

This doc contains steps to install below appdynamics agents :

1) Servers

2) Clients

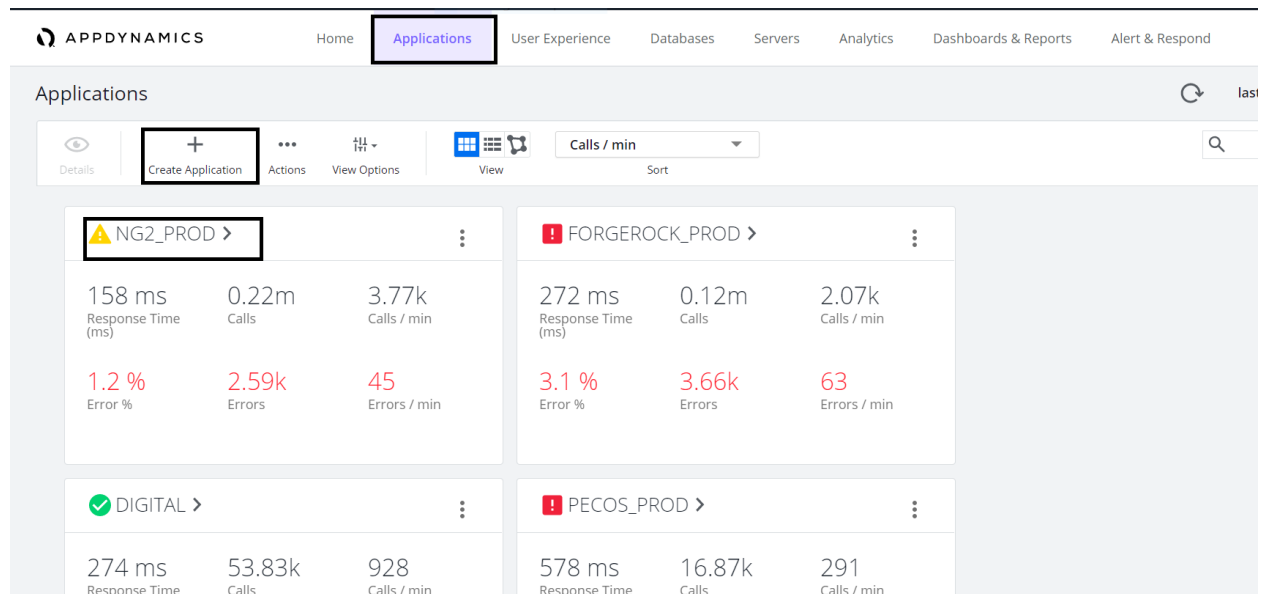
3) Databases

4) Analytics Agent

Servers:

To Install Appserver agent/ Machine agent / Network agent:

- 1) Before installing appd make sure to provide “**appd_application_name**” [for server applications only [Appserver agent] in the corresponding playbook to push metrics to respective Application in Appdynamics
- 2) If it needs to create under separate application then create new application as shown in the screenshot and then add the application name in the playbook



3) Run any playbook below to install agents

- i) Appdynamics playbook
- ii) Each playbook has appd.yml

Ref:

```
/usr/bin/ansible-playbook -i <playbook-dir>/<env>_azure_rm.yml <playbook-dir>/appd.yml --limit <env> --extra-vars ng_server_env=<env> --vault-password-file ~/.vault_pass.txt
```

Example:

```
/usr/bin/ansible-playbook -i ng-crimson-wlp-azure/encoreqa_azure_rm.yml ng-crimson-wlp-azure/appd.yml --limit encoreqa --extra-vars ng_server_env=encoreqa --vault-password-file ~/.vault_pass.txt
```

4) Once Appdynamics installation done login to appd controller to check the metrics

<http://sgdc2apds001t:8090/controller/> [QA/DEV]

<https://in-apds-cluster1-psg00.asia.essilor.group/controller/> [PROD]

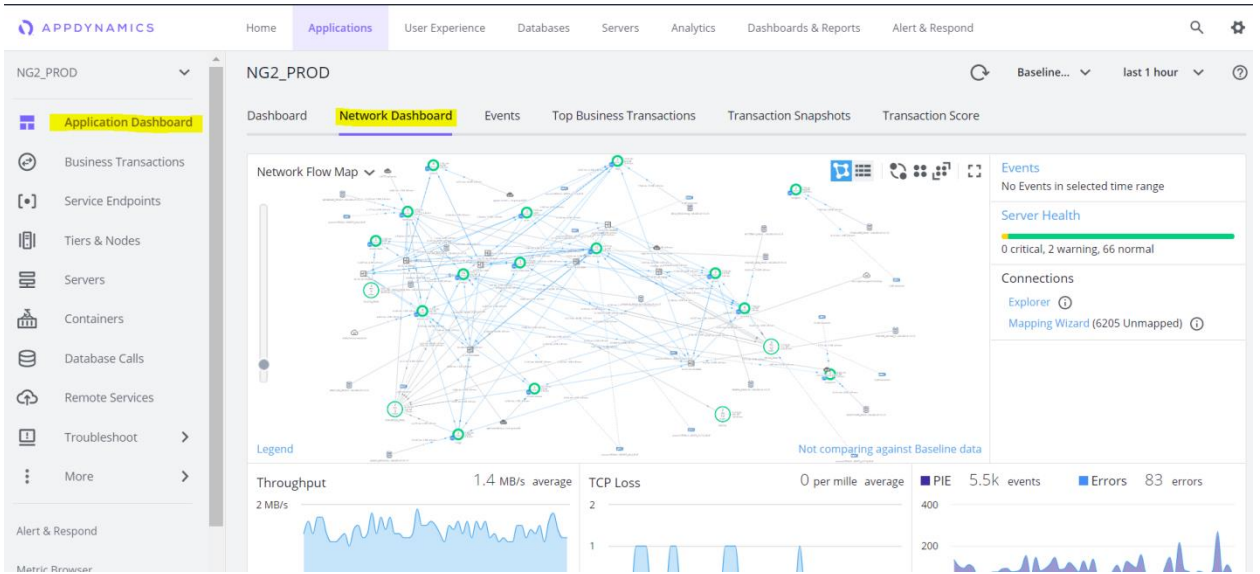
5) Go to the application tab and click on the application name

6) On the left side panel you can see **Tiers & Nodes** click on it

7) If appd installed is done then you can see **App Agent Status** as well as **Machine Agent**

The screenshot displays the AppDynamics 'Tiers & Nodes' page. The left sidebar shows the navigation menu with 'Tiers & Nodes' selected. The main panel shows a table of application tiers and their nodes. The 'Health' column shows green checkmarks for all nodes. The 'App Agent Status' column shows '↑ 98.3%' for the 'vega' nodes. The 'Machine Agent' column shows '↑ 98.3%' for the 'vega' nodes. The 'JVM Version' column shows 'OpenJDK 64-Bit Ser' for the 'vega' nodes. The 'Last JVM Restar...' column shows '07/25/21 7:17:55 AI' for the 'vega' nodes.

Name	#...	Health	App Agent Status	App Agent Vers...	Machine Agent ...	JVM Version	Last JVM Restar...
vega	2	✓	↑ 98.3%	Server Agent #20.9	↑ 98.3%	OpenJDK 64-Bit Ser	07/25/21 7:17:55 AI
vega_vegs-prd-vss-sea-farm01-001_7		✓	↑ 98.3%	Server Agent #20.9	↑ 98.3%	OpenJDK 64-Bit Ser	07/25/21 7:17:55 AI
vega_vegs-prd-vss-sea-farm01-001_0		✓	↑ 98.3%	Server Agent #20.9	↑ 98.3%	OpenJDK 64-Bit Ser	07/25/21 7:17:15 AI
encore-arcade	2	✓					
indigo	2	✓					
dispatcher	2	✓					
stargazer	2	✓					
encore-mdm	2	✓					
orion	3	✓					
genesis	3	✓					
encore-order-flow	3	✓					
encore-erp	4	✓					
mandrake	7	✓					



Clients:

- 1) For clients no need to add `app_application_name` you can directly installation using `playbook` command.

```
/usr/bin/ansible-playbook -i <playbook-dir>/<env>_azure_rm.yml <playbook-dir>/appd.yml -  
-limit <env> --extra-vars ng_server_env=<env> --vault-password-file ~/.vault_pass.txt
```

Now we need to get app key, adrum.js and code snippet to include in the nginx.conf from controller .

- 2) Firstly Go to controller → User Experience Tab

<http://sgdc2apds001t:8090/controller/> [QA/DEV]

<https://in-apds-cluster1-psg00.asia.essilor.group/controller/> [PROD]

[illegible]

- 3) Click on **Add app +** → **Diagloue box** appears to create app name -- > choose to create app manually and enter app name

Create Application

Create an Application using the Getting Started Wizard

Create an Application manually

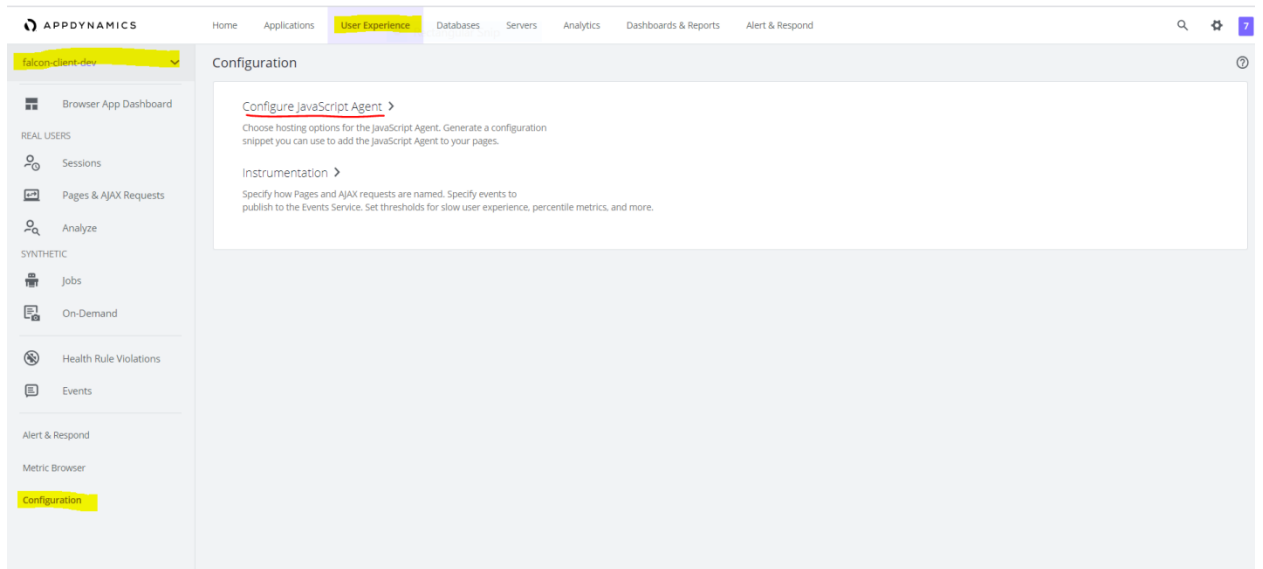
Enter application name

Cancel

OK

- 4) After creating application name , on the left side panel click on **Configuration** -- > **Configure JavaScript Agent**

Ref: below screen:



5) Now you can view your app key on next page. As next step **choose a JavaScript Agent hosting option --> I will host all JavaScript Agent Files**

6) Then add http url of the app of assets location

APPDYNAMICS Home Applications **User Experience** Databases Servers Analytics Dashboards & Reports Alert & Respond

falcon-client-dev

Configure JavaScript Agent

Your APP KEY IS: EUM-AAB-BAJ

No page view received for this application

These instructions help you manually instrument your web pages with the JavaScript Agent. But if you have a server application monitored with AppDynamics APM, you might be able to use [Automatic Injection](#) instead. Follow these steps to generate an HTML snippet that you can copy into your web pages.

Any change you make in this page will affect APM applications associated with this Browser Application. [Click here to see a list of associated APM applications.](#)

1 Choose a JavaScript Agent hosting option

☐ AppDynamics hosts all JavaScript Agent files from cdn.appdynamics.com

☒ I will host all the JavaScript Agent files

☐ I will host the main adrum-[version].js, but all other files should be loaded from cdn.appdynamics.com

Download and save the JavaScript Agent files to a location on your web server. Specify the URLs where the files can be referenced:

http://falcon-client.asia.essilor.group/assets

https://falcon-client.asia.essilor.group/assets

Hosting the JavaScript Agent

To minimize agent load times and overhead, and optimize caching, the AppDynamics JavaScript Agent is loaded from multiple parts.

The JavaScript Agent consists of three files: adrum-[version].js, adrum-ext-[version].js, and adrum-xd-[version].js.

adrum-[version].js is loaded synchronously when your webpage is loaded. The other two files - adrum-ext-[version].js & adrum-xd-[version].js - are loaded asynchronously.

To get you started quickly, AppDynamics hosts all the JavaScript Agent files at AppDynamics CDN. AppDynamics CDN is hosted on a highly reliable infrastructure, but might be less reliable than your infrastructure. So it is recommended that you load at least adrum-[version].js from your own infrastructure serving your web pages.

[Visit Docs](#)

7) After adding url you can download adrum.js file from the controller in the same page Or else we can take adrum file from any one of the client playbooks.

8) Finally Click on **Save Config and Generate HTML Snippet**. You will see code snippet

APPDYNAMICS Home Applications **User Experience** Databases Servers Analytics Dashboards & Reports Alert & Respond

falcon-client-dev

Configure JavaScript Agent

Download and save the JavaScript Agent files to a location on your web server. Specify the URLs where the files can be referenced:

http://falcon-client.asia.essilor.group/assets

https://falcon-client.asia.essilor.group/assets

adrum-ext-[version].js
adrum-xd-[version].js
adrum-[version].js is loaded synchronously when your webpage is loaded. The other two files - adrum-ext-[version].js & adrum-xd-[version].js - are loaded asynchronously.

To get you started quickly, AppDynamics hosts all the JavaScript Agent files at AppDynamics CDN. AppDynamics CDN is hosted on a highly reliable infrastructure, but might be less reliable than your infrastructure. So it is recommended that you load at least adrum-[version].js from your own infrastructure serving your web pages.

[Visit Docs](#)

2 Advanced >

3 Save Config & Generate HTML Snippet

Include this HTML snippet in your pages immediately after the head tag:

```
varset="UTF-8";
"un-start-time" = new Date().getTime();
onfig({
  .appkey = "EUM-AAB-BAJ";
  .adrumExtUrlHttp = "http://falcon-client.asia.essilor.group/assets";
  .adrumExtUrlHttps = "https://falcon-client.asia.essilor.group/assets";
  .beaconUrlHttp = "http://sgdc3apds002t:7001";
  .beaconUrlHttps = "https://sgdc3apds002t:7002";
  .xd = {enable : false};
  'adrum-config' || (window['adrum-config'] = {});
})
<script src="//falcon-client.asia.essilor.group/assets/adrum/adrum.js"></script>
```

Insert Generated HTML snippet

Add the generated snippet to all HTML pages that should be monitored by the JavaScript Agent.

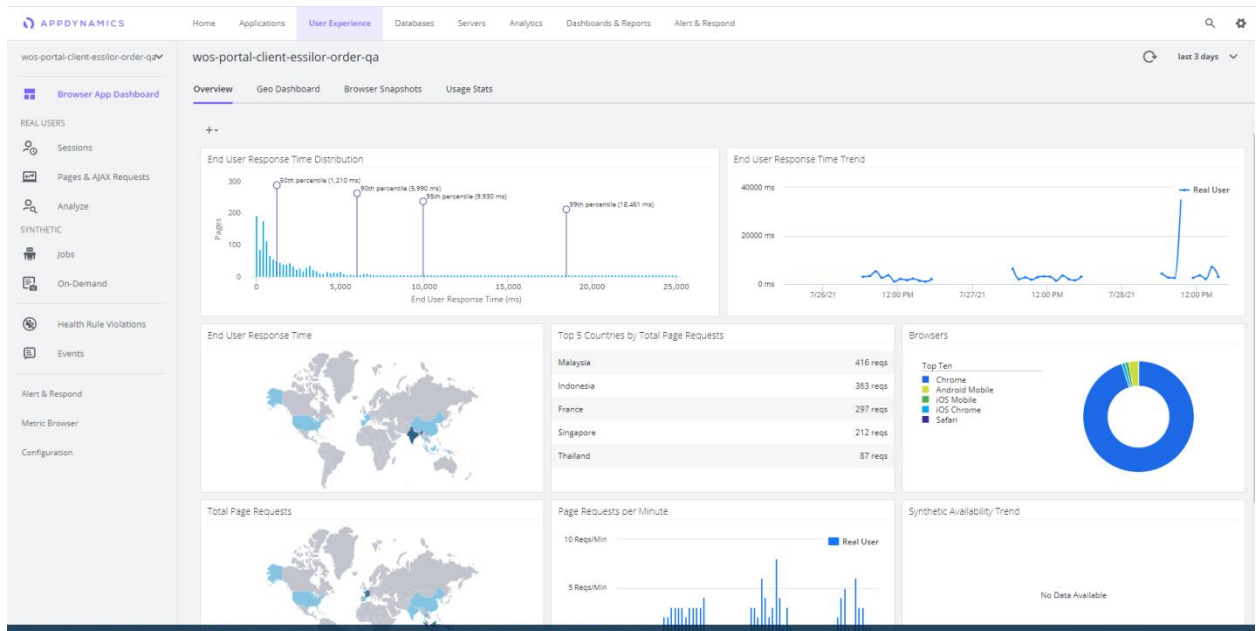
You should regenerate the snippet if you make any changes in steps 1 to 3.

[Visit Docs](#)

- 9) Finally go to client playbooks and edit nginx conf to add following code snippet(Refer existing client playbooks) Change appkey and http url values for existing code.

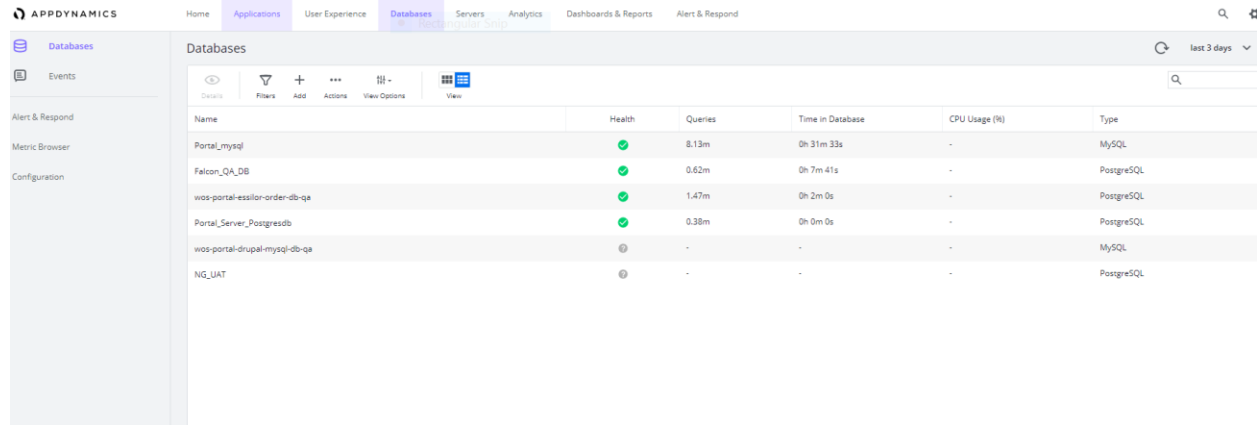
```
<script charset='UTF-8'>
window['adrum-start-time'] = new Date().getTime();
(function(config){
    config.appKey = 'EUM-AAB-BAJ';
    config.adrumExtUrlHttp = 'http://falcon-client.asia.essilor.group/assets';
    config.adrumExtUrlHttps = 'https://falcon-client.asia.essilor.group/assets';
    config.beaconUrlHttp = 'http://sgdc2apds002t:7001';
    config.beaconUrlHttps = 'https://sgdc2apds002t:7002';
    config.xd = {enable : false};
})(window['adrum-config'] || (window['adrum-config'] = {}));
</script>
<script src='//falcon-client.asia.essilor.group/assets/adrum/adrum.js'></script>
```

- 10) After deploying latest changes go to controller to view end user metrics from client apps.



DATABASES:

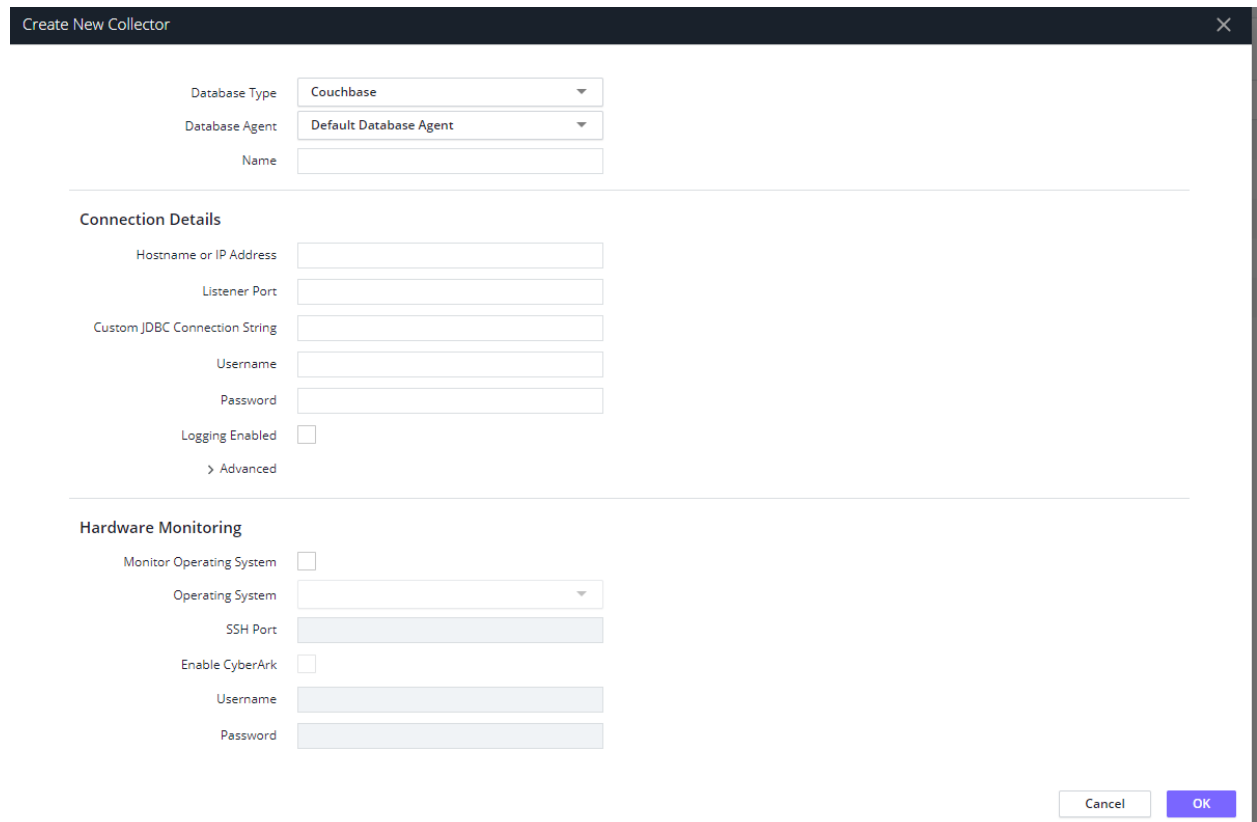
1) To add databases agent --> Go to Controller --> Databases Tab --> click on Add



The screenshot shows the AppDynamics interface with the 'Databases' tab selected. The table lists the following databases:

Name	Health	Queries	Time in Database	CPU Usage (R)	Type
Portal_mysql	✓	8.13m	0h 31m 33s	-	MySQL
Falcon_QA_DB	✓	0.62m	0h 7m 41s	-	PostgreSQL
wos-portal-esslon-order-db-qa	✓	1.47m	0h 2m 0s	-	PostgreSQL
Portal_Server_Postgresdb	✓	0.38m	0h 0m 0s	-	PostgreSQL
wos-portal-drupal-mysql-db-qa	?	-	-	-	MySQL
NG_UAT	?	-	-	-	PostgreSQL

2) Enter Database Type, Name, Hostname, Listener Port, UserName and Password



The 'Create New Collector' dialog box contains the following sections:

- Database Type:** Couchbase
- Database Agent:** Default Database Agent
- Name:** (empty text field)
- Connection Details:**
 - Hostname or IP Address: (empty text field)
 - Listener Port: (empty text field)
 - Custom JDBC Connection String: (empty text field)
 - Username: (empty text field)
 - Password: (empty text field)
 - Logging Enabled: ☐
 - > Advanced
- Hardware Monitoring:**
 - Monitor Operating System: ☐
 - Operating System: (empty dropdown menu)
 - SSH Port: (empty text field)
 - Enable CyberArk: ☐
 - Username: (empty text field)
 - Password: (empty text field)

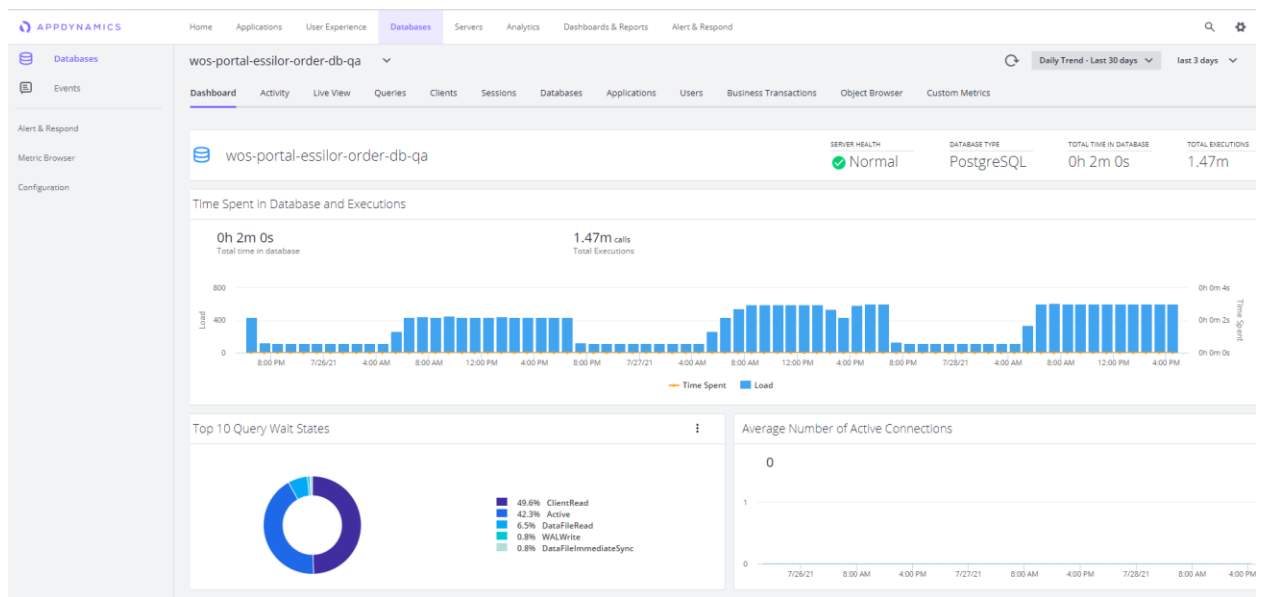
Buttons: Cancel, OK

3) Upon successfull addition you can view metrics by clicking on database name

Databases

last 3 days

Name	Health	Queries	Time in Database	CPU Usage (%)	Type
Portal_mysql	✓	8.13m	0h 31m 33s	-	MySQL
Falcon_QA_DB	✓	0.62m	0h 7m 41s	-	PostgreSQL
wos-portal-essilor-order-db-qa	✓	1.47m	0h 2m 0s	-	PostgreSQL
Portal_Server_Postgresdb	✓	0.38m	0h 0m 0s	-	PostgreSQL
wos-portal-drupal-mysql-db-qa	⚠	-	-	-	MySQL
NG_UAT	⚠	-	-	-	PostgreSQL



Analytics Agent :

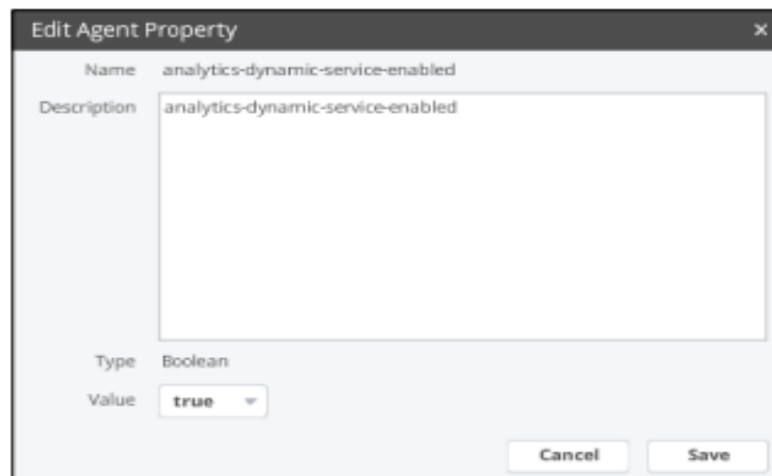
Installation & Configuration:

Download and install either the Standalone Machine Agent or the Analytics Agent (standalone binary) on each machine.

Enable Analytics Agent as Standalone Machine Agent Extension

The Analytics Agent is implemented as an extension to the Standalone Machine Agent (and runs as a machine agent monitor). In environments with the Standalone Machine Agent already running, you enable and run the Analytics Agent as an extension. Use the following steps:

1. Add analytics-dynamic-service-enabled property in app server agent configuration for analytics agent below 4.3 version. Path: Go to **Tiers & Nodes** → **Agents** → **App Server agents** → **Configure** → + (create property)



2. On the host running the Standalone Machine Agent, use a text editor to open **/monitors/analytics-agent/monitor.xml**

Set the enabled tag to true as follows, saving the file when you are finished:

```
<monitor>
  <name>AppDynamics Analytics Agent</name>
  <type>managed</type>
  <!-- Enabling this requires JRE 7 or higher -->
  <enabled>true</enabled>
```

3. Configure connectivity from the analytics-agent to the Events Service by editing the analyticsagent.properties file:
/monitors/analytics-agent/conf/analytics-agent.properties

4. In the analytics-agent.properties file, edit the following fields
 - **http.event.endpoint=(Event Service IP:Port)**
 - **http.event.accountName=(Global Account Name)**
 - **http.event.accessKey = (Controller's Access Key)**
5. Save and close the file.
6. If the machine-agent is already running at this point, it needs to be restarted to pick up the changes in the configuration.

To start and stop the Analytics Agent without starting the machine agent (Linux):

/monitors/analytics-agent/bin/analytics-agent.sh

To start and stop the Analytics Agent (Windows):

You can use native windows services menu to start/stop the service or you can do it directly from command line using the following two commands.

To start the agent service from the command line: **bin\analytics-agent.exe service-start**

To stop the agent service for the command line: **bin\analytics-agent.exe service-stop**

To uninstall the Windows service:

Run the .exe file with the uninstall command as follows:

bin\analytics-agent.exe service-uninstall.