

XML met PLS QL

Cor van Zalk

Wie ben ik?

- Cor van Zalk
 - Sinds 2005 Oracle ontwikkelaar, OCA, OCE SOA
- Interesses
 - XML
 - SOA
 - SCA
- E-mail: cvzalk@vxcompany.com

Onderwerpen

- Methoden
- Best practices
- Vragen

Methoden

- S O A suite en andere tools
- XMLTYPE
- S Q L XML Functions
- ||
- E tc..

|| voorbeeld

Declare

```
v_xml varchar2(2000);
```

Begin

```
v_xml := '<DatumTijd>' || sysdate || '</DatumTijd>';
```

END;



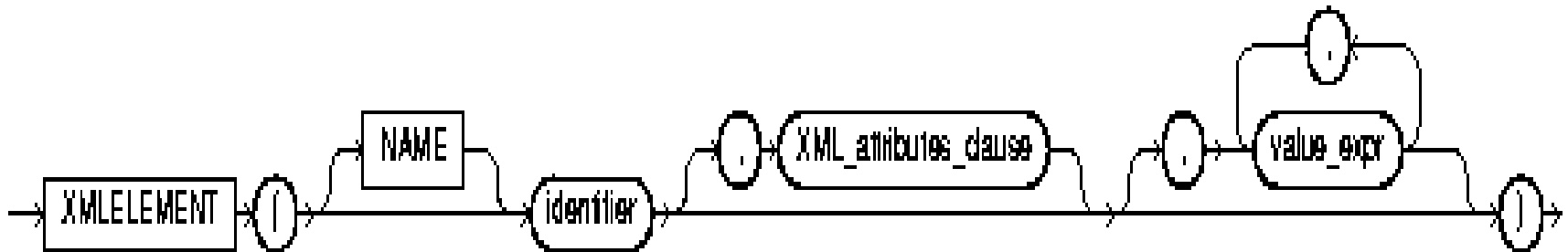
- Voor
 - Werkt altijd
 - Lage leercurve

- Tegen
 - Geen controle
 - Geen validatie
 - Niet flexibel
 - Arbeidsintensief

ISO 1EC SQL XML Functions

- XMLElement()
- XMLAttributes()
- XMLForest()
- XMLCONCAT()
- XMLAGG()

XMLELEMENT()



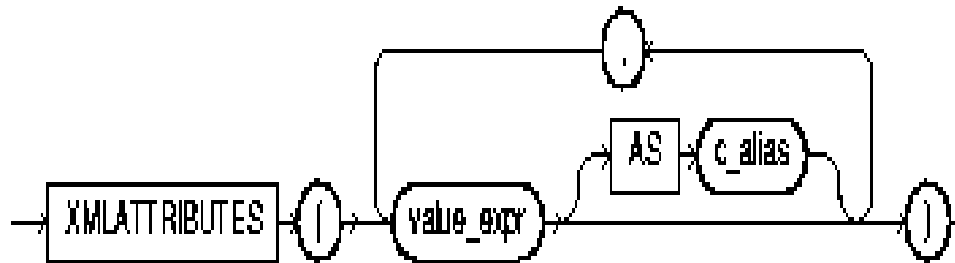
```
SELECT XMLEMENT("Lid", XMLEMENT("naam", I.voornaam || ' ' || I.achternaam),  
              XMLEMENT ("inschrijving", I.startdatum)) AS "result"  
FROM leden I  
WHERE lid_id > 200 ;
```

Lever op:

result

```
<Lid>  
  <naam>Cor Zalk</naam>  
  <inschrijving>15-02-2010</startdatum>  
</Lid>  
<Lid>  
  <naam>Martin Test</naam>  
  <inschrijving>02-01-2010</startdatum>  
</Lid>
```


XMLATTRIBUTE ()



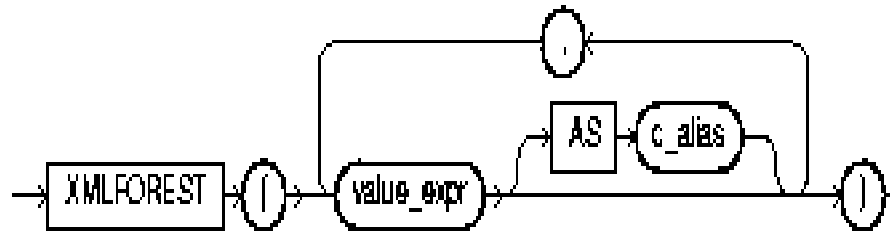
```
SELECT XMLELEMENT ( "Lid",  
    XMLATTRIBUTES (l.id,l.voornaam || ' ' || l.achternaam AS "naam")) AS "result"  
FROM Leden l  
WHERE lid_id > 200;
```

Lever op:

result

```
<Lid ID="1001" naam="Cor Zalk"/>  
<Lid ID="1206" naam="Martin Test"/>
```

XMLForest()

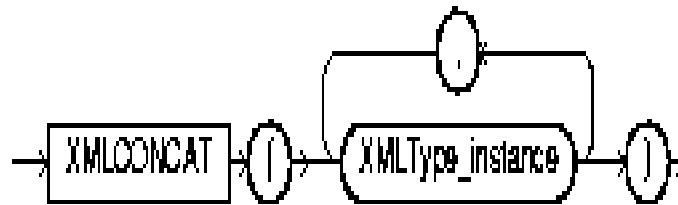


```
SELECT XMLELEMENT("Lid", XMLATTRIBUTES ( l.voornaam || ' ' || l.achternaam AS "naam" ),  
XMLForest ( l.startdatum, l.groep AS "speltak")) AS "result"  
FROM leden l;
```

Levert op:

```
<Lid name="Cor Zalk">  
  <STARTDATUM>12-12-2000</STARTDATUM>  
  <speltak>Bestuur</speltak>  
</Lid>  
<Lid name="Martin Test">  
  <STARTDATUM>01-01-1996</STARTDATUM>  
  <speltak>Leiding</speltak>  
</Lid>
```

XMLCONCAT()



```
SELECT XMLConcat ( XMLElement ("voor", l.voornaam), XMLElement ("achter", l.achternaam))  
AS "result"  
FROM leden l ;
```

Lever op:

```
<voor>Mary</voor>  
<achter>Martin</achter>
```

```
<voor>John</voor>  
<achter>Smith</achter>
```

XMLAGG()



```
SELECT XMLELEMENT("Speltak", XMLAttributes(speltak_id AS sp_id),
  XMLAGG(XMLELEMENT("Lid", l.lid_id)))
  AS "Spel_lijst"
FROM leden l
GROUP BY l.sp_id;
```

Spel_lijst

<Speltak speltak_id="BST">
 <Lid>CVZ</Lid>
</Speltak>

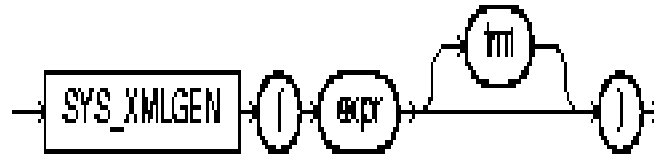
<Speltak speltak_id="LDG">
 <Lid>MT</Lid>
</Speltak>

<Speltak speltak_id="WPM">
 <Lid>JDB</Lid>
 <Lid>HDK</Lid>
 <Lid>PS</Lid>
</Speltak>

Oracle SQL/XML Functions

- SYS_XMLGEN
- XMLSEQUENCE
- SYS_XMLAGG
- XMLCOLATTVAL
- UPDATEXML
- XMLTRANSFORM
- EXTRACTVALUE

SYS_XMLGEN()

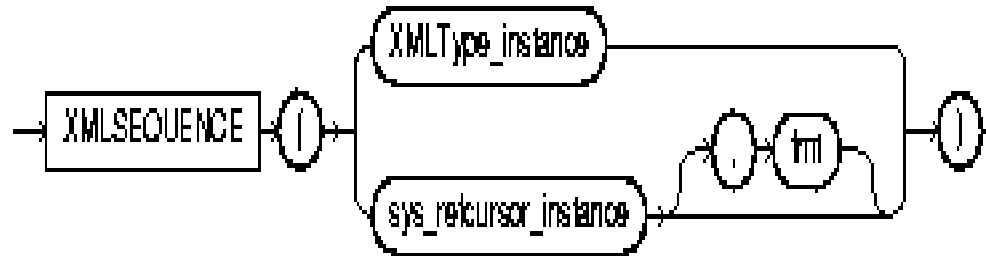


```
select sys_xmlgen(lid_id)
from leden
where rownum < 2;
```

Leveret het volgende XML dokument:

```
<?xml version="1.0"?>
<LID_ID>30</LID_ID>
```

XMLSequence()



```
SELECT value(I).getClobVal()  
FROM TABLE(XMLSequence(Cursor(SELECT * FROM leden))) I;
```

Leveret op:

XMLType

```
<ROW>  
  <LID_ID>300</LID_ID>  
  <VOORNAAM>Rob</VOORNAAM>  
</ROW>
```

```
<ROW>  
  <LID_ID>413</LID_ID>  
  <VOORNAAM>Cor</VOORNAAM>  
</ROW>
```

```
<ROW>  
  <LID_ID>968</LID_ID>  
  <VOORNAAM>Bert</VOORNAAM>  
</ROW>
```

SYS_XMLAGG

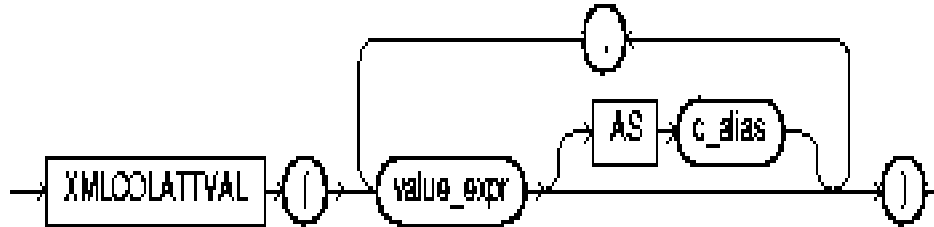


```
SELECT SYS_XMLAGG(value(l), xmlformat('LEDEN'))  
  FROM TABLE(XMLSequence(Extract(doc, '/LEDEN/LID'))) l  
 WHERE EXTRACTVALUE(value(l), '/LID/ACHTERNAAM') = 'Zalk';
```

Levert het volgende XML document:

```
<LEDEN>  
  <LID>  
    <LID_ID>112</LID_ID>  
    <VOORNAAM>Cor</VOORNAAM>  
    <ACHTERNAAM>Zalk</ACHTERNAAM>  
  </LID>  
  <LID>  
    <LID_ID>217</LID_ID>  
    <VOORNAAM>Klaas</VOORNAAM>  
    <ACHTERNAAM>Zalk</ACHTERNAAM>  
  </LID>  
</LEDEN>
```


XMLCOLATTVAL

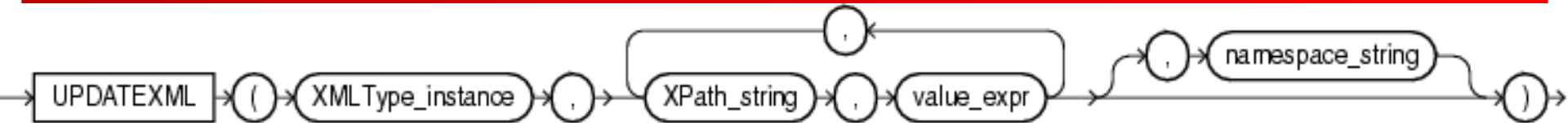


```
SELECT XMLELEMENT("Emp",XMLATTRIBUTES(l.voornaam || ' '||l.achternaam AS "naam" ),
XMLCOLATTVAL ( l.startdatum, l.spltk AS "speltak")) AS "result"
FROM leden l;
```

Lever op:

```
<Lid naam="Cor Zalk">
  <column name="STARTDATUM">12-12-2000</column>
  <column name="speltak">Bestuur</column>
</Lid>
<Lid naam="Martin Test">
  <column name="STARTDATUM">02-01-2008</column>
  <column name="speltak">Leiding</column>
</Lid>
<Lid naam="Klaas Zalk">
  <column name="STARTDATUM">16-11-2005</column>
  <column name="speltak">Bestuur</column>
</Lid>
```

UPDATEXML



```
Ledenxml
```

```
-----
```

```
leden_spec  xmltype  
lid_id      Varchar2(4)
```

```
UPDATE ledenxml  
SET leden_spec =  
    UPDATEXML(leden_spec,  
    '/Leden/Funcities/text()', 'Leiding')  
WHERE lid_id = 'CVZ';
```

XMLTRANSFORM



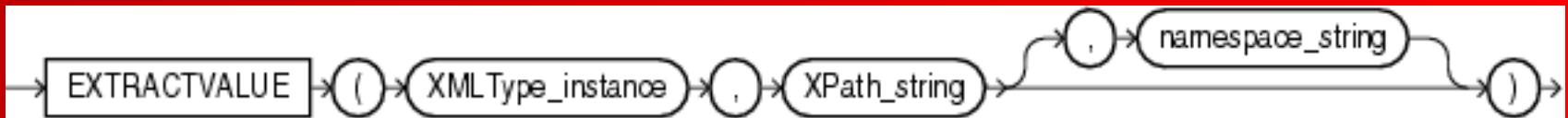
```
'<?xml version="1.0"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform" >
  <xsl:output encoding="utf-8"/>
  <xsl:template match="*">
    <xsl:copy>
      <xsl:apply-templates select="*|text()">
        <xsl:sort select="name(.)" data-type="text" order="ascending"/>
      </xsl:apply-templates>
    </xsl:copy>
  </xsl:template>
  <xsl:template match="text()">
    <xsl:value-of select="normalize-space()"/>
  </xsl:template>
</xsl:stylesheet> '
```

```
SELECT XMLTRANSFORM(l.leden_spec, x.col1).GetClobVal()
FROM ledenxml l, xsl_tabel x
WHERE l.lid_id = 'CVZ';
```

```
XMLTRANSFORM(L.LEDEN_SPEC,X.COL1).GETCLOBVAL()
```

```
-----
<Lid>
  <Achternaam>Zalk</Achternaam>
  <Speltak>Bestuur</Speltak>
  <Startdatum>15-07-2010</Startdatum>
  <Voornaam>Cor</Voornaam>
</Lid>
```

EXTRACTVALUE



```
SELECT lid_id,  
       EXTRACTVALUE(l leden_spec, '/Lid/Voornaam')  
       "Naam"  
FROM ledem l  
WHERE leden_spec IS NOT NULL;
```

```
LID_ID  Naam  
-----  
CVZ     Cor  
KVZ     Klaas  
MT      Martin
```

Verder heb je nog

- XMLGEN package
- DBMS_XML package
- DBMS_XMLDOM package
- DBMS_XMLQUERY package
- DBMS_XMLSAVE package
- Veel meer ...

XMLTYPE operations

- Existsnode()
- Extract()
- Extractvalue()
- Updatexml()

exists node()

```
SELECT extract(object_value,'PurchaseOrderReference') "REFERENCE"  
FROM PURCHASEORDER  
WHERE  
  existsNode(object_value,'PurchaseOrder[SpecialInstructions="Expedite"]')  
    = 1
```

extract()

```
SELECT extract(object_value,'PurchaseOrderReference') "REFERENCE"  
FROM PURCHASEORDER  
WHERE  
existsNode(object_value,'PurchaseOrder[SpecialInstructions="Expedite"]'  
) = 1;
```

REFERENCE

<Reference>AMCEWEN-20021009123336271PDT</Reference>

<Reference>SKING-20021009123336321PDT</Reference>

<Reference>AWALSH-20021009123337303PDT</Reference>

extractvalue()

```
SELECT extractValue(object_value,'PurchaseOrderReference')
"REFERENCE"

FROM PURCHASEORDER

WHERE
existsNode(object_value,'PurchaseOrder[SpecialInstructions="Expedite"]') =
1;
```

REFERENCE

AMCEWEN-20021009123336271PDT

SKING-20021009123336321PDT

AWALSH-20021009123337303PDT

updatexml()

```
SELECT extract(object_value,'/PurchaseOrder/Actions/Action[1]') ACTION
FROM PURCHASEORDER
WHERE
existsNode(object_value,'/PurchaseOrder[Reference="SBELL-2002100912333601PDT"]') = 1;

ACTION
-----
```

```
<Action>
  <User>SVOLLMAN</User>
</Action>
```

```
UPDATE PURCHASEORDER
SET object_value = updateXML(object_value,'/PurchaseOrder/Actions/Action[1]/User/text()','SKING')
WHERE existsNode(object_value,'/PurchaseOrder[Reference="SBELL-2002100912333601PDT"]') = 1;
```

```
SELECT extract(object_value,'/PurchaseOrder/Actions/Action[1]') ACTION
FROM PURCHASEORDER
WHERE existsNode(object_value,'/PurchaseOrder[Reference="SBELL-2002100912333601PDT"]') = 1;

ACTION
-----
```

```
<Action>
  <User>SKING</User>
</Action>
```

Best practices

- Xslt of opnieuw genereren
- XMLTYPE of Relationeel opslaan
- Xmltypeviews of packages
- Packages
- Architectuur
- Overdraagbaarheid

???