Automate DBA Tasks With Ansible

Automation

Ivica Arsov – October 19, 2017

Pythian

Ivica Arsov

Database Consultant

- Oracle Certified Master 12c & 11g
- Oracle ACE Associate
- Blogger

Twitter: IvicaArsov

Blog: https://iarsov.com

LOVE YOUR DATA

ORACLE ACE Associate

ORACLE Certified Master
Oracle Database 12c Administrator

ORACLE Certified Master
Oracle Database 11g Administrator

ORACLE Certified Expert
Oracle Exadata X3 and Oracle Exadata X4 Administrator

ORACLE Certified Expert
Oracle Real Application Clusters 11g and Grid Infrastructure Administrator

ABOUT PYTHIAN

Pythian's 400+ IT professionals help companies adopt and manage disruptive technologies to better compete

PYTHIAN MAKES DIGITAL TRANSFORMATION EASIER AND IT OPERATIONS MORE RELIABLE

Expert disruptors for
innovation and transformation

Trusted protectors for
ongoing delivery

TECHNICAL EXPERTISE

Big Data: Harnessing the transformative power of data on a massive scale

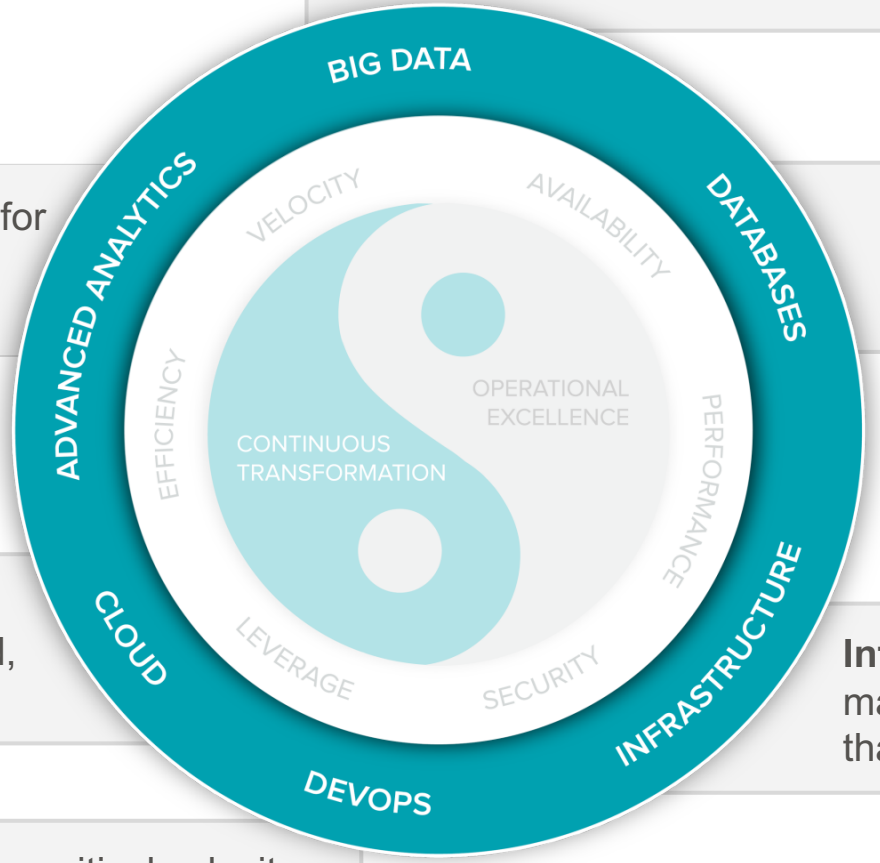
Advanced Analytics: Mining data for insights & business transformation using data science

Databases: Ensuring databases are reliable, secure, available and continuously optimized

Cloud: Using the disruptive nature of cloud for accelerated, cost-effective growth

Infrastructure: Transforming and managing the IT infrastructure that supports the business

DevOps: Providing critical velocity in software deployment by adopting DevOps practices



AGENDA

- Introduction to Ansible
- Installation
- Playbooks
- Roles
- Templates
- Modules
- Custom modules

first things first … why automation ?

some considerations

- Manage multiple servers (> 20-30)
- Multiple people are doing same thing differently
- Different configurations
- Forgotten steps/checks for long action plans
- Automation reduces human errors

etc…

Ansible introduction and installation

ansible engine

- Open - source automation engine
- Owned by RedHat
  - Available for Red Hat Enterprise Linux, CentOS, Debian, Ubuntu, OEL …
  - Windows support - only as a target machine

- Written in Python
- Agentless architecture

- Git repository: https://github.com/ansible/ansible

ansible consist of …

architecture

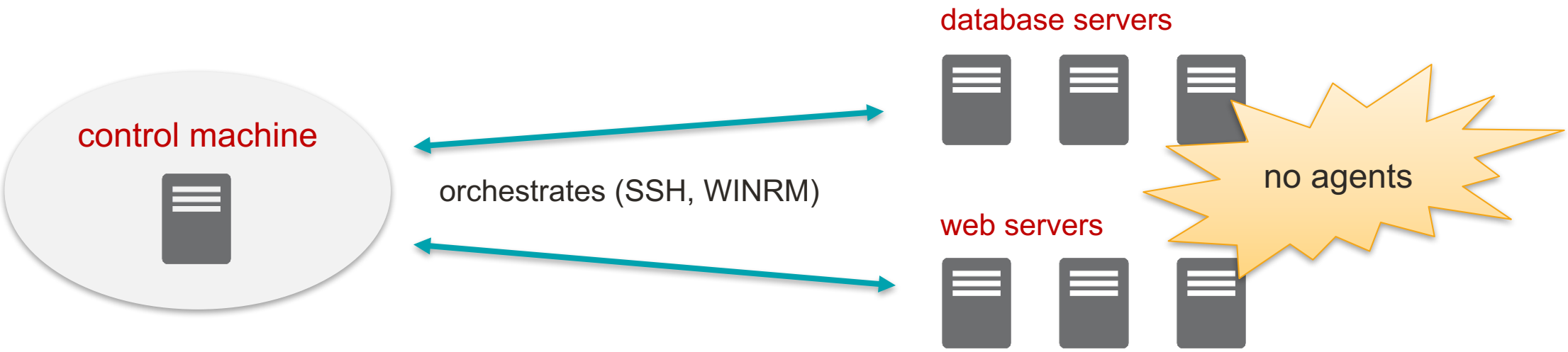
- Inventory configuration
- Modules
- Playbooks
- Roles

ansible architecture

architecture

- Two types of servers: controlling machines and nodes (targets)

- Targets are managed by a controlling machine over SSH, WINRM



ansible execution

- The modules are copied and executed on target machines
- Destination can be controlled with `local_tmp` in `ansible.cfg`

Execution steps:

1. Create local temporary directory: `$HOME/.ansible/tmp/ansible-tmp-xxx/ansiballz_cache`
2. Copy module
3. Execute module
4. Return result in JSON format
5. Clear (remove) `ansible-tmp-xxx/ansiballz_cache`

ansible installation

- RPMs for Enterprise Linux 6, 7 are available from yum via EPEL

http://fedoraproject.org/wiki/EPEL

- Add EPEL on OEL7, RHEL7 or CentOS

```
wget https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
yum install epel-release-latest-7.noarch.rpm
```

- Install Ansible

```
yum install ansible
```

ansible installation

OSX

- Installation is done via pip
- Install Xcode
- Install pip and ansible

```
sudo easy_install pip
pip install ansible
```

- For more information see:

http://docs.ansible.com/ansible/intro_installation.html#latest-release-via-yum

ansible configuration files

- Ansible configuration file: `/etc/ansible/ansible.cfg`

- Define the hosts in a 'hosts' file, by default in `/etc/ansible/hosts`

  List all defined hosts: **`ansible all --list-hosts`**

```
...

db8.example.org

[webservers]
app1.example.org
app2.example.org

[dbservers]
db1.example.org ansible_host=... ansible_port=... ansible_ssh_private_key_file=... ansible_user=...
db2.example.org
```

ansible – windows support

- Control machine requires **pywinrm**, a Python module for the Windows Remote Management (WinRM)

Option	Local Accounts	Active Directory Accounts	Credential Delegation
Basic	Yes	No	No
Certificate	Yes	No	No
Kerberos	**No**	**Yes**	**Yes**
NTLM	Yes	Yes	No
CredSSP	Yes	Yes	Yes

ansible – windows support

example configuration with kerberos

- Kerberos configuration file `/etc/krb5.conf`

```
...

[realms]
 WINGROUP.AD = {
  kdc = win-hrms-srv.WINGROUP.AD
  admin_server = win-hrms-srv.WINGROUP.AD
  default_domain = WINGROUP.AD
}

[domain_realm]
 .wingroup.ad = WINGROUP.AD
 wingroup.ad = WINGROUP.AD
```

ansible – windows support

‘hosts’ file

- Inventory/hosts

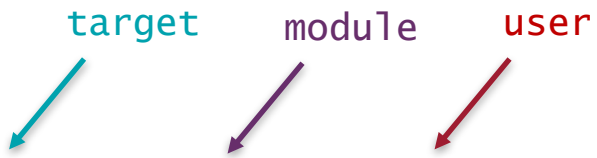
```
...

[windows]
win-hrms-srv.wingroup.ad

[windows:vars]
ansible_user = iarsov@WINGROUP.AD
ansible_password = ********
ansible_connection = winrm
ansible_port = 5986
ansible_winrm_server_cert_validation =ignore
```

Demo

Ansible installation

ad-hoc task execution

- The “ping” example
- Ad-hoc tasks are run with `ansible` command
- Ansible uses SSH authentication, you need to specify an user




```
ansible vm12r1 -m ping -u root -k
SSH password:
vm12r1 | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
```

Playbooks

YAML basic rules

YAML Ain't Markup Language

- Human readable data structure
- Less complex than JSON or XML
- YAML is case sensitive
- Does not allow tabs. You must use spaces.

YAML

```
db_list:
 - oracle_owner: oracle
   oracle_home: /u01/app...
   sid: orcl
 ...
```

XML

```
<db_list>
  <oracle_owner>oracle</oracle_owner>
  <oracle_home>/u01/app...</oracle_home>
  <sid>orcl</sid>
  ...
</db_list>
```

playbooks

- Organize the “play” tasks
- Playbook’s language is YAML
- Each play contains set of tasks
- Tasks are executed one at a time in order against all matched targets
- Command `ansible-playbook`

Play example:

`playbook.yml`

```
- hosts: dbservers
  user: root
  tasks:
   - name: ping hostname
     ping:
   - name: create directory
     file: path=/home/ansible/poug2017 state=directory
```

playbooks

```
- hosts: dbservers
   user: root
   tasks:
    - name: ping hostname
      ping:

 - hosts: webservers
   user: root
   tasks:
    - name: create directory
      file: path=/home/ansible/poug2017 state=directory
```

- Playbooks can contain multiple plays
- Split tasks performed per different host groups

variables

using variables

- Variables are used using the Jinja2 templating system
- You can also use variables in Templates
- Variables are referenced with double curly brackets: `{{ variable_name }}`

Variables file definition (`var_def.yml`)

```
hostname: ansible-demo

dir_path: /home/ansible/poug2017

user: root
```

Playbook file definition

```
- hosts: "{{ hostname }}"
  user: "{{ user }}"
  vars_files:
   - var_def.yml
  tasks:
   - name: create directory
     file: path={{ dir_path }} state=directory
```

variables

- Variables can be defined within the playbook

```
- hosts: "{{ hostname }}"
  user: "{{ user }}"
  tasks:
   - name: create directory
     file: path={{ dir_path }} state=directory
  vars:
    - hostname: ansible-demo
      user: root
      dir_path: /home/ansible/poug2017
```

variables

- Variables can also be defined from command line

```
ansible-playbook demo2.yml -k --extra-vars="dir_path=/home/ansible/poug2017 user=root
hostname=ansible-demo"

SSH password:

PLAY [ansible-demo]
*********************************************************************************

ok: [ansible-demo]

TASK [create directory]
*********************************************************************************

ok: [ansible-demo]

PLAY RECAP
*********************************************************************************

ansible-demo               : ok=2    changed=0    unreachable=0    failed=0
```

ansible – windows support

- Playbook wintest.yml

```
---
  - name: Windows demo
    hosts: windows
    tasks:
     - name: Create directory "D:\ansible\poug2017"
       win_command: Powershell.exe "mkdir D:\ansible\poug2017"
```

ansible – windows support

playbook execution

```
ansible-playbook wintest.yml

PLAY [Windows demo] ************************************************************

TASK [Gathering Facts] *********************************************************
ok: [win-hrms-srv.wingroup.ad]

TASK [Create directory "D:\ansible\poug2017"] ***********************************
changed: [win-hrms-srv.wingroup.ad]

PLAY RECAP *********************************************************************
win-hrms-srv.wingroup.ad   : ok=2    changed=1    unreachable=0    failed=0
```

Roles and Templates

roles

how to organize the tasks?

- Represent “automation” within ansible
- You can define variables for all roles or per role
- Not complex at all. No hidden magic.
- Supports automatic load of *main.yml* file for tasks, handlers, variables, meta definitions and role “default” variables

project structure

dbservers role

```
site.yml
dbswinstall.yml
roles/
      dbinstall/
              files/
              templates/
              tasks/
                    main.yml
              handlers/
              vars/
                    main.yml
              defaults/
...
```

playbook

role

tasks definitions

variable definitions

`dbswinstall.yml` snippet:

```
---
   - hosts: chicago
     user: oracle
     roles:
         - hostconfig
         - dbinstall
         - dbmonitor
```

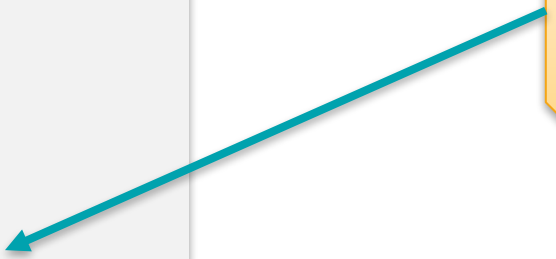
**main.yml* files are automatically loaded

role dependencies

- Role dependencies are defined in *meta/main.yml* file
- Allows you to reference other roles before running a role

```
...
dbswinstall.yml
roles/
        common/
        dbinstall/
                files/
                meta/
                     main.yml
                tasks/

                ...
```

```
---
  dependencies:
    - { role: common }
```



Templates

Introduction to ansible templating

- Sometimes pre-defined configuration files are needed for installation
- An example is the Oracle dbca configuration file
- Templating allows us to use ansible variables substation within templates

Templates

```
...
#-------------------------------------------------------------------------------
# Specify the hostname of the system as set during the install. It can be used
# to force the installation to use an alternative hostname rather than using the
# first hostname found on the system. (e.g., for systems with multiple hostnames
# and network interfaces)
#-------------------------------------------------------------------------------
ORACLE_HOSTNAME={{ ansible_hostname }}

#-------------------------------------------------------------------------------
# Specify the Unix group to be set for the inventory directory.
#-------------------------------------------------------------------------------
UNIX_GROUP_NAME={{ oracle_group }}
...
```

Templates

- Source template file is copied to target destination and variables are substituted

```
- name: create 11g installer response file
  template:
      src: db_11204.rsp.j2
      dest: /tmp/db_11204.rsp
      owner: "{{ oracle_user }}"
      group: "{{ oracle_group }}"
      mode: 0640
```

Demo

Example: Oracle 11.2.0.4 software installation

oracle database install

installation of oracle 11.2.0.4

```
oracle_base: "/oracle/app/oracle"
oracle_home: "/oracle/app/oracle/product/11.2.0.4/db1"
oracle_user: oracle
oracle_group: dba
swlib_path: "/oracle/install"
db_edition: EE
DBComponents: "oracle.rdbms.partitioning:11.2.0.4.0"
```

oracle database install

installation of oracle 11.2.0.4

```
- name: create oracle base directory
    file:
      path: "{{ oracle_base }}"
      state: directory
      owner: "{{ oracle_user }}"
      group: "{{ oracle_group }}"
      mode: 0775
```

oracle database install

installation of oracle 11.2.0.4

```
- name: create 11g installer response file
    template:
      src: db_11204.rsp.j2
      dest: /tmp/db_11204.rsp
      owner: "{{ oracle_user }}"
      group: "{{ oracle_group }}"
      mode: 0640
```

continued

installation of oracle 11.2.0.4

```
- name: install base software
    command: "{{ swlib_path }}/11204/installer/database/runInstaller -silent
-ignorePrereq -ignoreSysPrereqs -waitforcompletion -responseFile
/tmp/db_11204.rsp"
    register: install_db_software
    args:
      creates: "{{ oracle_home }}"
    failed_when: "'skipped' not in install_db_software.stdout and
'Successfully Setup Software.' not in install_db_software.stdout“
```

continued

installation of oracle 11.2.0.4

```
- name: run root-script orainstRoot.sh
    command: "{{ oracle_base }}/../oraInventory/orainstRoot.sh"
    when: "'skipped' not in install_db_software.stdout"
    become: true
    become_user: root
    become_method: sudo
```

continued

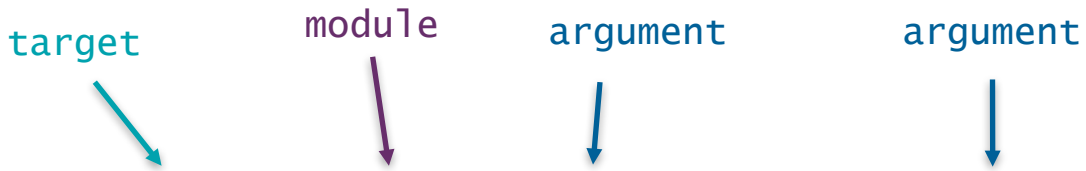
installation of oracle 11.2.0.4

```
- name: run root-script root.sh from ORACLE_HOME
    command: "{{ oracle_home }}/root.sh"
    when: "'skipped' not in install_db_software.stdout"
    become: true
    become_user: root
    become_method: sudo

  - name: clean up installer response file
    file:
      path: /tmp/db_11204.rsp
      state: absent
```

Modules

modules

- Library plugins , always are executed on target machine
- Ansible comes with large number of modules
- Each module supports specific number of arguments




```
ansible vm12r1 -m copy -a "src=/tmp/file.txt dest=/tmp/file.txt"
SSH password:
vm12r1 | SUCCESS => {
    "changed": true,
    "checksum": "da39a3ee5e6b4b0d3255bfef95601890afd80709",
    "dest": "/tmp/file.txt",
    "gid": 500,
    "group": "dba",
    ...
}
```

custom modules

developing modules

- First check if similar module already exist
  - http://docs.ansible.com/ansible/latest/list_of_all_modules.html
- GitHub (all module updates): https://github.com/ansible/ansible/labels/module
- Not enough documentation regarding development
- If you want additional checks on control machine use *action_plugins*

our first custom module

developing modules

- Test module

```
ansible ansible-demo --module-path=/home/ansible -m date

ansible-demo | SUCCESS => {
        "changed": false,
        "time": "2017-08-31 14:48:10.978621”
}
```

`date.py` code

```
#!/usr/bin/python

import datetime
import json

date = str(datetime.datetime.now())
print(json.dumps({
    "time" : date
}))
```

orapatch

custom module

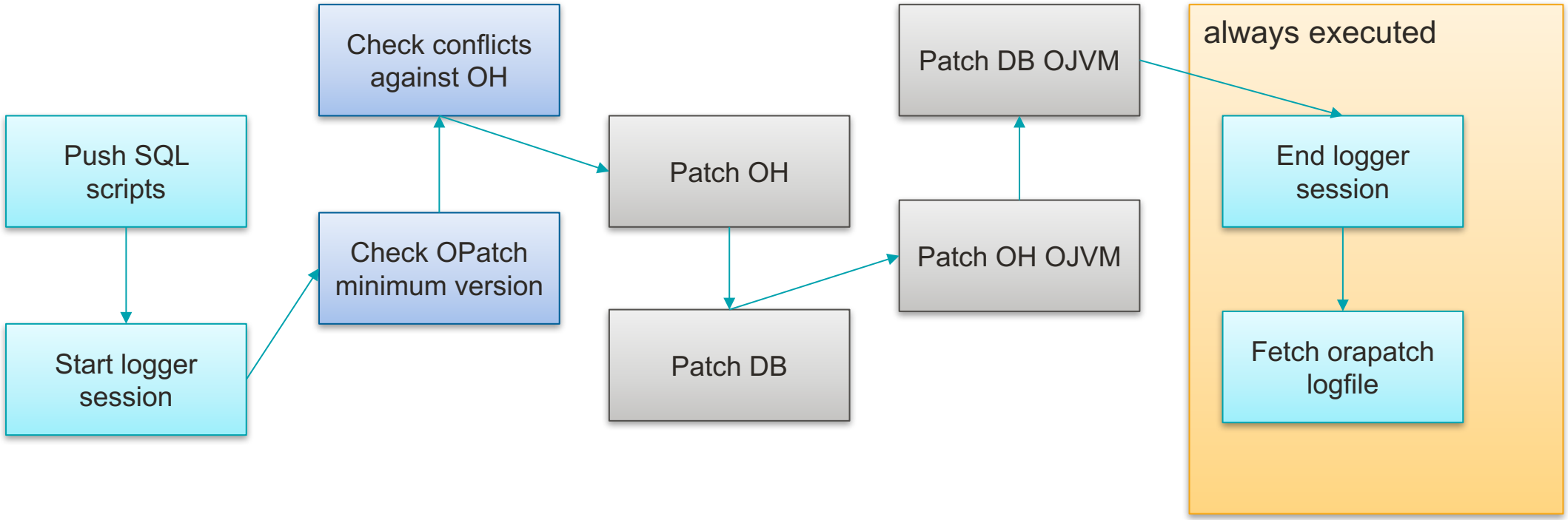
- Can be used to apply PSU on 10g, 11g and 12c

In summary:

- You need to specify:
  - oracle home path, PSU or DBBP patch ID, whether to patch only the oracle home binaries, all databases or specific set of databases
- It will automatically:
  - stop OH services, apply PSU or DBBP according specified settings, start back the services as before the patching

orapatch

workflow



orapatch

vars/main.yml

```
db_list:
    - oracle_owner: oracle
      oracle_home_path: /oracle/app/oracle/product/12.2.0.1/db1
      patch_directory: "12.1.0.2/psu"
      run_only_checks: False
      patch: True
      patch_id: 26129945
      patch_only_oh: True
      patch_ojvm: False
      patch_db_all: False
      patch_db_list:
        - '{ "dbname": "orcl" }'
```

oracle home to patch

directory where patch binaries are located

run only prerequisite checks

patch only oracle specified oracle home

apply OJVM PSU

patch all databases for specified OH

patch specific databases

```
patch_dict:

     26129945:
        patch_proactive_bp_id: 26129945
        patch_gi_id:
        patch_db_id: 25983138
        patch_ocw_id: 26187629
        patch_ojvm_id:
        patch_acfs_id:
        patch_dbwlm_id:
        patch_dir: 26129945
        file: p26129945_121020_Linux-x86-64.zip
        only_oh: False
        desc: "DATABASE PROACTIVE BUNDLE PATCH 12.2.0.1.170620”

     26550339:
        patch_proactive_bp_id:
        patch_gi_id: 26610308
        ...
```

```
- name: Patch oracle software
  serial: 1
  vars_prompt:
    - name: "root_password"
      prompt: "Enter root password (press enter to skip)"
      private: yes
    - name: "root_password_confirm"
      prompt: "Enter root password again (press enter to skip)"
      private: yes
  pre_tasks:
    - assert:
        that: root_password == root_password_confirm
        msg: "Root password missmatch."
  hosts: database
  user: oracle
  roles:
    - orapatch
```

```
[ansible@ansible-control oracle-ansible-pythian]$ ansible-playbook orapatch.yml -k -K
SSH password:
SUDO password[defaults to SSH password]:


Enter root password (press enter to skip):
Enter root password again (press enter to skip):

PLAY [Patch oracle software] ***************************************************************************************

TASK [Gathering Facts] *********************************************************************************************
ok: [rac-srv1]

TASK [assert] ******************************************************************************************************
ok: [rac-srv1] => {
    "changed": false,
    "msg": "All assertions passed"
}

TASK [orapatch : [SYSTEM] Include vars] ****************************************************************************
ok: [rac-srv1]

TASK [orapatch : [SYSTEM] Push sql scripts] ************************************************************************
changed: [rac-srv1]

TASK [orapatch : [SYSTEM] Ensure 'orapatch' log file exists] *******************************************************
changed: [rac-srv1]

TASK [orapatch : [SYSTEM] Start logger session] ********************************************************************
 [WARNING]: Module did not set no_log for root_password

ok: [rac-srv1]
```

```
TASK [orapatch : Check OPatch minimum version] *********************************************************************************
ok: [rac-srv1] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'oracle_owne
39, u'patch_db_all': False, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_file': None})
ok: [rac-srv1] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'oracle_owne
atch_id': 26550339, u'patch_db_all': True, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_fil

TASK [orapatch : Check conflicts against OH] ***********************************************************************************
ok: [rac-srv1] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'oracle_owne
39, u'patch_db_all': False, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_file': None})
ok: [rac-srv1] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'oracle_owne
atch_id': 26550339, u'patch_db_all': True, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_fil

TASK [orapatch : Patch OH] *****************************************************************************************************
ok: [rac-srv1] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'oracle_owne
39, u'patch_db_all': False, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_file': None})
ok: [rac-srv1] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'oracle_owne
atch_id': 26550339, u'patch_db_all': True, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_fil
```

```
TASK [orapatch : Patch DB] ********************************************************************************************************
ok: [rac-srvl] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'oracle_own
39, u'patch_db_all': False, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_file': None})
ok: [rac-srvl] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'oracle_own
atch_id': 26550339, u'patch_db_all': True, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_fi

TASK [orapatch : Patch OH OJVM] ***************************************************************************************************
skipping: [rac-srvl] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'orac
26550339, u'patch_db_all': False, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_file': None
skipping: [rac-srvl] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'orac
e, u'patch_id': 26550339, u'patch_db_all': True, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'ora

TASK [orapatch : Patch DB OJVM] ***************************************************************************************************
skipping: [rac-srvl] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'orac
26550339, u'patch_db_all': False, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_file': None
skipping: [rac-srvl] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'orac
e, u'patch_id': 26550339, u'patch_db_all': True, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'ora

TASK [orapatch : [SYSTEM] End logger session] *************************************************************************************
ok: [rac-srvl]

TASK [orapatch : [SYSTEM] Fetch orapatch logfile] *********************************************************************************
changed: [rac-srvl]
```

```
PLAY [Patch oracle software] ***************************************************************************************************

TASK [Gathering Facts] *********************************************************************************************************
ok: [rac-srv2]

TASK [assert] ******************************************************************************************************************
ok: [rac-srv2] => {
    "changed": false,
    "msg": "All assertions passed"
}

TASK [orapatch : [SYSTEM] Include vars] ****************************************************************************************
ok: [rac-srv2]

TASK [orapatch : [SYSTEM] Push sql scripts] ************************************************************************************
changed: [rac-srv2]

TASK [orapatch : [SYSTEM] Ensure 'orapatch' log file exists] *******************************************************************
ok: [rac-srv2]

TASK [orapatch : [SYSTEM] Start logger session] ********************************************************************************
ok: [rac-srv2]
```

```
TASK [orapatch : Check OPatch minimum version] *********************************************************************
ok: [rac-srv2] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'oracle_own
39, u'patch_db_all': False, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_file': None})
ok: [rac-srv2] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'oracle_own
atch_id': 26550339, u'patch_db_all': True, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_fi

TASK [orapatch : Check conflicts against OH] ***********************************************************************
ok: [rac-srv2] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'oracle_own
39, u'patch_db_all': False, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_file': None})
ok: [rac-srv2] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'oracle_own
atch_id': 26550339, u'patch_db_all': True, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_fi

TASK [orapatch : Patch OH] *****************************************************************************************
ok: [rac-srv2] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'oracle_own
39, u'patch_db_all': False, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_file': None})
ok: [rac-srv2] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'oracle_own
atch_id': 26550339, u'patch_db_all': True, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_fi
```

```
TASK [orapatch : Patch DB] ********************************************************************************************
ok: [rac-srv2] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'oracle_owr
39, u'patch_db_all': False, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_file': None})
ok: [rac-srv2] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'oracle_owr
atch_id': 26550339, u'patch_db_all': True, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_fi

TASK [orapatch : Patch OH OJVM] ***************************************************************************************
skipping: [rac-srv2] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'orac
26550339, u'patch_db_all': False, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_file': None
skipping: [rac-srv2] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'orac
e, u'patch_id': 26550339, u'patch_db_all': True, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'ora

TASK [orapatch : Patch DB OJVM] ***************************************************************************************
skipping: [rac-srv2] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'orac
26550339, u'patch_db_all': False, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'oratab_file': None
skipping: [rac-srv2] => (item={u'patch_db_list': [u'{ "dbname": "" }'], u'patch_ojvm': False, u'run_only_checks': False, u'orac
e, u'patch_id': 26550339, u'patch_db_all': True, u'patch_only_oh': False, u'patch_directory': u'12.1.0.2/psu/August2017', u'ora

TASK [orapatch : [SYSTEM] End logger session] *************************************************************************
ok: [rac-srv2]

TASK [orapatch : [SYSTEM] Fetch orapatch logfile] *********************************************************************
changed: [rac-srv2]
```

ansible blog posts

- AUTOMATINC ORACLE RMAN BACKUP CONFIGURATION ON LINUX WITH ANSIBLE

https://www.pythian.com/blog/automating-oracle-rman-backup-configuration-linux-ansible

- CREATING ANSIBLE CUSTOM MODULE FOR AWR REPORT GENERATION

https://www.pythian.com/blog/creating-ansible-custom-module-for-awr-reports-generation

- SIMPLE STEPS TO PERFORM OPATCH MAINTENANCE WITH ANSIBLE

https://www.pythian.com/blog/opatch-maintenance-with-ansible

- ANSIBLE AND AWS AUTOMATION

https://www.pythian.com/blog/ansible-and-aws-automation

More at: https://www.pythian.com/blog/?s=ansible

Thank you.

Questions?