Don’t Stay Restless; Enable Your Database for REST

Pieter Van Puymbroeck
Small Fonts ahead
Some Fonts used
Will be
As small as this. so ...
About Pieter

- Dad
- Engineered systems Oracle DBA
- Sailor
- Musician

Working for Exitas Oracle Platinum Partner in BELGIUM

http://vanpupi.stepi.net
About Exitas

Quality - Passion - Personality

• Core Team of 25 dba’s
• Consultancy, managed services,…
• http://www.exitas.be
500+ Technical Experts Helping Peers Globally

3 Membership Tiers
- Oracle ACE Director
- Oracle ACE
- Oracle ACE Associate

Connect:
- oracle-ace_ww@oracle.com
- Facebook.com/oracleaces
- @oracleace

Nominate yourself or someone you know: acenomination.oracle.com

bit.ly/OracleACEProgram
Agenda

- History
- What Is REST?
- Why use it?
- Configuration
- Python
History

SOAP

http://www.soapuser.com/basics1.html
History

http://www.soapuser.com/basics1.html
Don't Stay Restless; Enable Your Database for REST: A quick start guide
Agenda

- History
- What Is REST?
- Why use it?
- Configuration
- Python
What is REST?

I had comments from well over 500 developers, many of whom were distinguished engineers with decades of experience, and I had to explain everything from the most abstract notions of Web interaction to the finest details of HTTP syntax. That process honed my model down to a core set of principles, properties, and constraints that are now called REST.
What is REST?

- Client–server architecture
- Statelessness
- Cacheability
- Layered system
- Code on demand
- Uniform interface
What is REST?

Develop in a browser.

APEX is a runtime engine built using PL/SQL in an Oracle database.

Run in a browser.

A gateway maps URIs to SQL or PL/SQL calls.
(ORDS, mod_plsql or EPG)

Use a load balancer or reverse proxy for SSL termination.

APEX uses metadata to describe your application.

What is REST?

<table>
<thead>
<tr>
<th>Action</th>
<th>Method</th>
<th>Path</th>
<th>URI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read:</td>
<td>GET</td>
<td>/hr/employees</td>
<td>GET /hr/employees</td>
</tr>
<tr>
<td>Read:</td>
<td>GET</td>
<td>/hr/employees/200</td>
<td>GET /hr/employees/200</td>
</tr>
<tr>
<td>Create:</td>
<td>POST</td>
<td>/hr/employees</td>
<td>POST /hr/employees</td>
</tr>
<tr>
<td>Update:</td>
<td>PUT</td>
<td>/hr/employees/200</td>
<td>PUT /hr/employees/200</td>
</tr>
<tr>
<td>Delete:</td>
<td>DELETE</td>
<td>/hr/employees/200</td>
<td>DELETE /hr/employees/200</td>
</tr>
</tbody>
</table>

```json
{
    "id": 200,
    "name": "Jennifer Whalen",
    "issenior": "true",
    "jobhistory": [
        {
            "id": "AD_ASST",
            "departmentid": 90,
            "startdate": "17-SEP-1995",
            "enddate": "17-JUN-2001"
        },
        {
            "id": "AC_ACCOUNT",
            "departmentid": 90,
            "startdate": "01-JUL-2002",
            "enddate": "31-DEC-2006"
        }
    ]
}
```
Agenda

- History
- What is it?
- Why use it?
- Configuration
- Python
Don't Stay Restless; Enable Your Database for REST: A quick start Guide

Why use it?
Why use it?

Read: GET /hr/employees
Read: GET /hr/employees/200
Create: POST /hr/employees
Update: PUT /hr/employees/200
Delete: DELETE /hr/employees/200

{ "id": 200,
"name": "Jennifer Whalen",
"issenior": "true",
"jobhistory": [

{ "id": "AD_ASST",
"departmentid": 90,
"startdate": "17-SEP-1995",
"enddate": "17-JUN-2001"
},

{ "id": "AC_ACCOUNT",
"departmentid": 90,
"startdate": "01-JUL-2002",
"enddate": "31-DEC-2006"
}]
}
Agenda

- History
- What Is It?
- Why Use It?
- Configuration
- Python
Configuration

https://docs.oracle.com/cd/E56351_01/index.htm

Oracle Database (Relational)

Oracle Database 12c (Document Store)

Oracle NoSQL Database

REST

Oracle REST Data Service (WLS, Glassfish, Tomcat)
Configuration
Configuration

Assumption: Apex is already installed in the Database

• Unlock DB-users:
  • APEX_LISTENER
  • APEX_PUBLIC_USER
  • APEX_REST_PUBLIC_USER

Check also: https://oracle-base.com/articles/misc/oracle-rest-data-services-ords-installation-on-tomcat
Configuration

- Ords install
  - Unzip the software

  $ mkdir /u01/app/oracle/product/ords/
  $ cd /u01/app/oracle/product/ords/
  $ unzip /tmp/<whatever you named the zipfile>.zip
Configuration

- Images
  - `mkdir $CATALINA_HOME/webapps/i/`
  - `cp -R /tmp/apex/images/* $CATALINA_HOME/webapps/i/`

- Ords.war
  - `cp $ORDS_HOME/ords.war $CATALINA_HOME/webapps/`

http://<server-name>:<port>/ords/
Configuration

[root@laboel74 webapps]# java -jar ords.war setup --database mypdb
This Oracle REST Data Services instance has not yet been configured.
Please complete the following prompts

Enter the location to store configuration data:/u01/app/oracle/product/18.0.0.0/ords/conf
Enter the name of the database server [localhost]: 192.168.56.4
Enter the database listen port [1521]:
Enter 1 to specify the database service name, or 2 to specify the database SID [1]:
Enter the database service name: mypdb
Enter 1 if you want to verify/install Oracle REST Data Services schema or 2 to skip this step [1]:
Enter the database password for ORDS_PUBLIC_USER:
Confirm password:
Requires SYS AS SYSDBA to verify Oracle REST Data Services schema.

Enter the database password for SYS AS SYSDBA:
Confirm password:
Configuration

Retrieving information.
Enter the default tablespace for ORDS_METADATA [SYSAUX]:
Enter the temporary tablespace for ORDS_METADATA [TEMP]:
Enter the default tablespace for ORDS_PUBLIC_USER [SYSAUX]:
Enter the temporary tablespace for ORDS_PUBLIC_USER [TEMP]:
Enter 1 if you want to use PL/SQL Gateway or 2 to skip this step.
If using Oracle Application Express or migrating from mod_plsql then you must enter 1 [1]:
Enter the PL/SQL Gateway database user name [APEX_PUBLIC_USER]:
Enter the database password for APEX_PUBLIC_USER:
Confirm password:
Enter 1 to specify passwords for Application Express RESTful Services database users (APEX_LISTENER, APEX_REST_PUBLIC_USER) or 2 to skip this step [1]: 2

Oct 17, 2018 12:13:26 PM
INFO: reloaded pools: []
Installing Oracle REST Data Services version 18.3.0.r2701456
... Log file written to /root/ords_install_core_2018-10-17_121327_00509.log
... Verified database prerequisites
... Created Oracle REST Data Services schema
... Created Oracle REST Data Services proxy user
... Granted privileges to Oracle REST Data Services
... Created Oracle REST Data Services database objects
... Log file written to /root/ords_install_datamodel_2018-10-17_121411_00630.log
... Log file written to /root/ords_install_apex_2018-10-17_121415_00705.log
Completed installation for Oracle REST Data Services version 18.3.0.r2701456. Elapsed time: 00:00:51.976

[root@laboel74 webapps]#
Configuration

[root@laboel74 webapps]# java -jar ords.war map-url --type base-path /mypdb mypdb
Oct 17, 2018 12:17:04 PM
INFO: Creating new mapping from: [base-path,/mypdb] to map to: [mypdb, null, null]
[root@laboel74 webapps]#

http://<server-name>:<port>/ords/mypdb
Configuration

- Defaults.xml - Removed these entries (DEMO only!!! )
  - <entry key="db.connectionType">basic</entry>
  - <entry key="db.port">1521</entry>
  - <entry key="db.serviceNameSuffix">.***.oraclecloud.internal</entry>
  - <entry key="db.servicename">****.****.oraclecloud.internal</entry>

[root@laboel74 webapps]# java -jar ords.war configdir
Oct 17, 2018 12:18:21 PM
INFO: The config.dir value is /u01/app/oracle/product/18.0.0.0/ords/conf
[root@laboel74 webapps]#
Configuration

• Other XML files ($ORDS_HOME/conf)

```
[root@laboe174 conf]# ls -l
total 32
-rw-r--r-- 1 root root  606 Aug 31 14:20 mypdb_pu.xml
-rw-r--r-- 1 root root  707 Aug 31 14:21 mypdb.xml
[root@laboe174 conf]#
```
Configuration

https://docs.oracle.com/cd/E56351_01/doc.30/e87809/about-REST-configuration-files.htm#AELIG7201

Settings specific to a particular database connection (for example, the default apex connection) are stored in conf/\(\langle\text{db-name}\rangle\).xml, where \(\langle\text{db-name}\rangle\) is the name of the database connection.

If the database connection uses Oracle Application Express RESTful Services, the files with names including _al.xml, _rt.xml, and _pu.xml store the configuration for the APEX_LISTENER, APEX_REST_PUBLIC_USER, and ORDS_PUBLIC_USER database users, respectively.

If the database connection uses Oracle REST Data Services RESTful Services, the file \(<\text{db-name}>_\text{pu}.xml\) stores the configuration for the ORDS_PUBLIC_USER database user.
Configuration

- Apex.xml

```
[root@appsrv conf]# cat mypdb.xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE properties SYSTEM "http://java.sun.com/dtd/properties.dtd">
<properties>
  <entry key="db.username">apex_public_user</entry>
  <entry key="db.password">@056CA4****8AD4</entry>
  <entry key="db.connectionType">customURL</entry>
  <entry key="db.customURL">jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_LIST=(LOAD_BALANCE=OFF)
  (FAILOVER=ON)(ADDRESS=(PROTOCOL=TCP) (HOST= <primary IP>)(PORT=1521))(ADDRESS=(PROTOCOL=TCP)
  (HOST= <Secondary IP>)(PORT=1521)))
  (CONNECT_DATA=(SERVICE_NAME=mypdb.****.oraclecloud.internal))</entry>
</properties>
[root@appsrv conf]#```
Configuration

- ApexPu.xml

```
[root@appsrv conf]# cat mypdb_pu.xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE properties SYSTEM "http://java.sun.com/dtd/properties.dtd">
<properties>
    <entry key="db.username">ORDS_PUBLIC_USER</entry>
    <entry key="db.connectionType">customurl</entry>
    <entry key="db.customURL">jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_LIST=(LOAD_BALANCE=OFF)(FAILOVER=ON)(ADDRESS=(PROTOCOL=TCP) (HOST= <primary IP>)(PORT=1521))(ADDRESS=(PROTOCOL=TCP) (HOST= <Secondary IP>)(PORT=1521))) (CONNECT_DATA=(SERVICE_NAME=mypdb.****.oraclecloud.internal)))</entry>
</properties>
[root@appsrv conf]#
```
Don’t Stay Restless; Enable Your Database for REST: A quick start Guide

**Configuration**

- **The schema**

```sql
BEGIN
    ORDS.enable_schema(
        p_enabled => TRUE,
        p_schema => 'VANPUPI',
        p_url_mapping_type => 'BASE_PATH',
        p_url_mapping_pattern => 'vanpupirest',
        p_auto_rest_auth => TRUE
    );

    COMMIT;
END;
/

http://laboel74:8080/ords/mypdb/vanpupirest
```
Configuration

• The table

BEGIN
    ORDS.enable_object (
        p_enabled => TRUE,
        p_schema  => 'VANPUPI',
        p_object  => 'TEST',
        p_object_type => 'TABLE',
        p_object_alias => 'testtable'
    );

    COMMIT;
END;
/

http://laboel74:8080/ords/mypdb/vanpupirest/testtable
Configuration

http://laboel74:8080/ords/mypdb/vanpupirest/testtable
Configuration

• For a new schema, it needs to be enabled

BEGIN
  ORDS.enable_schema(
    p_enabled => TRUE,
    p_schema => 'VANPUPI_INS',
    p_url_mapping_type => 'BASE_PATH',
    p_url_mapping_pattern => 'insrest',
    p_auto_rest_auth => TRUE
  );

  COMMIT;
END;
/

http://laboel74:8080/ords/mypdb/insrest
Configuration

• The table

create table vanpupi_ins.t_test(edate date, msg varchar2(4000));
Configuration

• The procedure

```sql
create or replace procedure vanpupi_ins.p_testrest(Arg1A in varchar2)
is
begin
    insert into vanpupi_ins.t_test values (sysdate,Arg1A);
end;
/
```
Configuration

• The module

```ruby
BEGIN
  ORDS.define_module(
    p_module_name => 'insmod',
    p_base_path => 'insmod/',
    p_items_per_page => 0);
end;
/

laboel74:8080/ords/mypdb/insrest/insmod/
```
Configuration

• The template

```plaintext
begin
    ORDS.define_template(
        p_module_name    => 'insmod',
        p_pattern        => 'insrest/');
end;
/
```

laboei74:8080/ords/mypdb/insrest/insmod/insrest/
Configuration

• The handler

```sql
begin

ORDS.define_handler(
    p_module_name   => 'insmod',
    p_pattern       => 'insrest/',
    p_method        => 'POST',
    p_source_type   => ORDS.source_type_plsql,
    p_source        => 'BEGIN
        VANPUPI_INS.p_testrest(
            Arg1A => :Arg1A
        );
        END;',
    p_items_per_page => 0);
end;
/

Commit;
```

Don't Stay Restless; Enable Your Database for REST: A quick start Guide
Configuration

- The payload

  mbp-vanpupi:~ pieter$ cat /tmp/ins_rest_test.json
  {
  "Arg1A": "This is a string"
  }
  mbp-vanpupi:~ pieter$
Configuration

- The curl request

```
mbp-vanpupi:~ pieter$ curl -i -X POST --data-binary @/tmp/ins_rest_test.json \
-H "Content-Type: application/json" \
http://192.168.56.4:8080/ords/mypdb/insrest/insmod/insrest/
```

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Transfer-Encoding: chunked
Date: Mon, 22 Oct 2018 18:22:48 GMT

mbp-vanpupi:~ pieter$
Configuration

- The proof

```sql
select * from t_test;
```

All Rows Fetched: 1 in 0.002 seconds

<table>
<thead>
<tr>
<th></th>
<th>EDATE</th>
<th>MSG</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>22/10/18</td>
<td>This is a string</td>
</tr>
</tbody>
</table>
Agenda

- History
- What is it?
- Why use it?
- Configuration
- Python
Python - Table test

```python
>>> import requests
>>> import json
>>> r = requests.get(http://laboel74:8080/ords/mypdb/vanpupirest/testtable')
>>> jo = r.json()
>>> print(json.dumps(jo, indent=4))
```
Don't Stay Restless; Enable Your Database for REST: A quick start Guide

Python - Table test

[oracle@laboel74 ~]$ python
Python 2.7.5 (default, May 29 2017, 20:42:36)
[GCC 4.8.5 20150623 (Red Hat 4.8.5-11)] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import requests
>>> import json
>>> r = requests.get('http://laboel74:8080/ords/mypdb/vanpupirest/testtable')
>>> jo = r.json()
>>> print(json.dumps(jo, indent=4))
{
   "count": 1,
   "links": [
   {
      "href": "http://laboel74:8080/ords/mypdb/vanpupirest/testtable/",
      "rel": "self"
   }
   ],
   "hasMore": false,
   "offset": 0,
   "limit": 25
}
>>> quit()
[oracle@laboel74 ~]$
>>> import requests
>>> import json
>>> import pprint

>>> data = {"Arg1A": "This is a python string"}
>>> data_json = json.dumps(data)

>>> url = 'http://192.168.56.4:8080/ords/mypdb/insrest/insmod/insrest/'
>>> headers = {'Content-type': 'application/json'}

>>> response = requests.post(url, data=data_json, headers=headers)

>>> pprint.pprint(response)
Don't Stay Restless; Enable Your Database for REST: A quick start Guide

Python - PL/SQL test

mbp-vanpupi:~ pieter$ python
Python 2.7.10 (default, Oct 6 2017, 22:29:07)
[GCC 4.2.1 Compatible Apple LLVM 9.0.0 (clang-900.0.31)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> import requests
>>> import json
>>> import pprint

>>> data = {"Arg1A": "This is a python string"}
>>> data_json = json.dumps(data)
>>> url = 'http://192.168.56.4:8080/ords/mypdb/insrest/insmod/insrest/'
>>> headers = {'Content-type': 'application/json'}
>>> response = requests.post(url, data=data_json, headers=headers)
>>> pprint.pprint(response)
<Response [200]>

mbp-vanpupi:~ pieter$
```sql
SELECT * FROM t_test;
```

<table>
<thead>
<tr>
<th>EDATE</th>
<th>MSG</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>22/10/18 This is a string</td>
</tr>
<tr>
<td>2</td>
<td>22/10/18 This is a python string</td>
</tr>
</tbody>
</table>
Agenda

- History
- What Is REST?
- Why use it?
- Configuration
- Python
Mail: pieter.vanpuymbroeck@exitas.be
Twitter: @vanpupi