

XML

Extensible Markup Language

XML - Historia

Historia

Varför XML?

XML-exempel

Korrekt XML

DTD

Schema

Att parsas XML

DOM

SAX

Fler delar

Namespaces

XSLT

XPath

Stylesheets

XML Linking

1969 Generalized Markup Language (GML)

1979-85 Standard Generalized Markup Language (SGML)

1990 HyperText Markup Language (HTML)

1996-98 Extensible Markup Language (XML)

- Delmängd av SGML
- Är inte ett språk, utan standard för att skapa språk

Varför XML?

Historia

Varför XML?

XML-exempel

Korrekt XML

DTD

Schema

Att parse XML

DOM

SAX

Fler delar

Namespaces

XSLT

XPath

Stylesheets

XML Linking

- + Standard
- + Många verktyg
- + Enkelt att bygga ut
- + Läsbart
- Stora filer

XML-exempel

Historia
Varför XML?
XML-exempel
Korrekt XML
 DTD
 Schema
Att parsas XML
 DOM
 SAX
Fler delar
 Namespaces
 XSLT
 XPath
 Stylesheets
 XML Linking

```
<?xml version="1.0"
      encoding="ISO-8859-1"?>
<groups>
  <group name="RÄJS&TRÄJS" id="5">
    <members>
      <member>
        <name>Angela Sjöholm</name>
        <persnr>123456-7890</persnr>
        <mail>ansjovis@student.uu.se</mail>
      </member>
      ...
    </members>
  </group>
  ...
</groups>
```

Korrekt XML

- Historia
- Varför XML?
- XML-exempel
- Korrekt XML**
- DTD
- Schema
- Att parsas XML
- DOM
- SAX
- Fler delar
- Namespaces
- XSLT
- XPath
- Stylesheets
- XML Linking

- Allt mellan `<` och `>` är en tag.
- `<name>` är en start-tag, `</name>` är en end-tag.
- Varje start-tag måste matchas av en end-tag.
- `<name></name>` \Leftrightarrow `<name/>`
- `<name>Till Lindemann</name>` är ett element.
- Element kan innehålla andra element.
- Det kan endast finnas ett element på toppnivån (root element)
- Element måste vara snyggt nästlade, dvs `<a>` är inte tillåtet.
- + ett gäng med fler regler om element.
- Ett attribut är ett namn-värde-par som finns i start-tagen.

Document Type Definition (DTD)

Historia
Varför XML?
XML-exempel
Korrekt XML
DTD
Schema
Att parsas XML
DOM
SAX
Fler delar
Namespaces
XSLT
XPath
Stylesheets
XML Linking

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE scene SYSTEM
"http://www.csd.uu.se/kurs/pm2/ht02/www/pm2gl.dtd">
<scene>
  <camera>
    <location>
      <vector x="0.000" y="3.000" z="-10.000">
      </vector>
    </location>
    <sky>
      <vector x="0.000" y="1.000" z="0.000">
      </vector>
    </sky>
    <look_at>
      <vector x="0.000" y="0.000" z="0.000">
      </vector>
    </look_at>
  </camera>
  ...
```

Document Type Definition (DTD)

- Historia
- Varför XML?
- XML-exempel
- Korrekt XML
- DTD**
- Schema
- Att parsas XML
- DOM
- SAX
- Fler delar
 - Namespaces
 - XSLT
 - XPath
 - Stylesheets
 - XML Linking

```
<!ELEMENT scene (camera, background?,
                 world)>

<!ELEMENT camera (location, sky,
                 look_at)>

<!ELEMENT sky (vector)>
<!ELEMENT look_at (vector)>
<!ELEMENT location (vector)>
<!ELEMENT vector EMPTY>
<!ATTLIST vector
  x CDATA #required
  y CDATA #required
  z CDATA #required>
```

Document Type Definition (DTD)

Historia

Varför XML?

XML-exempel

Korrekt XML

DTD

Schema

Att parse XML

DOM

SAX

Fler delar

Namespaces

XSLT

XPath

Stylesheets

XML Linking

- Använder egen syntax (ärvt från SGML).
- Begränsad uttrycks kraft.
 - Ordning på element.
 - Dålig hantering av datatyper.
- + Enkelt att lära och använda.
- + Många verktyg stöder DTD:er.

XML Schemas

Historia
Varför XML?
XML-exempel
Korrekt XML
DTD
Schema
Att parsas XML
DOM
SAX
Fler delar
Namespaces
XSLT
XPath
Stylesheets
XML Linking

```
<?xml version="1.0"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema">

  <element name="vector">
    <complexType>
      <attribute name="x" type="float"/>
      <attribute name="y" type="float"/>
      <attribute name="z" type="float"/>
    </complexType>
  </element>

  <element name="location">
    <complexType>
      <sequence>
        <element ref="vector"/>
      </sequence>
    </complexType>
  </element>
```

XML Schemas

Historia

Varför XML?

XML-exempel

Korrekt XML

DTD

Schema

Att parsas XML

DOM

SAX

Fler delar

Namespaces

XSLT

XPath

Stylesheets

XML Linking

- + Kraftfulla
- + Använder XML-syntax
- + Bra stöd för arv.
- Svårare att lära sig.
- Stöds ännu inte 100%-igt av verktygen.

Att parse XML

Historia
Varför XML?
XML-exempel
Korrekt XML
 DTD
 Schema
Att parse XM
 DOM
 SAX
Fler delar
 Namespaces
 XSLT
 XPath
 Stylesheets
 XML Linking

Det finns massor av verktyg för att läsa XML-filer, t.ex. expat.

I huvudsak två kategorier:

- Document Object Model (DOM)
- Simple API for XML (SAX)

Document Object Model (DOM)

Historia
Varför XML?
XML-exempel
Korrekt XML
 DTD
 Schema
Att parse XML
DOM
 SAX
Fler delar
 Namespaces
 XSLT
 XPath
 Stylesheets
 XML Linking

- Bygger en trädstruktur utifrån XML-dokumentet.
 - Ett standard-API används sedan för att komma åt information i trädet.
- Läser in hela XML-dokumentet i minnet.
 Kan ta mycket minne.
 Kan ta lång tid.

Simple API for XML (SAX)

Historia
Varför XML?
XML-exempel
Korrekt XML
 DTD
 Schema
Att parse XML
 DOM
 SAX
Fler delar
 Namespaces
 XSLT
 XPath
 Stylesheets
 XML Linking

- Händelsestyrt.
- Objektorienterat.

+ Enkelt att använda.

+ Man behöver inte läsa in hela XML-filen.

- Man måste själv hålla reda på var i filen man befinner sig.

Expat är också händelsestyrt.

Namespaces

Historia
Varför XML?
XML-exempel
Korrekt XML
DTD
Schema
Att parse XML
DOM
SAX
Fler delar
Namespaces
XSLT
XPath
Stylesheets
XML Linking

- Ett schema eller en DTD beskriver en vokabulär.
- Man kan använda element från flera vokabulärer i samma XML-dokument.
- Kan leda till krockar!
- En namnrymd är en grupp av namn som hör ihop rent konceptuellt.
- Prefix kan användas för att ange namnrymd.
Ex. `<html:title>` och `<book:title>`
- Problem: Hur ska man göra prefixen unika?
- Lösning: Utnyttja URI (Uniform Resource Identifier), t.ex. URL.

Ex.

```
<book xmlns="http://www.it.uu.se/book"
```

```
xmlns:html="http://www.w3.org/1999/xhtml">
```

Extensible Stylesheet Language Transformations (XSLT)

Historia
Varför XML?
XML-exempel
Korrekt XML
 DTD
 Schema
Att parse XML
 DOM
 SAX
Fler delar
 Namespaces
 XSLT
 XPath
 Stylesheets
 XML Linking

- Ett XML-baserat språk.
- Används för att översätta från XML till andra text-baserade språk (t.ex. HTML, XML).
- Deklarativt språk. Det säger *vad*, istället för *hur*.
- Ett program, t.ex. xsltproc, läser källfilen (XML), applicerar mallarna i stylesheet-filen (XSL), och skriver ut resultatet till en fil (t.ex. HTML).

Extensible Stylesheet Language Transformations (XSLT)

Historia
Varför XML?
XML-exempel
Korrekt XML
 DTD
 Schema
Att parsas XML
 DOM
 SAX
Fler delar
 Namespaces
XSLT
 XPath
 Stylesheets
 XML Linking

Exempel:

```
<?xml version="1.0"?>
<xsl:stylesheet version="1.0" xmlns:xsl=
    "http://www.w3.org/1999/XSL/Transform">
  <xsl:output method="html" ident="yes"
    encoding="ISO-8859-1"/>
  <xsl:template match="/">
    <html>
      <head><title>PM2-grupper</title></head>
      <body>
        <h1>PM2-grupper</h1>
        <table border="1">
          <xsl:apply-templates/>
        </table>
      </body>
    </html>
  </xsl:template>
  ...
```


Extensible Stylesheet Language Transformations (XSLT)

Historia
Varför XML?
XML-exempel
Korrekt XML
 DTD
 Schema
Att parsas XML
 DOM
 SAX
Fler delar
 Namespaces
XSLT
 XPath
 Stylesheets
 XML Linking

Resultat av groups.xsl:

PM2-grupper

(1) The Shadow	Liselotte Burvall libu3 Maryam Ladjvardi mal Mordin Dilshad moc
(2) SkyBar	Olof Sivertsson olsi076 Jöns Åkerlund joak48 Jakob Ahlen jaah22
(3) Dick Charles (tm)	Martin Kjellin Johannes Skäremo Holmberg Juan Pablo Fernandez

Extensible Stylesheet Language Transformations (XSLT)

Historia
Varför XML?
XML-exempel
Korrekt XML
 DTD
 Schema
Att parsas XML
 DOM
 SAX
Fler delar
 Namespaces
XSLT
 XPath
 Stylesheets
 XML Linking

Resultat av group-status.xsl:

PM2-status

Ni som inte har gruppnamn, skicka ett mail med namn
I = inlämnad

Grupp	Del 1	Del 2	Del 3
(1) The Shadow	I		
(2) SkyBar	I		
(3) Dick Charles (tm)	I		
(4) e-light	I		
(5) RÄJS&TRÄJS	I		
(6) Raytracer Schmaytracer	I		

XPath

Historia
Varför XML?
XML-exempel
Korrekt XML
 DTD
 Schema
Att parsas XML
 DOM
 SAX
Fler delar
 Namespaces
 XSLT
 XPath
 Stylesheets
 XML Linking

- Hitta rätt i XML-filer.
- Exempel:

```
<xsl:template match="//group">
  <tr><td>
    (<xsl:value-of select="./@id"/>)
    <xsl:value-of select="./@name"/>
  </td>
  <xsl:apply-templates/>
</tr>
</xsl:template>
```

- Används inte bara av XSL, utan även av t.ex. XPointer.

Cascading Style Sheets (CSS)

Historia
Varför XML?
XML-exempel
Korrekt XML
DTD
Schema
Att parsas XML
DOM
SAX
Fler delar
Namespaces
XSLT
XPath
Stylesheets
XML Linking

- Ett sätt att styra hur XML och HTML ska visas.
- Ett HTML-exempel:

```
<h1><font size="7" face="New Times Roman">  
Det här är en titel.</font></h1>
```

Med CSS skulle detta i stället kunna göras som:

```
<link href="mystyle.css" rel="stylesheet"  
      type="text/css"/>  
<h1>Det här är en titel</h1>
```

mystyle.css:

```
h1 {  
    font-family: New Times Roman;
```

XML Linking

- Historia
- Varför XML?
- XML-exempel
- Korrekt XML
 - DTD
 - Schema
- Att parsas XML
 - DOM
 - SAX
- Fler delar
 - Namespaces
 - XSLT
 - XPath
 - Stylesheets
 - XML Linking**

- XLink är ett sätt att länka XML-dokument till varandra.
- Begränsningar hos HTML-länkar:
 - Länkar bara två saker: dokumentet den finns i och dokumentet den länkar till.
 - Är enkelriktade.
- XLink löser detta.
- XPointer används för att länka till delar av XML-dokument.
- För att göra detta använder XPointer XPath.
- Tillsammans kan XLink och XPointer skapa komplexa länkar.

Slut - frågor?

Historia
Varför XML?
XML-exempel
Korrekt XML
 DTD
 Schema
Att parsas XML
 DOM
 SAX
Fler delar
 Namespaces
 XSLT
 XPath
 Stylesheets
 XML Linking

Fler XML saker:

XQuery - frågespråk, jämför med SQL.

Nähä. Bra, då slutar vi.

- Historia
- Varför XML?
- XML-exempel
- Korrekt XML
 - DTD
 - Schema
- Att parse XML
 - DOM
 - SAX
- Fler delar
 - Namespaces
 - XSLT
 - XPath
 - Stylesheets
 - XML Linking