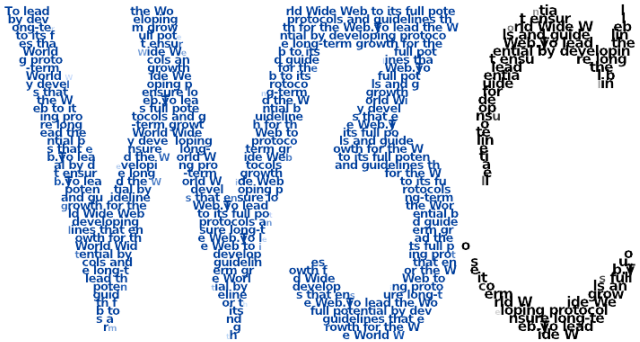
- **1.0**:
  - 1st edition from February 1998 r.,
  - **5th edition from 2008 r.**,
- 1.1 — from 2004 r.,
- 2.0 — more questions than answers.

`http://www.w3.org/TR/REC-xml`

XML-related family of standards



WWW Consortium (W3C)



www.w3.org

W3C is an international consortium developing Internet standards (not just XML-related, but dominated by XML).

WD $\rightarrow$ CR $\rightarrow$ PR $\rightarrow$ REC

standard = agreement + compromise

What XML looks like?

```
<?xml version="1.0" encoding="iso-8859-2"?>
<ustawa>
  <tytuł>o realizacji przedpłat na samochody</tytuł>
  <ustęp nr="1">
    Osoby, które w 1981 r. wniosły przedpłaty na zakup
    samochodów i dotychczas samochodu nie odebrały,
    mają prawo do otrzymania rekompensaty w wysokości:
    <punkt nr="1">
      5.930 zł, jeśli przedpłata została wniesiona
      na samochód marki Fiat 126p,</punkt>
    <punkt nr="2">
      8.400 zł dla samochodu marki FSO 1500.</punkt>
  </ustęp>
</ustawa>
```

What XML looks like?

XML declaration:

```
<?xml version="1.0"
      encoding="windows-1250"
      standalone="yes"?>
```

- Always at the beginning of document (first line, starting from the first character).
- Can be omitted if XML version is intended to be 1.0 and the text is UTF-8 or UTF-16-encoded.
- Specification definintion: `XMLDecl ::= '<?xml' VersionInfo EncodingDecl? SDDecl? S? '?>'`
- Encoding-related issues → see annex F of the XML specification.

What XML looks like?

Elements i attributes:

```
<element attribute="value">
  <empty1></empty1>
  <empty2/>
</element>
```

- Each document has one and only one root element.
- Elements can be nested and can contain text nodes.
- Attribute values must be surrounded by quotation marks or apostrophes.
- Having two attributes with the same name at a given element is forbidden.
- Attribute values cannot store < character.

What XML looks like?

XML comments:

```
<!-- XML comment
  ...
  -->
```

- Can span over multiple lines.
- Can store arbitrary characters, including <, >, &.
- Cannot have two adjacent minus characters in content.
- Cannot be embedded into element tags.
- Can be present outside the root element.

What XML looks like?

Processing instructions:

```
<?instruction and-its content?>
```

- Can be placed outside the root element.
- PI name must be other than „XML" (and different case variants).

What XML looks like?

CDATA sections:

```
<!$[$CDATA[any <text " which won't be processed \
& by [parser]]>
```

- The only forbidden string inside CDATA section is `]]>` (not only here, but also in the "ordinary" text content, so we can cite XML files which do not contain CDATA section).
- CDATA sections cannot be nested.

Well-formedness

If the document is constructed according to XML syntax rules, we call it *well-formed*.

This means that e.g.:

- all tags are closed and properly nested,
- attribute values are enclosed in apostrophes or quotation marks,
- attribute names must be unique for the element,
- ... and 9 other requirements (→ specification).

Verification of well-formedness of XML documents is done by **XML parsers**.

Namespaces

Problem:

How to merge:

1. ```
   <author>
     <title>prof.</title>
     <first-name>John</first-name>
     <surname>Curious</surname>
   </author>
   ```
2. ```
   <chapter no="1">
     <title>Deep blue something</title>
     ...
   </chapter>
   ```

Namespaces

Solution:

Using *namespaces* — associations between URI/IRI and selected „context name".

Namespaces:

- disambiguate names,
- identify source of their definition.

Namespaces

Usage:

1. declare namespace for the subtree with `xmlns:`*prefix* attribute and having URI/IRI as its value — e.g. `xmlns:authors="http://example.com/authors"`,
2. to express that **name** of the `<surname>` element comes from `http://example.com/authors` namespace, `<authors:surname>` notation is used,
3. no prefix (`xmlns="http://example.com/authors"`) creates so-called default namespace.

Best namespace document:
`http://www.rpbourret.com/xml/NamespacesFAQ.htm`

Namespaces

An example:

```
<book>
  <authors xmlns:a="http://example.com/authors">
    <a:author>
      <a:title>prof.</a:title>
      <a:first-name>John</a:first-name>
      <a:surname>Curious</a:surname>
    </a:author>
  </authors>
  <content>
    <chapter no="1">
      <title>Deep blue something</title>
      ...
    </chapter>
  </content>
</book>
```

Which names belong to which namespace?

Another example:

```
<a xmlns:b="urn:b">
  <c xmlns="urn:x">
    <d:e xmlns="urn:y"
         xmlns:d="urn:d"
         b:f="g">
      <i xmlns:b="urn:b2">
        <j b:k="l"/>
      </i>
    </d:e>
    <m n="o"/>
  </c>
  <p/>
</a>
```

Solution:

a does not belong to any namespace

c belongs to urn:x

e belongs to urn:d

f belongs to urn:b

i belongs to urn:y

j belongs to urn:y

k belongs to urn:b2

m belongs to urn:x

n does not belong to any namespace!

p does not belong to any namespace

XML and diacritics

Default encoding for XML documents is UTF-8.

Usage in XML document:

- respective string,
- `&#`*code*`;` — decimal character code,
- `&#x`*code*`;` — hexadecimal character code.

Diacritics can be used in attribute and element names.

UTF-8 and Unicode

Unicode is a standard describing (in principles) a set of characters corresponding to all characters used around the world.

Unicode assigns numbers to characters, but it does not
define their representation (encoding).

	010	011	012	013	014	015	016	017
0	Ā 0100	Đ 0110	Ġ 0120	İ 0130	ŀ 0140	Ő 0150	Š 0160	Ű 0170
1	ā 0101	đ 0111	ġ 0121	ı 0131	Ł 0141	ő 0151	š 0161	ű 0171
2	Ă 0102	Ē 0112	Ģ 0122	IJ 0132	ł 0142	Œ 0152	Ţ 0162	Ţ 0172
3	ă 0103	ē 0113	ģ 0123	ij 0133	Ń 0143	œ 0153	ţ 0163	ţ 0173
4	Ą 0104	Ĕ 0114	Ĥ 0124	Ĵ 0134	ń 0144	Ŕ 0154	Ť 0164	Ŵ 0174
5	ą 0105	ĕ 0115	ĥ 0125	ĵ 0135	Ņ 0145	ŕ 0155	ť 0165	ŵ 0175

Most popular encodings:

- UTF-8 (first 128 = ASCII, 1 character can be 1-6 bytes),
- UTF-16 (each character is one or two 16-bit words),
- UTF-32/UCS-4 (each characters is one 32-bit word).

XML and meaning of document fragments

```
<ustawa data-wydania="2009-09-10">
  <tytuł>o zniesieniu podatków</tytuł>
  <artykuł nr="1">Wszystkie podatki zostają zniesione
           z dniem 1 stycznia 2010 r.</artykuł>
</ustawa>

<u45 d="2009-09-10">
  <t>o zniesieniu podatków</t>
  <a n="1">Wszystkie podatki zostają zniesione
           z dniem 1 stycznia 2010 r.</a>
</u45>
```

XML representation of a document
does not assign any meaning to its fragments!

Summary: XML in 10 points (according to W3C)

1. XML is for structuring data,
2. XML looks a bit like HTML,
3. XML is text, but isn't meant to be read,
4. XML is verbose by design,
5. XML is a family of technologies,
6. XML is new, but not that new,
7. XML leads HTML to XHTML,
8. XML is modular,
9. XML is the basis for RDF and the Semantic Web,
10. XML is license-free, platform-independent and well-supported.

`http://www.w3.org/XML/1999/XML-in-10-points.html`