

# XML

(CC BY-NC-SA 4.0)

[Wim.bertels@ucll.be](mailto:Wim.bertels@ucll.be)

# XML

- eXtensible Markup Language
- DTD (of xml schema)
- 1996
- hierarchische structuren in een nieuw kleedje
- platte tekstbestanden
- ISO standaard
- einde RDBMS..?

# eenvoudige XML Syntax

- men gebruikt een tag om aan te geven waar men begint en eindigt
- elke document moet in 1 root tag omvat zijn
- beginnen met een cijfer mag niet
- case sensitive
- bv  
    <example> tekst </example>
- tags mogen genest worden

# Voorbeeld

<energie>

<gerecht>

<naam> frieten </naam>

<quoting> zeer wel </quoting>

</gerecht>

<gerecht>

<naam> sojapuddingzonderchoco </naam>

<quoting> twijfelachtig </quoting>

</gerecht>

</energie>

# Doel

- xml (data) vs html (vorm)  
(xsl vs css)
- om zowel hierarchische datastructuren te omschrijven als deze te bevatten
- om data uit wisselen tussen verschillende gegevensbronnen (cf s-patroon:mediator)
  - een gemeenschappelijke taal
  - geef hiervan een voorbeeld.

# METADATA

- je kan attributen aan je tags meegeven, deze moeten tussen " " of ' ' staan
- gebruik dit enkel om METADATA weer te geven, bv eigenschappen
- de data zelf horen daar niet thuis
- bv

`<example type='music'> lalala.mp3 </example>`

# Uitbreidingen

- xml word bv ook gebruikt om configuraties van software bij te houden bv menubalk, ..
- er zijn verschillende formaten die van deze standaard gebruik maken:  
xhtml, xml dom, xsl, xslt, xpath, xsl-fo, xlink, xpointer, dtd, xsd, xforms, xquery, soap, wsd, rdf, rss, svg, wap, smil, ..

# Pro Contra

- ISO
  - uitwisselbaarheid
  - eenvoud
  - geen hiërarchische engine nodig
  - ..
- redundantie tov relationeel model, dus niet misbruiken in die zin
  - sequentieel, traag
  - ..



# RDBMS

- Verschillende RDBMS voorzien manieren om xml te gebruiken
  - <http://www.postgresql.org/docs/current/static/datatype-xml.html>
  - <http://www.postgresql.org/docs/current/static/functions-xml.html>

# Voorbeelden

```
SELECT xmlcomment('hallo');
```

xmlcomment

-----

<!--hallo-->

```
SELECT xmlelement(name jos,  
  xmlattributes(current_date as ke), 'hal', 'lo');
```

xmlelement

-----  
<jos ke="2014-01-26">hallo</jos>

```
select xmlforest(divisie, teamnr) from teams;
```

xmlforest

-----

```
<divisie>ere </divisie><teamnr>1</teamnr>  
<divisie>tweede</divisie><teamnr>2</teamnr>
```

```
SELECT xmlpi(name php, 'echo "Patat";');
```

xmlpi

-----

```
<?php echo "Patat";?>
```

..  
table\_to\_xml(tbl regclass, nulls boolean,  
tableforest boolean, targetns text)

query\_to\_xml(query text, nulls boolean,  
tableforest boolean, targetns text)

cursor\_to\_xml(cursor refcursor, count int, nulls  
boolean, tableforest boolean, targetns text)

..

# Links

- <http://www.postgresql.org/docs/current/static/datatype-xml.html>
- <http://www.postgresql.org/docs/current/static/functions-xml.html>

Wim Bertels

Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International Public License