OpenLink Virtuoso

Adam Polický

OpenLink Virtuoso

- Modern multi-model RDBMS for managing data
  - tabular relations (Tables)
  - RDF sentence collections (Graphs)
- High-performance
- Scalable
- Secure
- Open
- Standards-based

More than a DBMS

- O-RDBMS
  - SQL
  - XML
  - RDF
- Web application server
- Web services platform
  - soap/rest

Installation

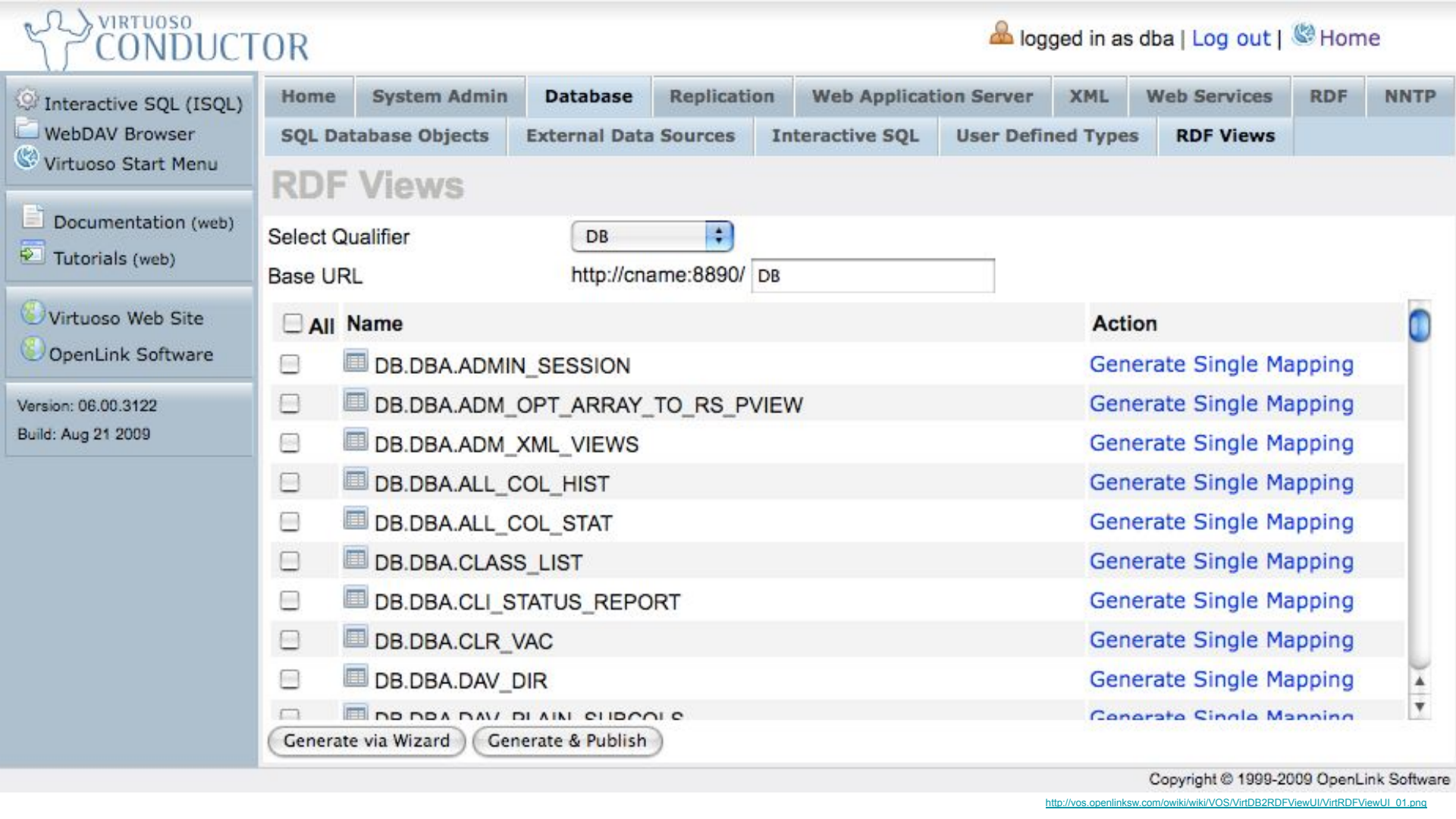
- Commercial and open-source version
- Docker
  - Image on docker hub
  - Free demo - however, after 10 minutes freezes and needs to be restarted
- Package manager
  - sudo apt install virtuoso-opensource
- Pre-built packages for Windows

Virtuoso APIs

- Both client and server APIs for system integration
- Examples
  - ADO.NET
  - JDBC
  - RDF4J API
  - RESTful HTTP API
  - SOAP webservices
  - XPath
  - XQuery
  - XSLT

GUI that can be opened in a web browser

- System administration
- Database management
  - Interactive SQL, overview of tables
- XML
  - SQL-XML, XQuery, XSLT
- Linked Data
  - RDF, relational data -> rdf conversion
- Web Server
- Web services



http://vos.openlinksw.com/owiki/wiki/VOS/VirtDB2RDFViewUI/VirtRDFViewUI_01.png

Relational data

- Traditional RDBMS
- Interactive SQL
- Visual browsing
  - database
  - schema
  - table
    - indexes, triggers, constraints
- CSV Export

SQL Create table

```
CREATE TABLE DB.FARM.WORKERS (
    WorkerID int PRIMARY KEY,
    Name varchar(255),
    Position varchar(255),
    Since date
);
```

SQL Insertion

```
INSERT INTO DB.FARM.WORKERS (WorkerID, Name, Position, Since)VALUES (1,
'John Doe', 'Farm CEO', '2001-03-04');

INSERT INTO DB.FARM.WORKERS (WorkerID, Name, Position, Since)VALUES (2,
'Bob', 'Animal Caretaker', '2010-09-23');

INSERT INTO DB.FARM.WORKERS (WorkerID, Name, Position, Since)VALUES (3,
'Alice', 'Accountant', '2013-05-14');

INSERT INTO DB.FARM.WORKERS (WorkerID, Name, Position, Since)VALUES (4,
'Earl', 'Plower', '2006-01-03');

INSERT INTO DB.FARM.WORKERS (WorkerID, Name, Position, Since)VALUES (5,
'Carl', 'Picker', '2018-12-04');
```

SQL Query

```
SELECT WorkerID AS id, Name, Position

FROM DB.FARM.WORKERS

WHERE Since < ‘2018-12-31’

ORDER BY Since ASC
```

XML documents

- XML documents can be stored on the web server
- XSLT transformation
- XQuery over the data
- SQL - XML Query

XSLT

```
<xsl:stylesheet>

  <xsl:template match="/">

    <html>

      <body>

        <h2>My CD Collection</h2>

        <table border="1">

          <tr bgcolor="#9acd32"> <th>Title</th> <th>Artist</th> <th>Price</th> </tr>

          <xsl:for-each select="catalog/cd">

            <xsl:if test="price &gt; 10">

              <tr>

                <td><xsl:value-of select="title"/></td>

                <td><xsl:value-of select="artist"/></td>

                <td><xsl:value-of select="price"/></td>

</xsl:stylesheet>
```

XQuery example

```
for $x in doc("books.xml")/bookstore/book
where $x/price>30
order by $x/title
return $x/title
```

SQL-XML Query

```
SELECT WorkerID AS Id, Name, Position
FROM DB.FARM.WORKERS AS Worker
```

```
<Workers>
  <Worker Id="1" Name="John Doe" Position="Farm CEO"/>
  <Worker Id="2" Name="Bob" Position="Animal Caretaker" />
  <Worker Id="3" Name="Alice" Position="Accountant"/>
  <Worker Id="4" Name="Earl" Position="Plower"/>
  <Worker Id="5" Name="Carl" Position="Picker"/>
</Workers>
```

Linked Data

- RDF file upload
- Possibility to publish linked data via SPARQL endpoint
- You can save your own namespaces
- Role management for graphs

Linked Data 2

- RDF data upload
  - from local file
  - from url
- Publishing on a sparql endpoint

http://localhost:8890/sparql

```
SELECT *
WHERE {
  ?s ?p ?o .
}
LIMIT 15
```

RDF Triples

```
<https://www.praha11.cz/zdroj/vzacnystrom/1> rdfs:type lgdo:Tree .

<https://www.praha11.cz/zdroj/vzacnystrom/1> foaf:age 120 .

<https://www.praha11.cz/zdroj/vzacnystrom/1> dbpedia-owl:name "Jasan
ztepilý"@cs  .

<https://www.praha11.cz/zdroj/pozice-vzacneho-stromu/1> rdfs:type
schema:Place .

<https://www.praha11.cz/zdroj/vzacnystrom/1> dbpedia-owl:location
<https://www.praha11.cz/zdroj/pozice-vzacneho-stromu/1> .

<https://www.praha11.cz/zdroj/pozice-vzacneho-stromu/1>
dbpedia-owl:district
<https://www.praha11.cz/zdroj/mestske-obvody/Praha-11-Chodov> .
```

Relational data can be converted to RDF

- linked data -> views -> generate mapping
- RDF data then can be published on a sparql endpoint
- Change in relational data can be seen on the published rdf data

http://localhost:8890/DB#

```
SELECT * WHERE { ?s ?p ?o . } LIMIT 30

SELECT DISTINCT ?s ?p1 ?o2
WHERE {
  ?s rdfs:type lgdo:Tree .
  ?s ?p1 ?o2 .
}
```