



**CREATIVE Associates**  
MEASURE & CONTROL

# Fortra VitylCM

## Oracle Automatic Workload Agent

Author: Walter Verhoeven

Doc. Ref.: ora\_awld\_agent\_vitylcm\_public\_v1.2.doc

Version: 1.2

Date: 5 March 2026

## I. **Document History**

<b>Version</b>	<b>Date</b>	<b>Name</b>	<b>Description</b>
1.0	19/11/2025	W. Verhoeven	First release
1.1	25/11/2025	W. Verhoeven	Removed "identifier" from collector output
1.2	05/03/2026	W. Verhoeven	Updated for VityICM 2.23

## II. **Stakeholders**

<b>Company</b>	<b>Responsible Stakeholder</b>
CREATIVE Associates	Walter Verhoeven

### **CREATIVE Associates**

Donderveldstraat 52  
1651 Lot/Beersel  
Belgium  
[www.creative-associates.com](http://www.creative-associates.com)  
[info@creative-associates.com](mailto:info@creative-associates.com)

III. Contents

# Table of Contents

- 1. MANAGEMENT SUMMARY.....4**
- 1.1 Introduction.....4
- 1.2 Versions.....4
- 1.3 Limitations.....4
- 1.4 Requirements.....5
- 2. ORACLE AUTOMATIC WORKLOAD GENERATOR.....7**
- 2.1 Deployment On Linux Vityl Collector Side.....7
- 2.2 Create And Assign A Configuration Policy In VitylCM For This Node.....9
- 2.3 Install The Workload Definer On The Central VitylCM Server.....16
- 3. CHARTS PERFORMANCE MONITOR & AUTOMATED ANALYTICS.....18**
- 3.1 Where Are The Oracle Instances Running.....18
  - 3.1.1 Performance Monitor.....18
  - 3.1.2 Automated Analytics.....20
- 3.2 What Resources Are The Oracle Instances Consuming.....20
  - 3.2.1 Performance Monitor.....20
  - 3.2.2 Automated Analytics.....22

## 1. **Management Summary**

### 1.1 **Introduction**

There are companies that have a quite large deployment (100+) of Oracle instances spread over multiple RHEL based servers configured in a cluster environment. This can be a quite dynamic environment where Oracle instances are frequently created, removed and (or) moved.

If that company is using the FORTRA VityICM solution it will benefit from our module called Automatic Workload Agent. So if there is an interest please go to [www.creative-associates.com](http://www.creative-associates.com) and contact us via the Contact Page.

VityICM uses a light weight collector called “vityl-collector” that passes performance data to the VityICM Cassandra database. It offers a Workload definition capability of grouping Oracle instances or other application processes, in order to have Workload based performance data, but by design this is a manual configuration process.

In order to automate the configuration process we developed this solution that consists of the agent named “**ora\_awld.py**” that runs on the managed node supporting this Vityl Collector performance agent and the module named “**ora\_awld\_vitylcm.py**” that runs on the VityICM server defining the actual groupings of the Oracle instances related processes.

This document describes the usage, functionality and installation of this Oracle Automated Workload Agent for VityICM.

### 1.2 **Versions**

First release 1.0 was on 19/11/2025.

Second release 1.1 was on 25/11/2025, here we removed the “identifier” definition in the “ora\_awld.py” module. Now the Oracle SID’s are visible in the detailed set of metrics of a server. There is no more an extra Plugin server reference like “serverName-<GUID>” created by VityICM due to this “identifier”.

The “ora\_awld\_vitylcm.py” has been modified too in order to look for Linux nodes with the “ora\_awld” group of data instead of systems with a Plugin reference.

Adapted release version towards 1.2 on 5 March 2026 after successful tests on VityICM version 2.23 which is now also supported.

### 1.3 **Limitations**

This version 1.1 has been developed for VityICM 2.21.0 and Vityl Collector 2.21.0 and should be re-evaluated/tested with each new patch and major release version of the VityICM and Vityl Collector modules.

We adapted our code in order to support VityICM version 2.23 as changes on the API level were introduced by Fortra. We also support Vityl Collector 2.22.0.

We know that the next release of VityICM 2.24, planned for Q1 or Q2 2026, will introduce enhanced API security features which will require an adapted version of our Oracle Automatic Workload Agent.

As this custom agent has been developed by CREATIVE Associates the support services of Fortra will not handle questions nor calls related to it. These will rather be handled by the support services of CREATIVE Associates itself.

Since we are using API’s to interact with VityICM these two Python modules “**ora\_awld.py**” and

“ora\_awld\_vitylcm.py” may not be modified nor copied by the customer as it falls under the Copyright law(s) and could possibly break a correct functioning VitylCM environment.

Currently we only support the Linux environment/platform.

## 1.4 Requirements

Both Python programs need at least Python version 3.6.x installed on the node that runs Vityl Collector and the VitylCM server. You can check this with the following command “python3 -V”:

```
[vverhoeven@vmorasimul2 ora_awld]$ /usr/bin/python3 -V
Python 3.6.8
```

```
[vverhoeven@vitylcm ora_awld_vitylcm]$ /usr/bin/python3 -V
Python 3.12.1
```

Vityl Collector installed and operational on the managed Linux node.

You can check the status as following “systemctl status vityl-collector”:

```
[root@vmorasimul2 plugin]# systemctl status vityl-collector
● vityl-collector.service - Vityl Collector
  Loaded: loaded (/usr/lib/systemd/system/vityl-collector.service; enabled; vendor preset: disabled)
  Active: active (running) since Tue 2025-11-11 11:11:54 CET; 21s ago
  Main PID: 1458324 (vityl-collector)
  Tasks: 10 (limit: 12227)
  Memory: 9.0M
  CGroup: /system.slice/vityl-collector.service
          └─1458324 /usr/bin/vityl-collector -c /etc/vityl/collector
```

```
Nov 11 11:11:54 vmorasimul2 systemd[1]: Started Vityl Collector.
```

```
Nov 11 11:11:54 vmorasimul2 vityl-collector[1458324]: [2025-Nov-11 10:11:54] Info: vityl-collector version 2.21.0 running on vmorasimul2
```

```
Nov 11 11:11:54 vmorasimul2 vityl-collector[1458324]: [2025-Nov-11 10:11:54] Info: CentOS Stream 8
```

```
Nov 11 11:11:54 vmorasimul2 vityl-collector[1458324]: [2025-Nov-11 10:11:54] Info: x86_64 vmorasimul2 4.18.0-553.6.1.el8.x86_64 #1 SMP Thu May 30 04:13:58 UTC 2024
```

```
Nov 11 11:11:55 vmorasimul2 vityl-collector[1458324]: [2025-Nov-11 10:11:55] Info: Found kafka connection string: vitylcm:9092
```

```
Nov 11 11:11:57 vmorasimul2 vityl-collector[1458324]: [2025-Nov-11 10:11:57] Info: Using configuration: /vityl_master_v2/configs/unassigned
```

```
Nov 11 11:11:57 vmorasimul2 vityl-collector[1458324]: [2025-Nov-11 10:11:57] Info: Collector configuration is not assigned.
```

```
Nov 11 11:11:57 vmorasimul2 vityl-collector[1458324]: [2025-Nov-11 10:11:57] Info: The zookeeper or plugin configuration has changed. Reconfiguring collector.
```

VitylCM server installed and operational with the collector(s) in a normal state.

VCM								
Status Collectors Configurations Retentions Help								
Collectors								
Filter		<input type="checkbox"/>	Collector ↑↓	Status ↑↓	Configuration ↑↓	Version ↑↓	Last Updated ↑↓	Started On ↑↓
<input type="text"/>		<input type="checkbox"/>	vmorasimul2	Normal	unassigned	2.21.0	2025-11-11 11:11	2025-11-11 11:11
Status		<input type="checkbox"/>	vmorasimul	Normal	vmorasimul	2.21.0	2025-11-11 11:12	2025-11-05 09:31
<input checked="" type="checkbox"/> Normal		<input type="checkbox"/>	teamquest1	Normal	teamquest1	2.21.0	2025-11-11 11:08	2025-11-08 17:58

It is mandatory that each server, running the “ora\_awld” Plugin, has a custom configuration policy with the **same name** as the collector/system name. So this is a manually step that must be completed for each node where you install “ora\_awld”.

This is explained in section 2.2 of this document.

## 2. Oracle Automatic Workload Generator

### 2.1 Deployment On Linux Vityl Collector Side

As mentioned above we have a Python script named “**ora\_awld.py**” which is deployed on the RHEL server running the Oracle database(s). It is mandatory to install Vityl Collector prior to the deployment of this Python script as it will pass the detected Oracle instance names towards the Cassandra database.

After the installation of the Vityl Collector you will have a directory “/etc/vityl/collector” that contains the following:

```
[wverhoeven@vmorasimul2 collector]$ ls -l
total 8
-rw-----. 1 vityl-cm vityl-cm 42 Aug 18 13:27 guid.properties
drwxr-xr-x. 4 root    root    65 Aug 18 13:26 plugin
-rw-----. 1 vityl-cm vityl-cm 39 Aug 18 13:26 zookeeper.properties
```

The “plugin” directory contains the following:

```
[root@vmorasimul2 plugin]# ls -l
total 4
drwxr-xr-x. 2 root    root    6 Oct 25 2024 definitions.d
-rw-r--r--. 1 vityl-cm vityl-cm 4 Oct 25 2024 identifiers.json
drwxr-xr-x. 2 root    root    94 Aug 18 13:26 schema
```

Step1) Stop the Vityl Collector.

```
[root@vmorasimul2 plugin]# systemctl stop vityl-collector
```

Check the status afterwards with the following command.

```
[root@vmorasimul2 plugin]# systemctl status vityl-collector
● vityl-collector.service - Vityl Collector
   Loaded: loaded (/usr/lib/systemd/system/vityl-collector.service; enabled; vendor preset: disabled)
   Active: inactive (dead) since Thu 2025-11-13 13:23:27 CET; 9s ago
     Process: 1458324 ExecStart=/usr/bin/vityl-collector -c /etc/vityl/collector (code=exited, status=0/SUCCESS)
    Main PID: 1458324 (code=exited, status=0/SUCCESS)

Nov 13 11:48:46 vmorasimul2 vityl-collector[1458324]: [2025-Nov-13 10:48:46] Info: Collector configuration is not assigned.
Nov 13 11:48:46 vmorasimul2 vityl-collector[1458324]: [2025-Nov-13 10:48:46] Info: The zookeeper or plugin configuration has
changed. Reconfiguring collector.
Nov 13 11:48:59 vmorasimul2 vityl-collector[1458324]: [2025-Nov-13 10:48:59] Info: Kafka Event (Local: Broker transport
failure): vitylcm:9092/bootstrap: Connect to ipv4#15.160.6.204:9092 failed: Connection refused (after 0ms in state CONNECT)
Nov 13 11:48:59 vmorasimul2 vityl-collector[1458324]: [2025-Nov-13 10:48:59] Info: Kafka Event (Local: All broker connections
are down): 1/1 brokers are down
Nov 13 11:49:59 vmorasimul2 vityl-collector[1458324]: [2025-Nov-13 10:49:59] Info: The Kafka connection is up.
Nov 13 13:21:55 vmorasimul2 vityl-collector[1458324]: [2025-Nov-13 12:21:55] Info: The zookeeper or plugin configuration has
changed. Reconfiguring collector.
Nov 13 13:23:27 vmorasimul2 systemd[1]: Stopping Vityl Collector...
Nov 13 13:23:27 vmorasimul2 vityl-collector[1458324]: [2025-Nov-13 12:23:27] Info: Termination has been requested with
signal 15
Nov 13 13:23:27 vmorasimul2 systemd[1]: vityl-collector.service: Succeeded.
Nov 13 13:23:27 vmorasimul2 systemd[1]: Stopped Vityl Collector.
```

Step2) As root user copy the file “ora\_awld.zip” into the “/etc/vityl/collector/plugin” directory.

```
[root@vmorasimul2 plugin]# pwd
/etc/vityl/collector/plugin
[root@vmorasimul2 plugin]# ls -l
total 8
```

```
drwxr-xr-x. 2 root root 6 Oct 25 2024 definitions.d
-rw-r--r--. 1 vityl-cm vityl-cm 4 Oct 25 2024 identifiers.json
-rw-r--r--. 1 root root 1398 Nov 13 13:15 ora_awld.zip
drwxr-xr-x. 2 root root 94 Aug 18 13:26 schema
[root@vmorasimul2 plugin]# ls -l definitions.d/
total 0
```

Step3) Unzip that file in this directory.

```
[root@vmorasimul2 plugin]# unzip ora_awld.zip
Archive: ora_awld.zip
  inflating: ora_awld.py
  inflating: definitions.d/ora_awld.json
  extracting: ora_awld.log
[root@vmorasimul2 plugin]# ls -l
total 12
drwxr-xr-x. 2 root root 27 Nov 13 13:21 definitions.d
-rw-r--r--. 1 vityl-cm vityl-cm 4 Oct 25 2024 identifiers.json
-rw-rw-rw-. 1 root root 0 Nov 13 12:19 ora_awld.log
-rwxr-xr-x. 1 root root 2608 Nov 13 12:19 ora_awld.py
-rw-r--r--. 1 root root 1398 Nov 13 13:15 ora_awld.zip
drwxr-xr-x. 2 root root 94 Aug 18 13:26 schema
[root@vmorasimul2 plugin]# ls -l ./definitions.d/
total 4
-rw-r--r--. 1 root root 86 Nov 13 11:35 ora_awld.json
```

It will extract the file “ora\_awld.py”, which is the Python script that will pass a string to STDOUT in a JSON format so that the Vityl Collector can pass it to the Cassandra database. An output example is shown below:

```
{ "groupName": "ora_awld", "metrics": [{"value": 1, "metricName": "RBK01"},
{"value": 1, "metricName": "LPSI"}, {"value": 1, "metricName": "OILM"}, {"value": 1, "metricName": "DSL1"},
{"value": 1, "metricName": "EFP01R"}]}
```

The metric group is named “oraawl” and several Oracle instance name “RBK01”, “LPSI”, “OILM”, “DSL1” and “EFP01R” have been detected.

It also extracts the file “./definitions.d/ora\_awld.json” that contains a reference to the Python script:

```
[root@vmorasimul2 definitions.d]# pwd
/etc/vityl/collector/plugin/definitions.d
[root@vmorasimul2 definitions.d]# ls -l
total 4
-rw-r--r--. 1 root root 86 Nov 13 11:35 ora_awld.json
[root@vmorasimul2 definitions.d]# cat ora_awld.json
{
  "name": "ora_awld",
  "pluginLocation": "/etc/vityl/collector/plugin/ora_awld.py"
}
```

You can test the Python script by executing it manually, it should produce a JSON string that contains the detected Oracle instances.

```
[root@vmorasimul2 plugin]# ./ora_awld.py
{"groupName": "ora_awld", "identifier": "vmorasimul2", "metrics": [{"value": 1, "metricName": "OAS1"}]}
```

Step4) Start the Vityl Collector again and check if it runs fine.

```
[root@vmorasimul2 plugin]# systemctl start vityl-collector
[root@vmorasimul2 plugin]# systemctl status vityl-collector
```

- vityl-collector.service - Vityl Collector

Loaded: loaded (/usr/lib/systemd/system/vityl-collector.service; enabled; vendor preset: disabled)

Active: active (running) since Thu 2025-11-13 13:54:13 CET; 12s ago

Main PID: 1484804 (vityl-collector)

Tasks: 10 (limit: 12227)

Memory: 5.4M

CGroup: /system.slice/vityl-collector.service

└─1484804 /usr/bin/vityl-collector -c /etc/vityl/collector

Nov 13 13:54:13 vmorasimul2 systemd[1]: Started Vityl Collector.

Nov 13 13:54:13 vmorasimul2 vityl-collector[1484804]: [2025-Nov-13 12:54:13] Info: vityl-collector version 2.21.0 running on vmorasimul2

Nov 13 13:54:13 vmorasimul2 vityl-collector[1484804]: [2025-Nov-13 12:54:13] Info: CentOS Stream 8

Nov 13 13:54:13 vmorasimul2 vityl-collector[1484804]: [2025-Nov-13 12:54:13] Info: x86\_64 vmorasimul2 4.18.0-553.6.1.el8.x86\_64 #1 SMP Thu May 30 04:13:58 UTC 2024

Nov 13 13:54:13 vmorasimul2 vityl-collector[1484804]: [2025-Nov-13 12:54:13] Info: Found kafka connection string: vitylcm:9092

Nov 13 13:54:13 vmorasimul2 vityl-collector[1484804]: [2025-Nov-13 12:54:13] Info: Using configuration: /vityl\_master\_v2/configs/unassigned

Nov 13 13:54:13 vmorasimul2 vityl-collector[1484804]: [2025-Nov-13 12:54:13] Info: Collector configuration is not assigned.

Nov 13 13:54:13 vmorasimul2 vityl-collector[1484804]: [2025-Nov-13 12:54:13] Info: The zookeeper or plugin configuration has changed. Reconfiguring collector.

## 2.2 Create And Assign A Configuration Policy In VitylCM For This Node

When the Fortra Vityl Collector has been installed on a system it will start collecting data based on a configuration policy that must be assigned in VitylCM. By default no policy is assigned. You can see this in the output of the command “systemctl status vityl-collector”:

...

Nov 13 13:54:13 vmorasimul2 vityl-collector[1484804]: [2025-Nov-13 12:54:13] Info: Found kafka connection string: vitylcm:9092

Nov 13 13:54:13 vmorasimul2 vityl-collector[1484804]: [2025-Nov-13 12:54:13] Info: Using configuration: /vityl\_master\_v2/configs/unassigned

Nov 13 13:54:13 vmorasimul2 vityl-collector[1484804]: [2025-Nov-13 12:54:13] Info: **Collector configuration is not assigned.**

Nov 13 13:54:13 vmorasimul2 vityl-collector[1484804]: [2025-Nov-13 12:54:13] Info: The zookeeper or plugin configuration has changed. Reconfiguring collector.

...

Attention: It is mandatory that each server, running the “ora\_awld” Plugin, has a custom configuration policy with the **same name** as the collector/system name. So for example system/collector “vmorasimul2” should have a configuration policy/definition with the name “vmorasimul2”.

Filter	Collector ↑↓	Status ↑↓	Configuration ↑↓	Version ↑↓	Last Updated ↑↓	Started On ↑↓
<input type="checkbox"/>	fortra1	✔ Normal	teamquest1	2.21.0	2025-11-13 11:44	2025-10-29 11:04
<input type="checkbox"/>	vmorasimul2	✔ Normal	unassigned	2.21.0	2025-11-13 14:30	2025-11-13 13:54
<input type="checkbox"/>	vmorasimul	✔ Normal	vmorasimul	2.21.0	2025-11-13 14:32	2025-11-13 11:37
<input type="checkbox"/>	teamquest1	✔ Normal	teamquest1	2.21.0	2025-11-13 14:33	2025-11-08 17:58
<input type="checkbox"/>	wavbell1	✔ Normal	wavbell1	2.5.0	2025-11-13 14:32	2007-01-01 01:00
<input type="checkbox"/>	crunchy	✔ Normal	default	2.21.0	2025-11-13 14:33	2025-11-06 09:59

In this case we see an “unassigned” configuration policy so we will create one via the following steps.

Step1) Go to the Configurations screen and click New Configuration.

Name ↑	Last Modified ↑↓
default	2025-10-28 08:44
example	2025-10-29 07:23
oracleWorkload	2025-08-06 14:17
teamquest1	2025-08-04 12:28

In our case we give the name “vmorasimul2” and activate the Plugin at the collection interval of 1 Minute.

**VCM** Status Collectors Configurations Retentions Help

## New Configuration Save Cancel

Name: **vmorasimul2** Logging Level: Info Heartbeat Interval Seconds: 300

Data Buffer: Disabled Data Streaming: Continuous

Standard Metrics

Metric	Interval
CPU	1 Minute
Disk Inventory	1 Hour
Disk Space	15 Minutes
Disk IO	1 Minute
File System Inventory	1 Hour
Kernel Queue	1 Minute
Kernel Details	1 Minute
Memory	1 Minute
Network IO	1 Minute
NFS	No Collection
NFS IO	No Collection
Plugin	1 Minute
System Properties	1 Hour

Step2) Enable Process Data Advanced Collection and specify a Workloads Advanced collection type.

The screenshot shows the Oracle Automatic Workload Agent configuration interface. An "Advanced Collection" dialog box is open, allowing configuration of collection settings. The dialog includes fields for "Full Command", "Collected" (set to "Collected"), and "Workloads" (set to "Advanced"). It also features a table for "Named Workloads" with columns for "Workload Name" and "Condition", and a section for "Reduce Process Data Records by" with checkboxes for "CPU (sec) Threshold" and "RSS (MB) Threshold", both set to 0. "OK" and "Cancel" buttons are at the bottom of the dialog.

The background configuration table includes sections for Standard Resource Metrics, Detailed Metrics, and Container Metrics. The "Process Data" row in the Detailed Metrics section is highlighted with a red border:

Metric	Interval	Top N Metric	Top N value	Advanced Collection
Process Data	1 Minute	CPU (sec)	20	Enabled

You can manually specify Workload Names and a corresponding conditions like for example one for "VityCollector" and the "tnslsnr" activity. The Oracle Workload Agent will not modify these entries, it will only add and delete Oracle related ones which are prefixed with "oraawld\_".

The "Add Named Workload" dialog box is shown. It contains the following fields:

- Workload Name:** VityCollector
- Condition:** Command contains "vityl-collector"
- Command:** Contains

"OK" and "Cancel" buttons are located at the bottom right of the dialog.

### Advanced Collection ✕

Full Command

Collected ▼

---

Workloads

Advanced ▼

---

Named Workloads [Add Workload](#)

	Workload Name	Condition	
=	VitylCollector	Command contains "vityl-collector"	✎ ✕
=	tnslsnr	Full Command contains "tnslsnr"	✎ ✕

Reduce Process Data Records by

CPU (sec) Threshold:

RSS (MB) Threshold:

OK
Cancel

Select OK and finally save the new Configuration Policy.

△ VCM 
Status Collectors Configurations Retentions Help

## Configurations

New Configuration
Kafka URI

🔍 Filter configurations...

Name <span style="font-size: 12px;">↑↓</span>	Last Modified <span style="font-size: 12px;">↑↓</span>
default	2025-10-28 08:44
example	2025-10-29 07:23
oracleWorkload	2025-08-06 14:17
teamquest1	2025-08-04 12:28
vitylcm	2025-03-18 13:30
vmoracle	2025-03-25 10:06
vmorasimul	2025-08-14 13:18
vmorasimul2	2025-11-13 15:04
wavbell1	2025-03-18 13:31

Step3) Assign this Configuration Policy to the “vmorasimul2” node by selecting it in the Collectors screen.

The screenshot shows the VCM interface with the 'Collectors' tab selected. A table lists several collectors. The 'vmorasimul2' collector is highlighted with a red box. The table has columns for Collector, Status, Configuration, Version, Last Updated, and Started On.

Filter	Collector ↑↓	Status ↑↓	Configuration ↑↓	Version ↑↓	Last Updated ↑↓	Started On ↑↓
<input type="checkbox"/> teamquest1	teamquest1	Normal	teamquest1	2.21.0	2025-11-13 15:03	2025-11-08 17:58
<input type="checkbox"/> vmorasimul2	vmorasimul2	Normal	unassigned	2.21.0	2025-11-13 15:05	2025-11-13 13:54
<input type="checkbox"/> fortra1	fortra1	Normal	teamquest1	2.21.0	2025-11-13 11:44	2025-10-29 11:04

The screenshot shows the 'Collector Details' page for 'vmorasimul2'. The configuration dropdown is set to 'vmorasimul2' and the 'Apply' button is highlighted with a red box. The configuration details include Name, Logging Level, Heartbeat Interval, Data Buffer, Data Streaming, and Last Modified.

Host Name vmorasimul2	Operating System CentOS Stream 8	Status Normal
Collector Version 2.21.0	Last Updated 2025-11-13 15:05	Started On 2025-11-13 13:54

Configuration

Configuration: vmorasimul2

Name vmorasimul2	Logging Level Info	Heartbeat Interval 300
Data Buffer Disabled	Data Streaming Continuous	Last Modified 2025-11-13 15:04

Standard Metrics

The screenshot shows the VCM interface with the 'Collectors' tab selected. A table lists several collectors. The 'vmorasimul2' collector is highlighted with a red box. The table has columns for Collector, Status, Configuration, Version, Last Updated, and Started On.

Filter	Collector ↑↓	Status ↑↓	Configuration ↑↓	Version ↑↓	Last Updated ↑↓	Started On ↑↓
<input type="checkbox"/> fortra1	fortra1	Normal	teamquest1	2.21.0	2025-11-13 11:44	2025-10-29 11:04
<input type="checkbox"/> vmorasimul	vmorasimul	Normal	vmorasimul	2.21.0	2025-11-13 15:07	2025-11-13 11:37
<input type="checkbox"/> teamquest1	teamquest1	Normal	teamquest1	2.21.0	2025-11-13 15:08	2025-11-08 17:58
<input type="checkbox"/> vmorasimul2	vmorasimul2	Normal	vmorasimul2	2.21.0	2025-11-13 15:09	2025-11-13 13:54

Once the Configuration Policy has been assigned the Plugin will be executed and the detected Oracle instance names will be passed towards the Cassandra database.

The screenshot shows the VCM interface with the 'Collectors' tab selected. The 'Collector Details' section displays the following information:

- Host Name: vmorasimul2
- Operating System: CentOS Stream 8
- Status: Normal (with a green checkmark icon)
- Collector Version: 2.21.0
- Last Updated: 2025-11-13 15:14
- Started On: 2025-11-13 13:54

Below this information is a 'Configuration' section and a 'Plugins' section. The 'Plugins' section contains a search bar labeled 'Filter plugins...' and a table with the following data:

Name ↑	Status ↓
ora_awld	Running

Step4) Check the status of the detected Oracle instances in the Performance Monitor.

Select the Details tab in order to see the active Oracle instances running on the node.

The screenshot shows the VCM Performance Monitor interface. On the left, there is a calendar for November 2025 with the 27th selected. Below the calendar, the 'Start Time' is set to 12:00 AM and the 'Interval' is set to 12:00 AM. The main area displays a chart titled 'ora awld ILINK02I'. The chart shows a single data point for 'Wednesday, Nov 26, 03:01' with a value of 1.00. The y-axis ranges from 0 to 1.2, and the x-axis shows dates from 26. Nov to 27. Nov. On the right side, there is a 'Metrics' panel with a search bar and a list of metrics:

- ora
- ora awld DSLI
- ora awld EFP01R
- ora awld ILINK02I
- ora awld Interval
- ora awld LPSI
- ora awld OASI
- ora awld OILM
- ora awld RBK01I

Remark that once a database instance name has been detected it will remain selectable as it is never removed from the Cassandra database. But the chart will only show a database instance, for the selected time period, if data can be found.

### 2.3 Install The Workload Definer On The Central VitylCM Server

Attention, in case of a three system deployment type the node of interest is **system1** responsible for the Front-End Components and Capacity Plans features.

Step1) As root user copy the file “ora\_awld\_vitylcm.zip” to the “/opt/vityl/util” directory and unzip it.

```
[root@vitylcm util]# unzip ora_awld_vitylcm.zip
Archive:  ora_awld_vitylcm.zip
  inflating: ora_awld_vitylcm.py
  extracting: ora_awld_vitylcm.log
```

Check the version Python, should be minimum version 3.6.x.

```
[root@vitylcm util]# python3 -V
Python 3.12.1
```

The following Python modules “pytz”, “requests”, “mysql-connector-python” and “pyodbc” are required in order to run “ora\_awld\_vitylcm.py”. Install these via the “pip3 install <module>” command.

There are three parameters you may modify in the “ora\_awld\_vitylcm.py” program:

- **DEBUG**; Can be False or True. Default is False. In normal operation this should be set to False. In case of issues set this to True. It will generate debugging information that is logged in a log file named “ora\_awld\_vitylcm.log” in the same directory as where the Python program resides.
- **TRACE**; Can be False or True. Default is False. In normal operation this should be set to False. In case of issues and we need extra detailed information, set this to True.
- **AGINGTIME**; Set in seconds, by default 43200 or 12h. It defines the time it retains the Oracle SID Workload definition in the Workload policy when the SID is no more detected by the “ora\_awld” Plugin running on the Oracle server(s).

```
# Parameters that can be modified by the user
#DEBUG=False
DEBUG=True
TRACE=False
#TRACE=True
AGINGTIME="43200" # In seconds so 12h (43200), max is 24h (86400). Time the Oracle SID remains in the Workload
definition when no more detected by the Plugin
```

Step2) In order to check the correct operation, set the **DEBUG** parameter to True and do a manual run of the program. It will create/append the logfile “ora\_awld\_vitylcm.log”. When the size of the logfile exceeds 10MBytes it will be overwritten during the next run.

In case you get the following error when running the code:

```
ERROR: Connection to the VitylCM database failed !
Connection error: ('01000', "[01000] [unixODBC][Driver Manager]Can't open lib 'usr/lib64/psqlodbcw.so' : file not found (0)
(SQLDriverConnect)")
```

You are missing the Linux package “postgresql-odbc”, so install this with “yum”.

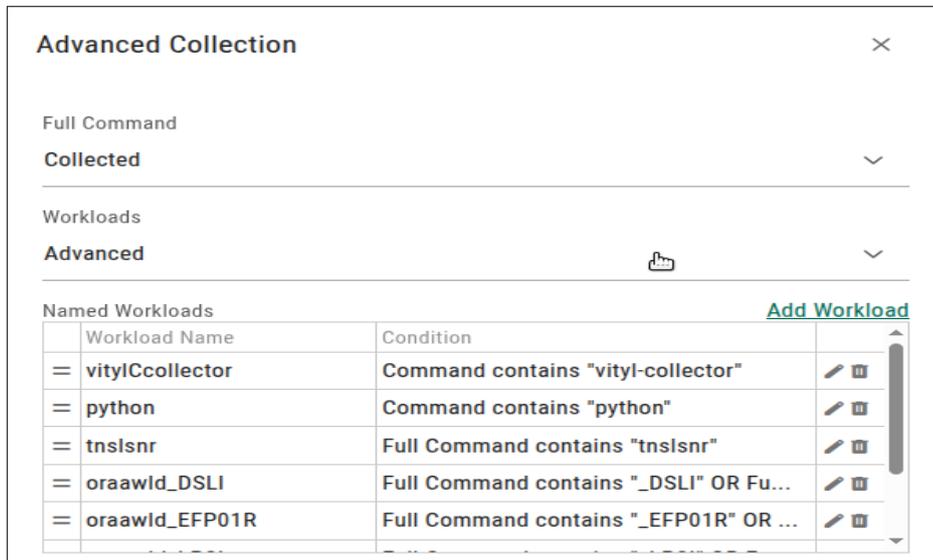
Step3) In case it runs correctly you may set the **DEBUG** parameter to False again and create a cron entry that looks as following “\*/15 \* \* \* \* /opt/vityl/util/ora\_awld\_vitylcm.py”, it will verify every 15 minutes if changes occurred on the Oracle server(s).

It is possible that a Warning pops-up due to HTTPS certification settings but it will not prevent the Workload Definer to operate correctly.

/usr/local/lib/python3.12/site-packages/urllib3/connectionpool.py:1097: InsecureRequestWarning: Unverified HTTPS request is being made to host '15.160.6.204'. Adding certificate verification is strongly advised. See: <https://urllib3.readthedocs.io/en/latest/advanced-usage.html#tls-warnings...>

After a successful run and when Oracle SID's where found on the nodes in the cluster you will find adapted Workload definitions in the configuration policy, below we show the configuration of the Oracle cluster server "vmorasimul".

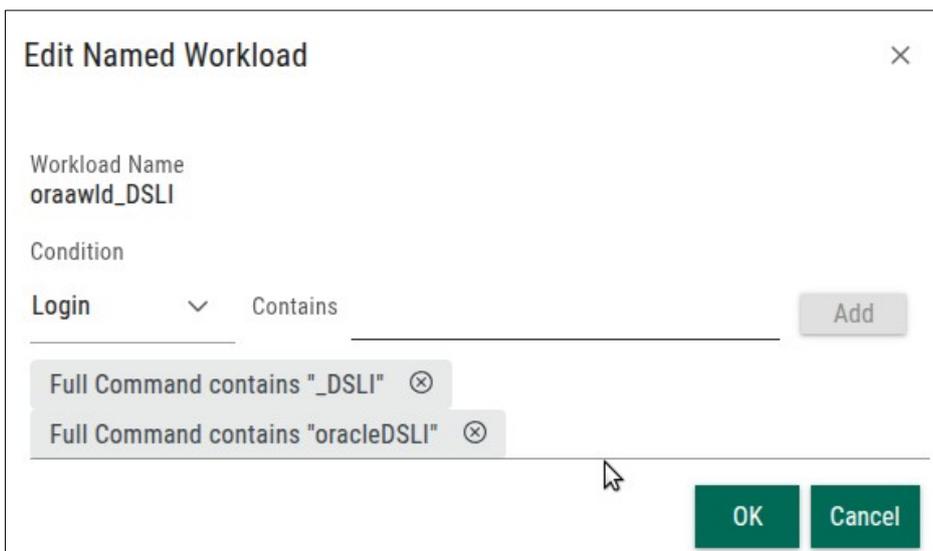
Remark that there are Workloads added with a prefix "oraawld\_" combined with the Oracle Instance name. So when changes occur these will be removed and others added. The Workloads that where added manually will not be modified.



The screenshot shows a dialog box titled "Advanced Collection" with a close button (X) in the top right corner. Below the title, there are sections for "Full Command", "Collected", "Workloads", and "Advanced". The "Advanced" section is expanded, showing a table of "Named Workloads". The table has columns for "Workload Name", "Condition", and a set of edit/delete icons. The "Add Workload" link is visible in the top right of the table area.

Workload Name	Condition	
vityIcollector	Command contains "vityl-collector"	 
python	Command contains "python"	 
tnslsnr	Full Command contains "tnslsnr"	 
oraawld_DSLI	Full Command contains "_DSL I" OR Fu...	 
oraawld_EFP01R	Full Command contains "_EFP01R" OR ...	 

The filter consists of the following two entries.



The screenshot shows a dialog box titled "Edit Named Workload" with a close button (X) in the top right corner. The "Workload Name" is "oraawld\_DSLI". The "Condition" is "Login" with a dropdown arrow. Below the condition, there is a list of conditions: "Full Command contains "\_DSL I" (with a delete icon) and "Full Command contains "oracleDSL I" (with a delete icon). There is an "Add" button to the right of the condition list. At the bottom right, there are "OK" and "Cancel" buttons.

### 3. Charts Performance Monitor & Automated Analytics

There are two types of data made available by the Oracle Automatic Workload Agent. Here we will describe how to chart these in both the Monitoring and the Analytics features of VityICM.

#### 3.1 Where Are The Oracle Instances Running

##### 3.1.1 Performance Monitor

As mentioned before on each Oracle server the “ora\_awld” Plugin is installed and executed every minute checking the availability of running Oracle instances. When an instance is detected the SID will be send to the VityICM server. As such you can easily see on what server the Oracle instances where running.

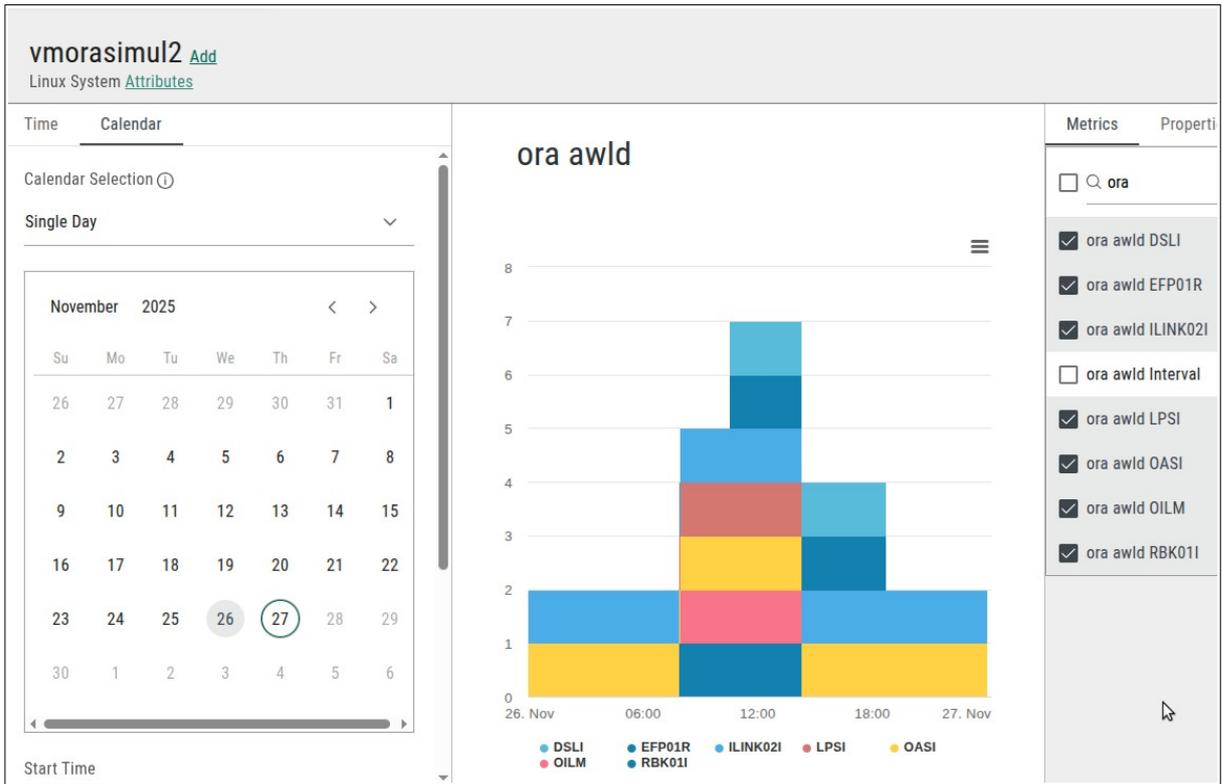
The example below shows a failover scenario where the five Oracle instances from “vmorasimul” are transferred to “vmorasimul2” and back.



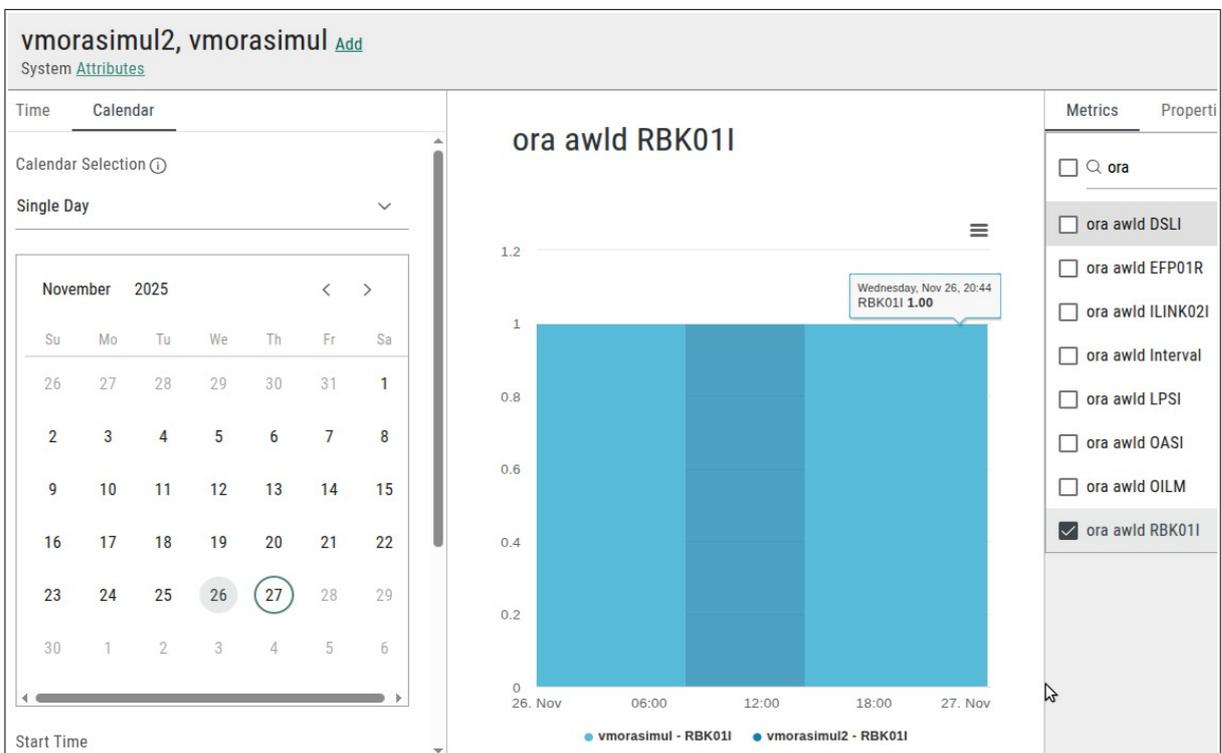
SID's “RBK01I”, “LPSI” and “OILM” are moved out at 8h. The remaining two “DSLI” and “EFP01R” around 10h29. No instances where running between 10h29 and 14h20 due to maintenance activities on the server for example.

Then at 14h20 the SID's “RBK01I”, “LPSI” and “OILM” are moved back and later at 18h44 “DSLI” and “EFP01R”.

When looking at server “vmorasimul2” we have the following.



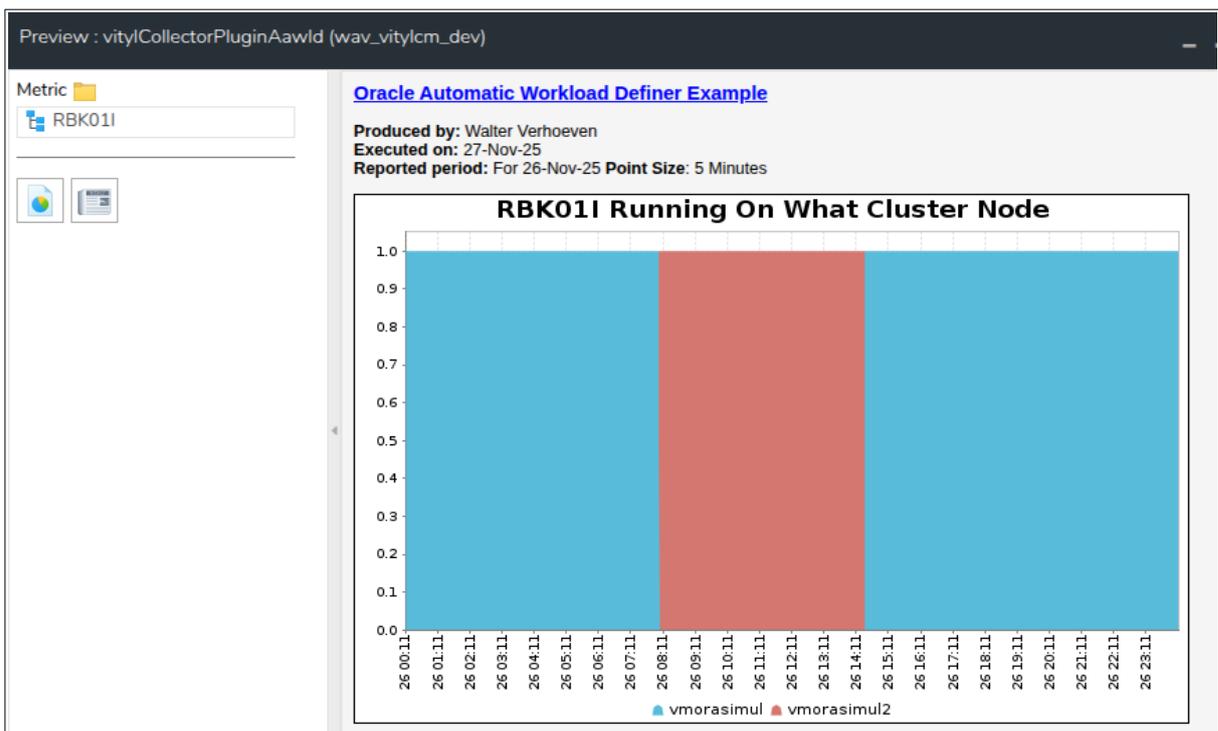
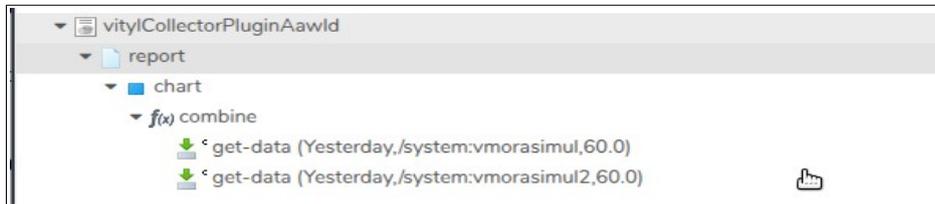
Two SID's were already running being “ILINK02I” and “OASI”. You can see that it received the five SID's from “vmorasimul” for a while and that these were transferred back. When stacking the systems together and selecting one SID you can see where it resided in the cluster.



### 3.1.2 Automated Analytics

The Automated Analytics engine allows you to create custom reports for automated analysis and reporting which can be scheduled and emailed or published in a portal called Report Navigator.

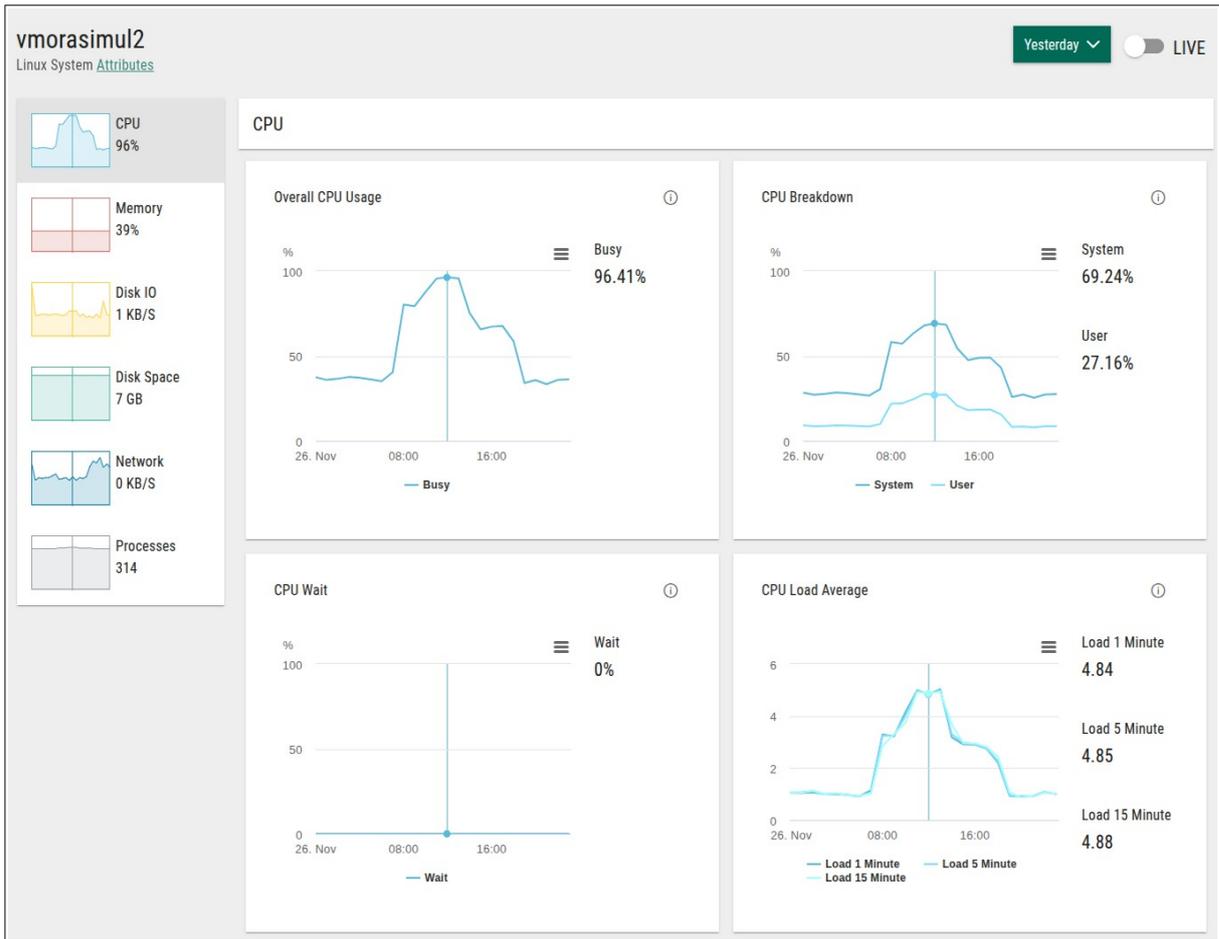
This report example allows you to select an Oracle SID and get an overview of on what server it ran at what time.



## 3.2 What Resources Are The Oracle Instances Consuming

### 3.2.1 Performance Monitor

Since we use the Workload statistics capability of Vityl Collector it is possible to drill down towards Oracle instance processes resources load and the corresponding Workload categorization.

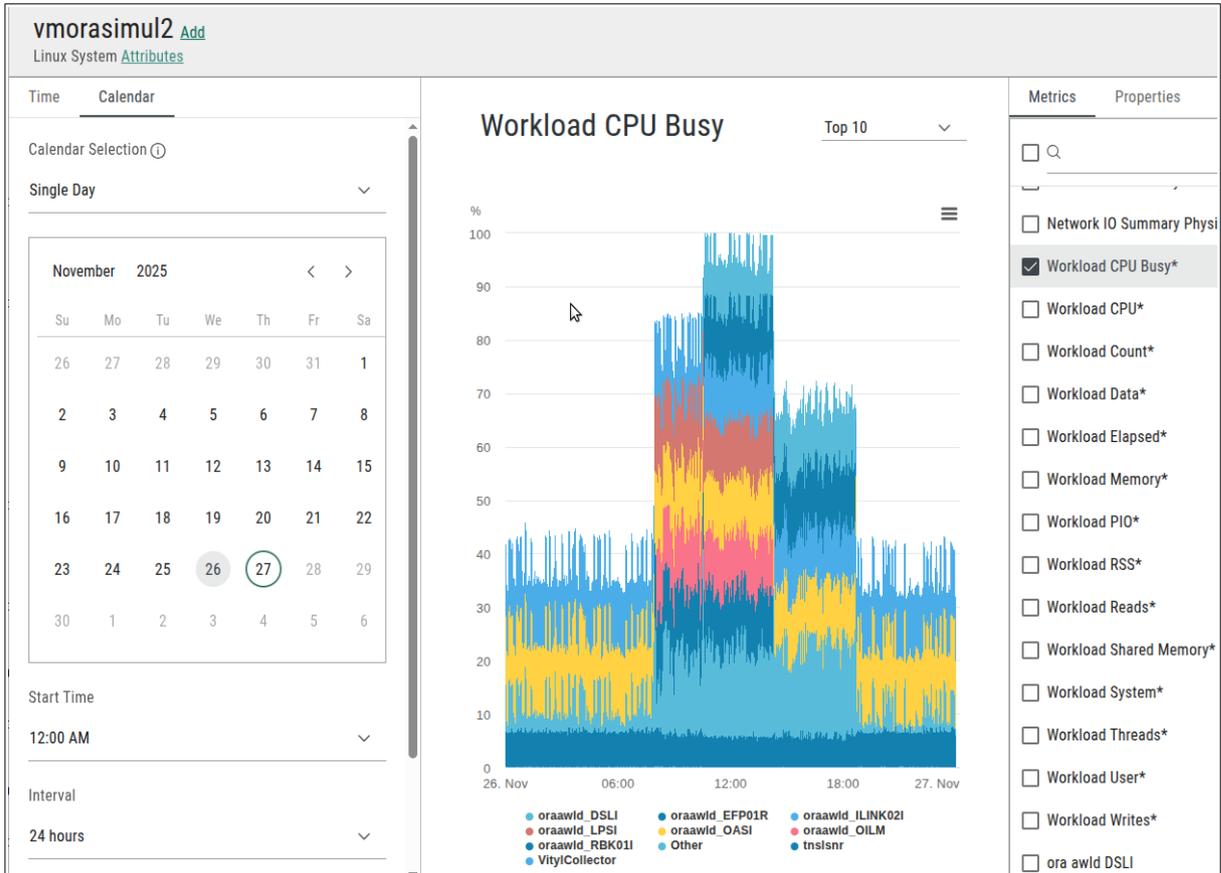


Select the server and the overview type of metrics, here you can select a point of interest and drill down towards process level to see what process is responsible for CPU, I/O or Memory consumption for example.

Filter Processes							
Login ↑↓	Command ↑↓	Workload ↑↓	Count ↑↓	ID ↑↓	PID ↑↓	CPU (%) ↑↓	CPU
<Completed>	<Completed>	<Other>	1	<Multi>	<Multi>	16.016	
oracle	ora_igwr_LPSI	oraawld_LPSI	1	1001	1845858	5.545	
oracle	oracleEFP01R	oraawld_EFP01R	1	1001	1848956	5.531	
oracle	oracleRBK01I	oraawld_RBK01I	1	1001	1845827	5.516	
oracle	oracleDSL1	oraawld_DSL1	1	1001	1848944	5.498	
oracle	ora_igwr_OILM	oraawld_OILM	1	1001	1845872	5.466	
oracle	ora_igwr_OASI	oraawld_OASI	1	1001	1484779	5.388	
oracle	ora_igwr_DSL1	oraawld_DSL1	1	1001	1848935	5.38	
oracle	ora_igwr_RBK01I	oraawld_RBK01I	1	1001	1845857	5.327	
oracle	oracleLPSI	oraawld_LPSI	1	1001	1845859	5.319	

At the same time there are specific Workload performance metrics that will become available too.

This is the CPU load by Oracle instance for yesterday for example. Other Workload metrics are also calculated and available for charting.

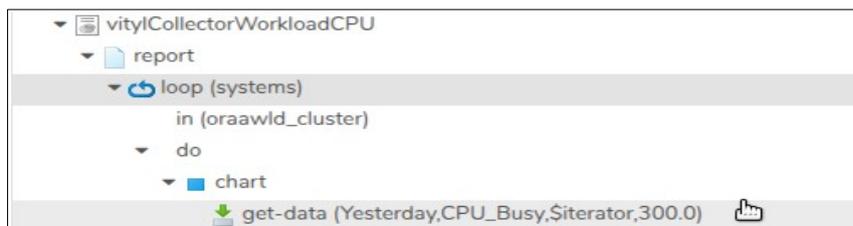


Remark that at the moment this server runs all seven Oracle Instances the CPU load reaches 100%.

### 3.2.2 Automated Analytics

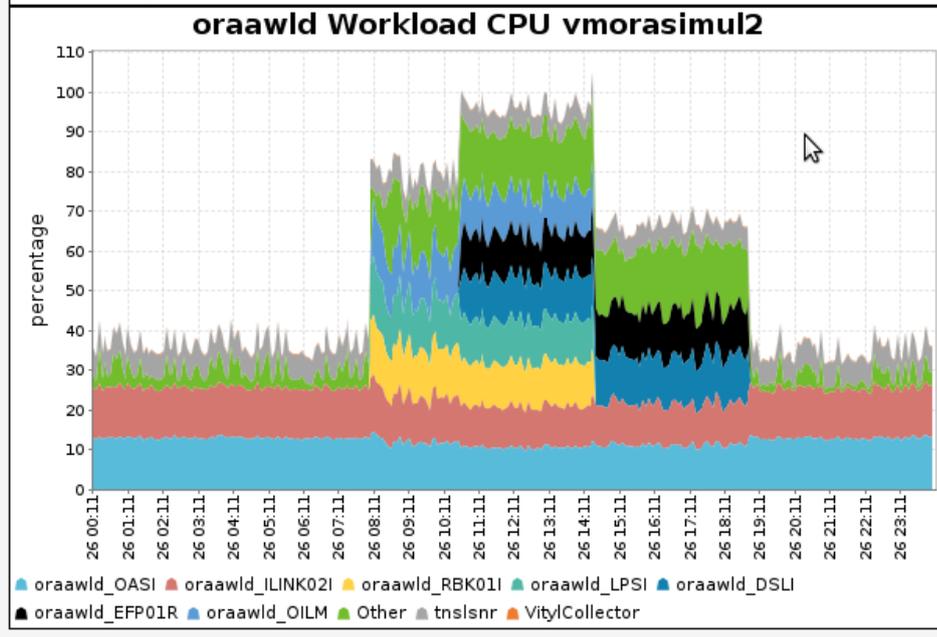
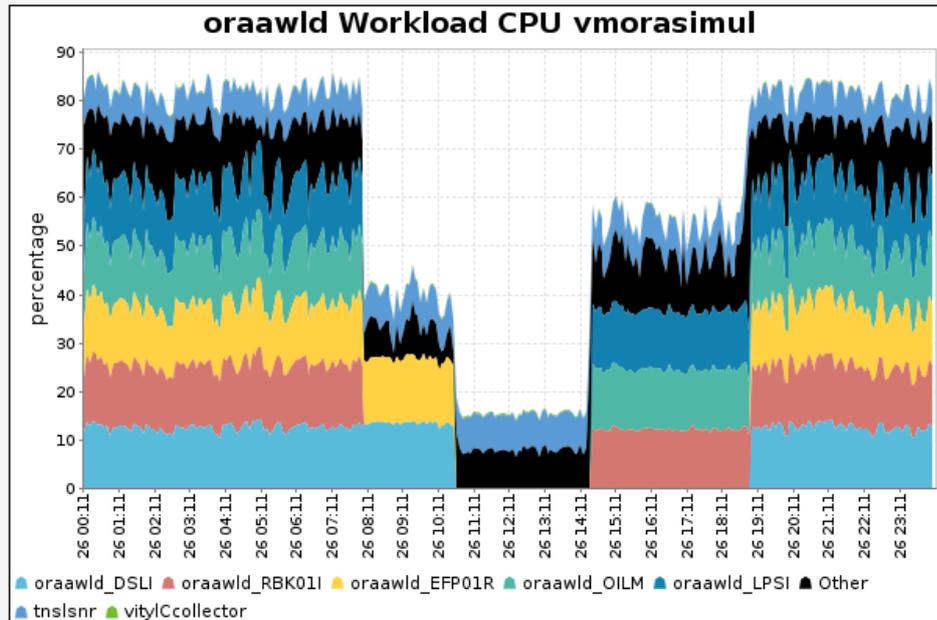
The Automated Analytics engine allows you to create custom reports for automated analysis and reporting which can be scheduled and emailed or published in a portal called Report Navigator.

An example custom view to generate the CPU consumption by Oracle instances per system in the cluster looks like this.



**Oracle Automatic Workload Definer Example**

**Produced by:** Walter Verhoeven  
**Executed on:** 27-Nov-25  
**Reported period:** For 26-Nov-25 Point Size: 5 Minutes



-END-