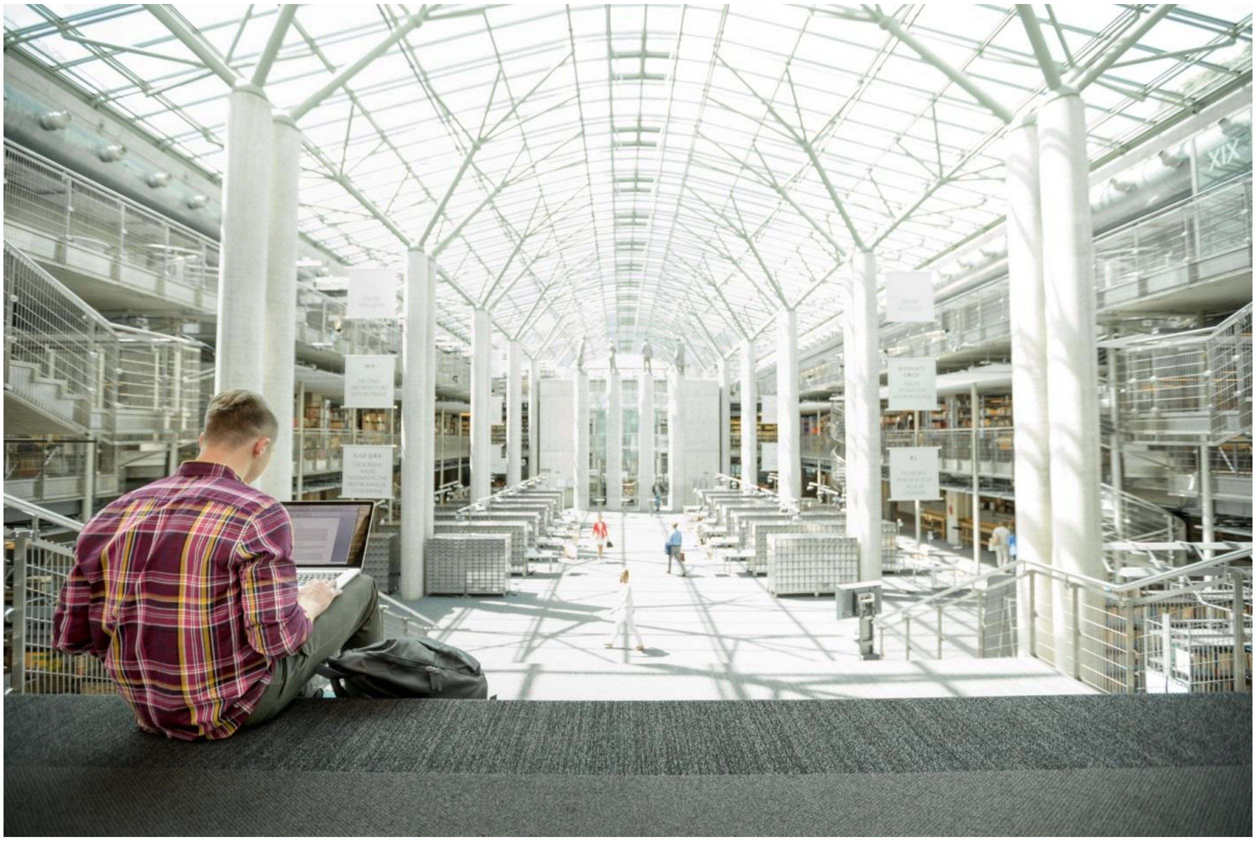


Cisco and AppDynamics SAP Monitoring Guide

Last Updated: August 14, 2021

Table of Contents

Executive Summary

AppDynamics is an Application Performance Monitoring (APM) Platform that helps you to understand and optimize the performance of your business, from its software to infrastructure to business journeys.

The AppDynamics APM Platform enables you to monitor and manage your entire application-delivery ecosystem, from the mobile app or browser client request through your network, backend databases and application servers and more. AppDynamics APM gives you a single view across your application landscape, letting you quickly navigate from the global perspective of your distributed application right down to the call graphs or exception reports generated on individual hosts.

AppDynamics has an agent based architecture. Once our agents are installed it gives you a dynamic flow map or topography of your application. It uses the concept of traffic lights to indicate the health of your application (green is good, yellow is slow and red indicates potential issues) with dynamic baselining. AppDynamics measures application performance based on business transactions which essentially are the key functionality of the application. When the application deviates from the baseline AppDynamics captures and provides deeper diagnostic information to help be more proactive in troubleshooting and reduce the MTTR (mean time to resolution).

Solution overview

Introduction

AppDynamics has a one of its kind ABAP agent for monitoring SAP ABAP systems. We have comprehensive coverage of the SAP landscape with our ABAP, Java, .net and Server visibility agents. In addition, Datavard Insights extends the AppDynamics for SAP solution with system-level monitoring for the overall SAP systems and SAP HANA databases. While AppDynamics agents provides transaction-level visibility, Datavard Insights collects performance metrics, logs and events, including processes outside of the user business transactions, such as background jobs or IDocs processing.

The complex and proprietary nature of SAP applications makes it difficult to diagnose issues. AppDynamics allows enterprises to instrument SAP applications, monitor performance, and understand the root cause of performance bottlenecks.

Audience

The intended audience for this document includes sales engineers, field consultants, professional services staff, IT managers, partner engineers, and customers deploying the Cisco® AppDynamics SAP monitoring solution for SAP HANA. External references are provided wherever applicable, but readers are expected to be familiar with the technology, infrastructure, and database security policies of the customer installation.

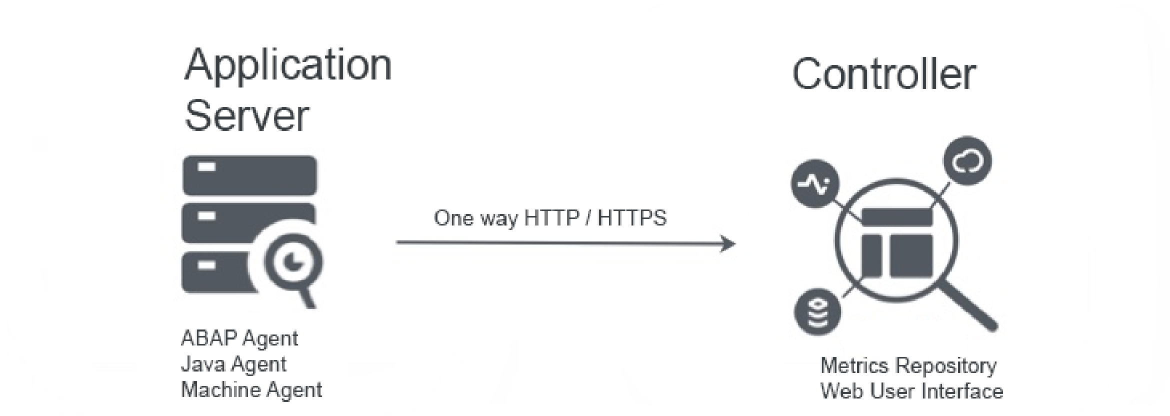
Purpose of this document

This document describes the steps required to configure Cisco® AppDynamics SAP monitoring solution for SAP HANA. This document focuses on one of the variants of Cisco's solution for SAP HANA. Although readers of this document are expected to have the knowledge needed to install and configure the products used, configuration details that are important to the deployment of this solution are provided in this document.

Solution summary

The AppDynamics architecture consists of two primary component types:

1. Software based Agents
    - agent binaries installed on the application servers
    - collects transactional diagnostic data and metrics
    - sends collected data to controller over one-way HTTP / HTTPS connection
2. Software based Controller
    - single centralized management server
    - repository for data collected by the agents
    - secure web-based user interface
    - SaaS or On-Premise option (*SaaS Controller is utilized during the proof of value trial*)


Application Server
ABAP Agent
Java Agent
Machine Agent
One way HTTP / HTTPS
Controller
Metrics Repository
Web User Interface

Deployment of Software

How to get started with AppDynamics

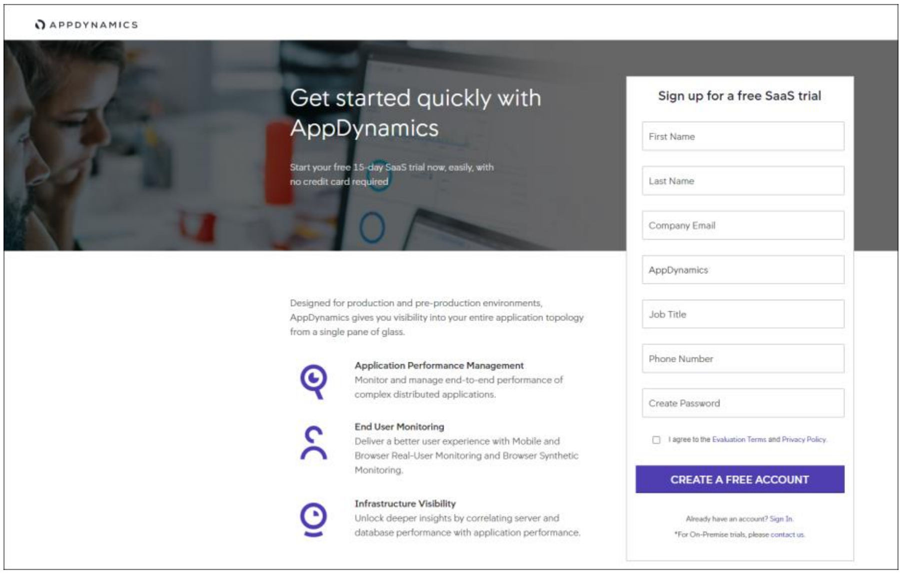
AppDynamics has several types of agents to monitor different language applications to user Experience to Infrastructure monitoring. Based on the SAP landscape and the underlying technology of the SAP systems the agents are installed.

The most frequently installed agents are:

1. Java Agent - For Java based SAP Systems
2. ABAP Agent - For ABAP based SAP systems
3. Server Visibility Monitoring - Provides extended hardware metrics and Service Availability Monitoring

Trial Registration

1. Connect to https://www.appdynamics.com/free-trial/
2. Provide the details to sign up for a free trial utilizing an AppDynamics SaaS controller



Get started quickly with AppDynamics

Start your free 15-day SaaS trial now, easily, with no credit card required

Designed for production and pre-production environments, AppDynamics gives you visibility into your entire application topology from a single pane of glass.

Application Performance Management
Monitor and manage end-to-end performance of complex distributed applications.

End User Monitoring
Deliver a better user experience with Mobile and Browser Real-User Monitoring and Browser Synthetic Monitoring.

Infrastructure Visibility
Unlock deeper insights by correlating server and database performance with application performance.

Sign up for a free SaaS trial

First Name

Last Name

Company Email

AppDynamics

Job Title

Phone Number

Create Password

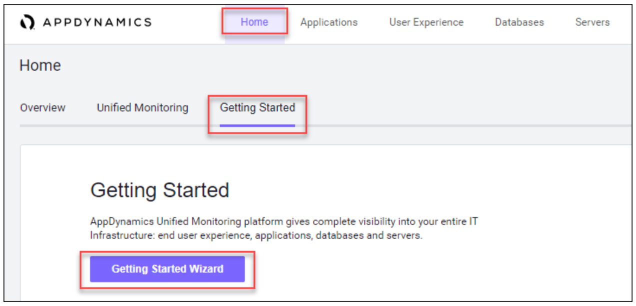
□ I agree to the Evaluation Terms and Privacy Policy.

CREATE A FREE ACCOUNT

Already have an account? Sign In.

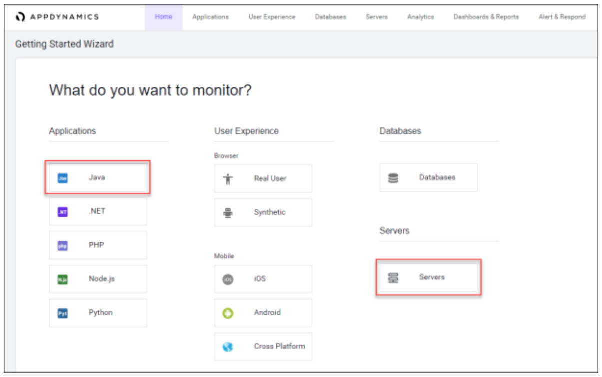
*For On-Premise trials, please contact us.

3. Once the AppDynamics SaaS Controller has been provisioned, you will receive an email with the information you need for you to login to the Controller.
4. You can download the Java Agent and the Machine / Server Visibility Agent directly from the Controller



Getting Started

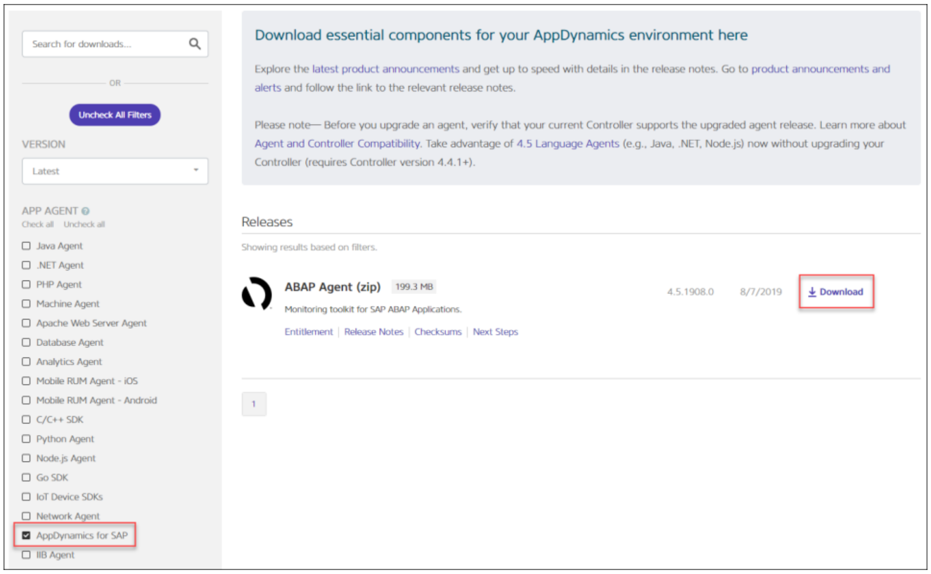
AppDynamics Unified Monitoring platform gives complete visibility into your entire IT Infrastructure: end user experience, applications, databases and servers.



What do you want to monitor?

5. You can use the email and password you provided to sign up for the trial to login to the agent download site at the URL listed below and download the ABAP Agent:

   https://download.appdynamics.com



Download essential components for your AppDynamics environment here

Explore the latest product announcements and get up to speed with details in the release notes. Go to product announcements and alerts and follow the link to the relevant release notes.

Please note— Before you upgrade an agent, verify that your current Controller supports the upgraded agent release. Learn more about Agent and Controller Compatibility. Take advantage of 4.5 Language Agents (e.g., Java, .NET, Node.js) now without upgrading your Controller (requires Controller version 4.4.1+).

☐ Java Agent
☐ .NET Agent
☐ PHP Agent
☐ Machine Agent
☐ Apache Web Server Agent
☐ Database Agent
☐ Analytics Agent
☐ Mobile RUM Agent - iOS
☐ Mobile RUM Agent - Android
☐ C/C++ SDK
☐ Python Agent
☐ Node.js Agent
☐ Go SDK
☐ IoT Device SDKs
☐ Network Agent
☑ AppDynamics for SAP
☐ IIB Agent

Showing results based on filters.

ABAP Agent (zip) 199.3 MB 4.5.1908.0 8/7/2019

Monitoring toolkit for SAP ABAP Applications.

AppDynamics Controller and Agent Properties

NOTE: we will only reference SaaS properties, not on-prem

There are several properties needed to configure the different agents and many of those properties can be found in the AppDynamics Controller.

The values for each of these properties will be unique based on the instance of the AppDynamics Controller in use and the SAP system that an AppDynamics agent is configured for.

The example values seen in the instructions and screenshots for the configuration instructions in the following sections are for reference only. Please follow the steps described in this section to obtain the actual property values specific for your environment.

1. **controller-host**

   - This is the fully qualified host name of the controller. This is the same host that you use to access the AppDynamics browser-based User interface.

- Example: “*mycompany.saas.appdynamics.com*”
- For more information about SaaS hosts see https://docs.appdynamics.com/display/PAA/SaaS+Domains+and+IP+Ranges

2. **controller-port**

    - This is the http(s) port of the AppDynamics Controller. This is the same port that you use to access the AppDynamics browser-based User interface.
    - If “*controller-ssl-enabled*” (next property below) is set to true, you must specify the HTTPS port of the controller; otherwise, specify the HTTP port.
    - If you are using a SaaS Controller you must use port “*443*” and set “*controller-ssl-enabled*” property to “*true*”
    - For more information about port values see https://docs.appdynamics.com/display/PRO45/Port+Settings

3. **controller-ssl-enabled**

    - This specifies if the AppDynamics agent should use SSL (HTTPS) to connect to the Controller. If you set this to true, the “*controller-port*” property above should be set to the HTTPS port of the Controller.
    - If you are using a SaaS Controller you must use port “*443*” and set the “*controller-ssl-enabled*” property to “*true*”

4. **account-name**

    - If the AppDynamics Controller is running in multi-tenant mode or you are using the AppDynamics SaaS Controller, you must specify the account name for the agent to authenticate with the controller.
    - See the next section titled “*Finding Agent Properties in Controller*” for more information about obtaining this property value.

5. **account-access-key**

    - This key is generated by the controller and can be found by viewing the license information in the controller settings.
    - See the next section titled “*Finding Agent Properties in Controller*” for more information about obtaining this property value.

6. **global-account-name**

    - This key is generated by the controller and can be found by viewing the license information in the controller settings.
    - See the next section titled “*Finding Agent Properties in Controller*” for more information about obtaining this property value.

7. **analytics-events-service-host**

    - This is the fully qualified host name of the Analytics Events Service
    - If you are using a SaaS Controller, then you should use one of the following Analytics Events Service hosts depending on your geo location:
        - Americas = “*analytics.api.appdynamics.com*”

- EMEA = "*fra-ana-api.saas.appdynamics.com*"
- APAC = "*syd-ana-api.saas.appdynamics.com*"

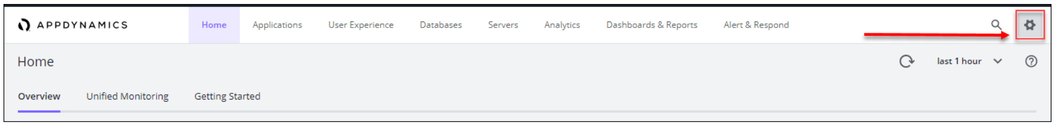
- For more information about SaaS hosts see https://docs.appdynamics.com/display/PAA/SaaS+Domains+and+IP+Ranges

8. **analytics-events-service-port**

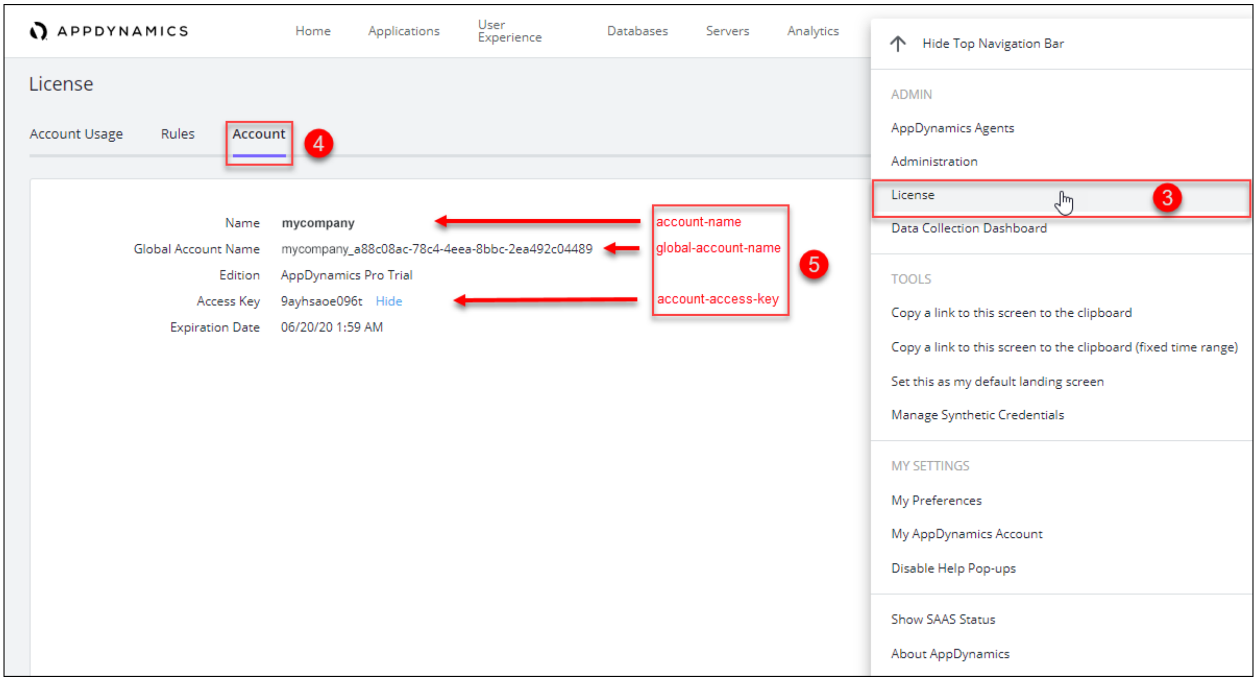
- If you are using a SaaS Analytics Events Service, the default values are 80 for HTTP and 443 for HTTPS.

Finding Agent Properties in Controller

1. Log into the Controller
2. Click on the gear icon located at the top right of the screen



3. Click on the "*License*" option from the drop-down menu
4. Click on the "*Account*" tab
5. Now you should see the "*account-name*", "*global-account-name*", and the "*account-access-key*"



Name **mycompany**
Global Account Name mycompany_a88c08ac-78c4-4eea-8bbc-2ea492c04489
Edition AppDynamics Pro Trial
Access Key 9ayhsaoe096t Hide
Expiration Date 06/20/20 1:59 AM

Agent Directory Structures

On the machines where you are going to install AppDynamics Agents, the "*/usr/sap/appdyn*" directory is recommended; however, an alternate directory can be used. This will be the parent directory where all the agents will be installed.

The following directory paths are suggested as the permanent installation locations for the different agent types:

- Java Agent (for Java servers)
  - */usr/sap/appdyn/app*
- ABAP Agent (for ABAP servers)
  - */usr/sap/appdyn/appdhttpsd*
- Machine / Server Visibility Agent (for all Java and ABAP servers)
  - */usr/sap/appdyn/machine*

The binary for each agent type (i.e. ZIP file) will need to be downloaded into a temporary directory on the appropriate servers and then unzipped and or copied to their permanent directory.

Ensure you are logged in as the appropriate *<SID>adm* or *SAPService<SID>* OS user when creating directories, unzipping agent files, and copying them to other directories.

Provide the OS user, *<SID>adm* or *SAPService<SID>*, with full permission (*CHMOD 775*) on the temporary and permanent directories (including all agent sub-directories) where the agents are placed on each application server node.

Agent Directory Creation Steps

On the machine(s) where you are going to install AppDynamics Agents, create the "*appdyn*" directory which will be the location where all the agents will be installed.

1. Become the "*sidadm*" user. For example, if the server is an SAP application server, then you would execute the following commands:

   ```
   cd

   . ./.profile
   ```

2. Create the "*appdyn*" directory as a subdirectory under "*/usr/sap*":

   ```
   cd /usr/sap

   mkdir appdyn
   ```

3. Set the permissions to "*775*" on the new directory:

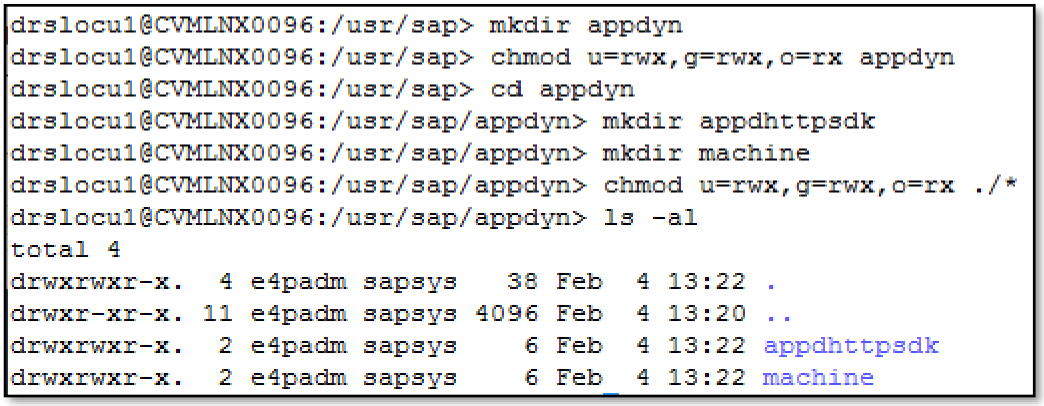
   ```
   chmod u=rwx,g=rwx,o=rx appdyn
   ```

4. Create the "*app*", "*machine*", and "*appdhttpsdk*" directories within the "*appdyn*" directory:

   ```
   cd appdyn

   mkdir app  (only for Java stack)

   mkdir appdhttpsdk  (only for ABAP stack)

   mkdir machine  (for machine / server visibility agent)
   ```

5. Set the permissions to "*775*" on all the directories created in step 4:

   ```
   chmod u=rwx,g=rwx,o=rx ./*
   ```



```
drslocu1@CVMLNX0096:/usr/sap> mkdir appdyn
drslocu1@CVMLNX0096:/usr/sap> chmod u=rwx,g=rwx,o=rx appdyn
drslocu1@CVMLNX0096:/usr/sap> cd appdyn
drslocu1@CVMLNX0096:/usr/sap/appdyn> mkdir appdhttpsdk
drslocu1@CVMLNX0096:/usr/sap/appdyn> mkdir machine
drslocu1@CVMLNX0096:/usr/sap/appdyn> chmod u=rwx,g=rwx,o=rx ./*
drslocu1@CVMLNX0096:/usr/sap/appdyn> ls -al
total 4
drwxrwxr-x.  4 e4padm sapsys   38 Feb  4 13:22 .
drwxr-xr-x. 11 e4padm sapsys 4096 Feb  4 13:20 ..
drwxrwxr-x.  2 e4padm sapsys    6 Feb  4 13:22 appdhttpsdk
drwxrwxr-x.  2 e4padm sapsys    6 Feb  4 13:22 machine
```

Install the Java Agent

The Java Agent must be installed on SAP Java application servers (e.g. Enterprise Portal and PO application servers). The agent binary ZIP file will be named like “*AppServerAgent-4.5.7.24863.zip*”.

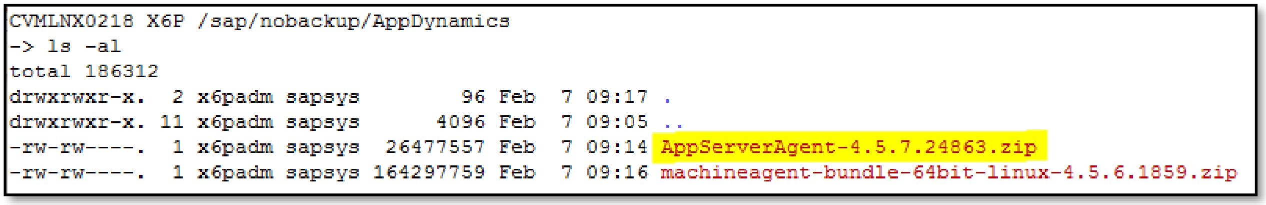
1. Download the Java Agent from the location mentioned earlier in this document

   See the section titled “Download AppDynamics Agents”

2. Save the agent zip file to a temporary location on the server, such as “*/sap/nobackup/AppDynamics*”

   Be sure to give the “*sidadm*” user full read/write/execute access to this directory.

   You should login as the “*sidadm*” user and then create the directory so that the “*sidadm*” user becomes the owner of the directory.



```
CVMLNX0218 X6P /sap/nobackup/AppDynamics
-> ls -al
total 186312
drwxrwxr-x.  2 x6padm sapsys        96 Feb  7 09:17 .
drwxrwxr-x. 11 x6padm sapsys      4096 Feb  7 09:05 ..
-rw-rw----.  1 x6padm sapsys  26477557 Feb  7 09:14 AppServerAgent-4.5.7.24863.zip
-rw-rw----.  1 x6padm sapsys 164297759 Feb  7 09:16 machineagent-bundle-64bit-linux-4.5.6.1859.zip
```

3. Unzip the Java agent zip file into the “*/usr/sap/appdyn/app*” directory:

   ```
   unzip AppServerAgent-4.5.7.24863.zip -d /usr/sap/appdyn/app
   ```

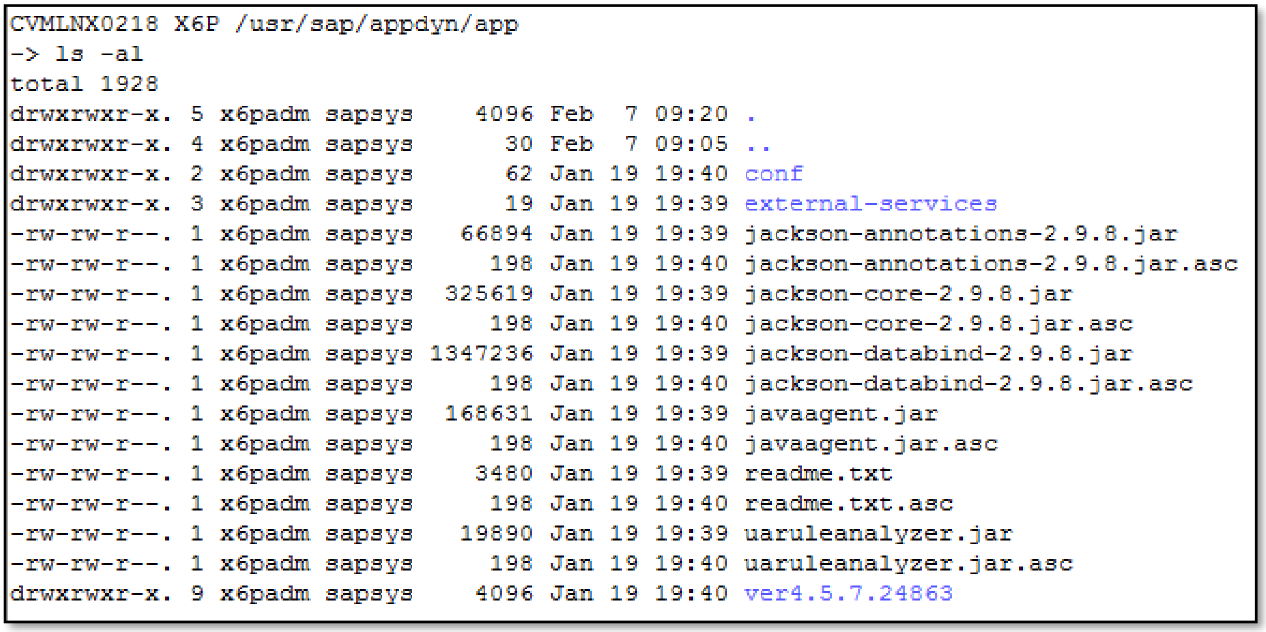
4. For Java applications which communicate with SAP systems using SAP JCo library (e.g. SAP PO app servers), we also need to copy an additional jar file from the ABAP agent installation directory. For more information see the link below:

   Configuring AppDynamics Java Agent in SAP

5. Change the permissions on the contents of the “*/usr/sap/appdyn/app*” directory to give full read/write privileges at the owner and group levels:

   ```
   cd /usr/sap/appdyn/app
   ```

```
chmod -R u+rw,g+rw,o-w ./*
```



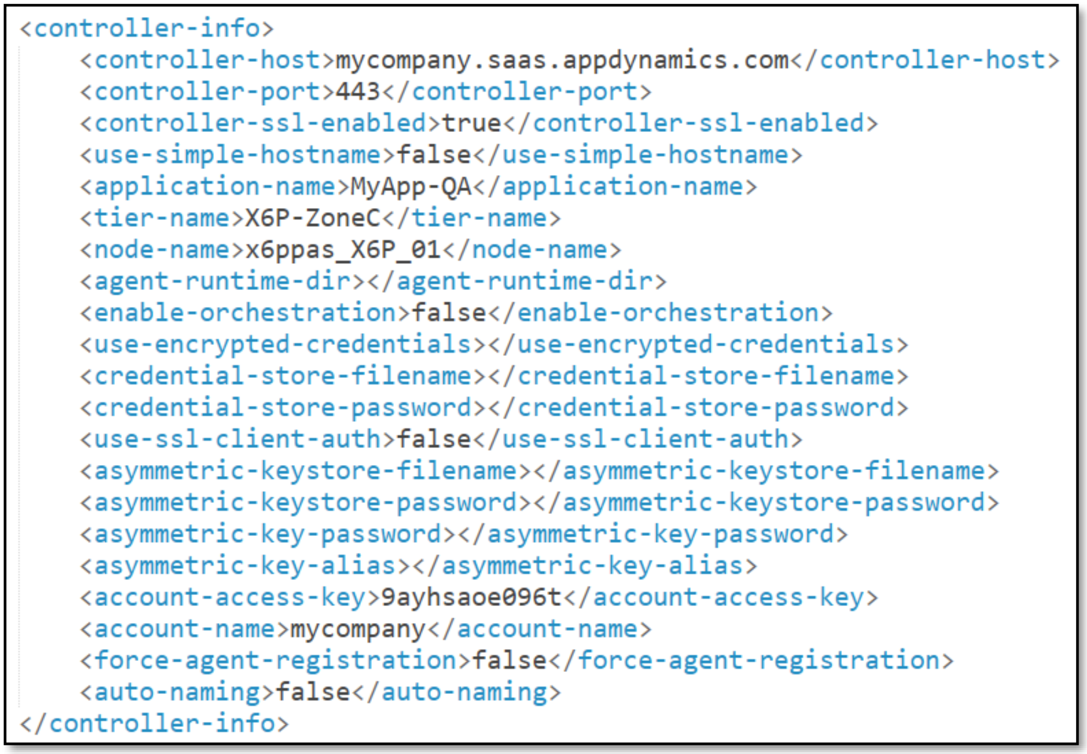
```
CVMLNX0218 X6P /usr/sap/appdyn/app
-> ls -al
total 1928
drwxrwxr-x. 5 x6padm sapsys    4096 Feb  7 09:20 .
drwxrwxr-x. 4 x6padm sapsys      30 Feb  7 09:05 ..
drwxrwxr-x. 2 x6padm sapsys      62 Jan 19 19:40 conf
drwxrwxr-x. 3 x6padm sapsys      19 Jan 19 19:39 external-services
-rw-rw-r--. 1 x6padm sapsys   66894 Jan 19 19:39 jackson-annotations-2.9.8.jar
-rw-rw-r--. 1 x6padm sapsys     198 Jan 19 19:40 jackson-annotations-2.9.8.jar.asc
-rw-rw-r--. 1 x6padm sapsys  325619 Jan 19 19:39 jackson-core-2.9.8.jar
-rw-rw-r--. 1 x6padm sapsys     198 Jan 19 19:40 jackson-core-2.9.8.jar.asc
-rw-rw-r--. 1 x6padm sapsys 1347236 Jan 19 19:39 jackson-databind-2.9.8.jar
-rw-rw-r--. 1 x6padm sapsys     198 Jan 19 19:40 jackson-databind-2.9.8.jar.asc
-rw-rw-r--. 1 x6padm sapsys  168631 Jan 19 19:39 javaagent.jar
-rw-rw-r--. 1 x6padm sapsys     198 Jan 19 19:40 javaagent.jar.asc
-rw-rw-r--. 1 x6padm sapsys    3480 Jan 19 19:39 readme.txt
-rw-rw-r--. 1 x6padm sapsys     198 Jan 19 19:40 readme.txt.asc
-rw-rw-r--. 1 x6padm sapsys   19890 Jan 19 19:39 uaruleanalyzer.jar
-rw-rw-r--. 1 x6padm sapsys     198 Jan 19 19:40 uaruleanalyzer.jar.asc
drwxrwxr-x. 9 x6padm sapsys    4096 Jan 19 19:40 ver4.5.7.24863
```

6. Edit the versioned "*controller-info.xml*" file located in the "*/usr/sap/appdyn/app/ver4.5.7.24863/conf*" directory. The name of the fifth directory in the file path will depend on the Java agent version downloaded. Set the following elements in the xml file.

Property Name	Actual Property Value	Example Property Value	Additional Info
controller-host	Refer to "*AppDynamics Controller and Agent Properties*" section above	mycompany.saas.appdynamics.com	
controller-port	Refer to "*AppDynamics Controller and Agent Properties*" section above	443	
controller-ssl-enabled	Refer to "*AppDynamics Controller and Agent Properties*" section above	true	
application-name	This value starts with the name of the application followed by a hyphen and the environment name	MyApp-QA	
tier-name	This value will start with the SAP system ID followed by a hyphen and the network zone (for example, for the X6P system which resides in Zone C, the name would be "X6P-ZoneC" without any spaces	X6P-ZoneC	
node-name	This value starts with the DNS alias name of the app server followed by a hyphen and then the SAP	x6ppas_X6P_01	

Property Name	Actual Property Value	Example Property Value	Additional Info
	system ID followed by another hyphen and finally the SAP instance number (for example, for the X6P primary application server, this name would be "x6ppas_X6P_01", which is consistent with the node names that get automatically generated for the ABAP agents)		
account-access-key	Refer to "*AppDynamics Controller and Agent Properties*" section above	9ayhsaoe096t	
account-name	Refer to "*AppDynamics Controller and Agent Properties*" section above	mycompany	
auto-naming	false	false	

The rest of the properties can be left as-is. This is what an example completed file looks like (all the comment blocks have been edited out to make it easier to read):



```
<controller-info>
    <controller-host>mycompany.saas.appdynamics.com</controller-host>
    <controller-port>443</controller-port>
    <controller-ssl-enabled>true</controller-ssl-enabled>
    <use-simple-hostname>false</use-simple-hostname>
    <application-name>MyApp-QA</application-name>
    <tier-name>X6P-ZoneC</tier-name>
    <node-name>x6ppas_X6P_01</node-name>
    <agent-runtime-dir></agent-runtime-dir>
    <enable-orchestration>false</enable-orchestration>
    <use-encrypted-credentials></use-encrypted-credentials>
    <credential-store-filename></credential-store-filename>
    <credential-store-password></credential-store-password>
    <use-ssl-client-auth>false</use-ssl-client-auth>
    <asymmetric-keystore-filename></asymmetric-keystore-filename>
    <asymmetric-keystore-password></asymmetric-keystore-password>
    <asymmetric-key-password></asymmetric-key-password>
    <asymmetric-key-alias></asymmetric-key-alias>
    <account-access-key>9ayhsaoe096t</account-access-key>
    <account-name>mycompany</account-name>
    <force-agent-registration>false</force-agent-registration>
    <auto-naming>false</auto-naming>
</controller-info>
```

7. You will need to add parameters to the SAP JVM to start the Java agent when the SAP system is started up by logging into the SAP app server as the "sidadm" user.

Use the SAP NetWeaver Administrator or the AS Java Config Tool (depending on your SAP system) to edit the JVM startup parameters. For more detailed information, see the link below:

Configuring AppDynamics Java Agent in SAP

8. Restart the SAP JVM for the settings to take effect

9. Validate the Java Agent is reporting to the controller by logging into the controller UI

Install the ABAP Agent

The ABAP Agent needs to be installed on SAP systems utilizing the ABAP stack. There are seven primary steps to perform:

1. Copy and unzip the ABAP Agent

2. Import the ABAP Agent Transports

3. Configure User Authorizations

4. Configure ABAP Agent and Install HTTP SDK

5. Activate Datavard Insight Collectors

6. Configure Generic Instrumentation

7. Validate the ABAP Agent Installation

Copy and unzip the ABAP Agent

The ABAP Agent binary ZIP file will be named like "*APPD-SAP-4.5.1904.2.zip*".

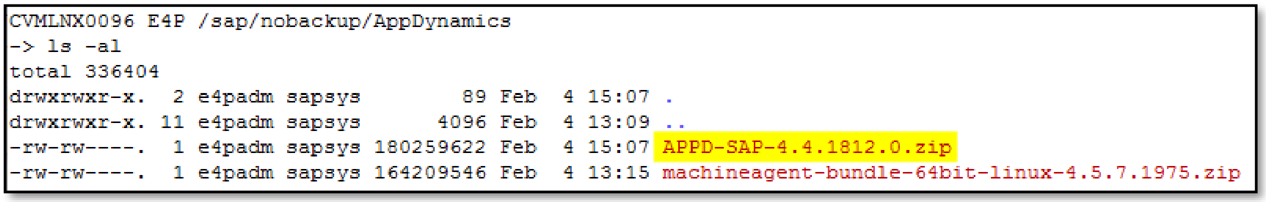
1. Download the ABAP Agent from the location mentioned earlier in this document. See the section titled "Download AppDynamics Agents"

   The ABAP agent zip file contains the ABAP transports, the HTTP SDK, the SAP_JCO_iSDK.jar, and the transports for uninstalling the ABAP agent.

2. Save the agent zip file to a temporary location on the server, such as "*/sap/nobackup/AppDynamics*"

   Be sure to give the "*sidadm*" user full read/write/execute access to this directory.

   You should login as the "*sidadm*" user and then create the directory so that the "*sidadm*" user becomes the owner of the directory.

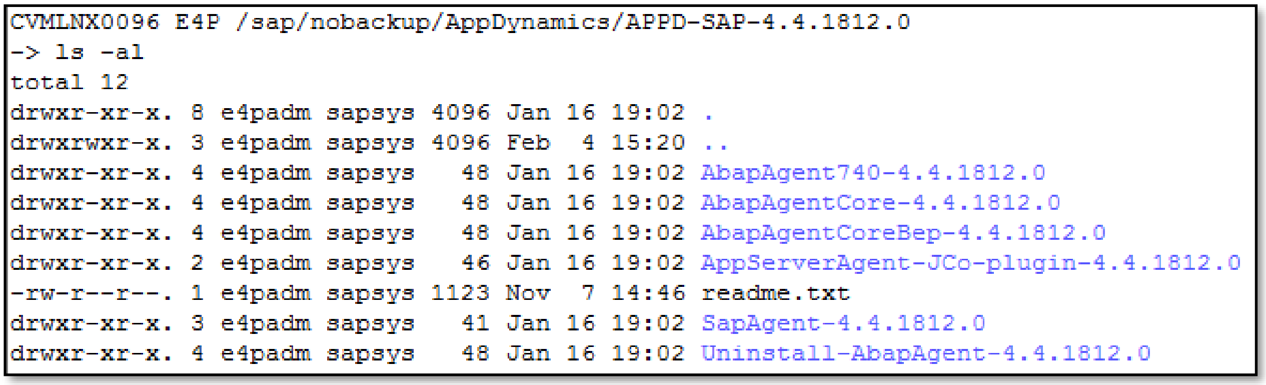


```
CVMLNX0096 E4P /sap/nobackup/AppDynamics
-> ls -al
total 336404
drwxrwxr-x.  2 e4padm sapsys        89 Feb  4 15:07 .
drwxrwxr-x. 11 e4padm sapsys      4096 Feb  4 13:09 ..
-rw-rw----.  1 e4padm sapsys 180259622 Feb  4 15:07 APPD-SAP-4.4.1812.0.zip
-rw-rw----.  1 e4padm sapsys 164209546 Feb  4 13:15 machineagent-bundle-64bit-linux-4.5.7.1975.zip
```

3. Unzip the ABAP agent zip file:

   ```
   unzip APPD-SAP-4.4.1812.0.zip
   ```

   This will create a directory named "*APPD-SAP-4.4.1812.0*" with the following contents:



```
CVMLNX0096 E4P /sap/nobackup/AppDynamics/APPD-SAP-4.4.1812.0
-> ls -al
total 12
drwxr-xr-x. 8 e4padm sapsys 4096 Jan 16 19:02 .
drwxrwxr-x. 3 e4padm sapsys 4096 Feb  4 15:20 ..
drwxr-xr-x. 4 e4padm sapsys   48 Jan 16 19:02 AbapAgent740-4.4.1812.0
drwxr-xr-x. 4 e4padm sapsys   48 Jan 16 19:02 AbapAgentCore-4.4.1812.0
drwxr-xr-x. 4 e4padm sapsys   48 Jan 16 19:02 AbapAgentCoreBep-4.4.1812.0
drwxr-xr-x. 2 e4padm sapsys   46 Jan 16 19:02 AppServerAgent-JCo-plugin-4.4.1812.0
-rw-r--r--. 1 e4padm sapsys 1123 Nov  7 14:46 readme.txt
drwxr-xr-x. 3 e4padm sapsys   41 Jan 16 19:02 SapAgent-4.4.1812.0
drwxr-xr-x. 4 e4padm sapsys   48 Jan 16 19:02 Uninstall-AbapAgent-4.4.1812.0
```

4. Change the permissions on the contents of the "*APPD-SAP-4.4.1812.0*" directory to give full read/write privileges at the owner and group levels:

   ```
   cd /sap/nobackup/AppDynamics

   chmod -R u+rw,g+rw ./APPD-SAP-4.4.1812.0
   ```

Import the ABAP Agent Transports

There are three vendor transports that need to be installed in the SAP ABAP application. Ensure the transports are imported in the order seen below. For more information about the specific transport names, please view the "readme.txt" file in the agent directory after unzipping the agent. The relevant transports are:

- Datavard Reuse Library
- Datavard Insights
- AppDynamics ABAP agent 7.40 (on <7.40 import "ABAP agent core" transport)

1. Copy the cofiles and data files to the correct transport directory. For example, for ECC we would copy the transports to "*/usr/sap/trans/ECC2/cofiles*" and "*/usr/sap/trans/ECC2/data*", respectively.

   (NOTE: Be sure to login with the "*sidadm*" user before executing these commands.)

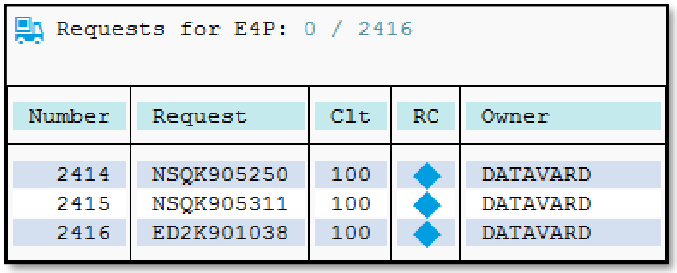
   ```
   cd /usr/sap/trans/ECC2/cofiles

   cp /sap/nobackup/AppDynamics/APPD-SAP-4.4.1812.0/AbapAgent740-
   4.4.1812.0/cofiles/* .

   cd ../data

   cp /sap/nobackup/AppDynamics/APPD-SAP-4.4.1812.0/AbapAgent740-4.4.1812.0/data/* .
   ```

2. Set the permissions on the cofiles and data files to allow read/write access at the owner and group levels:

```
cd /usr/sap/trans/ECC2/cofiles

chmod u+rw,g+rw *.ED2 *.NSQ

cd ../data

chmod u+rw,g+rw *.ED2 *.NSQ
```

3. Login to the SAP system, execute transaction STMS, and go to the import queue of the system where you want to install the ABAP agent.

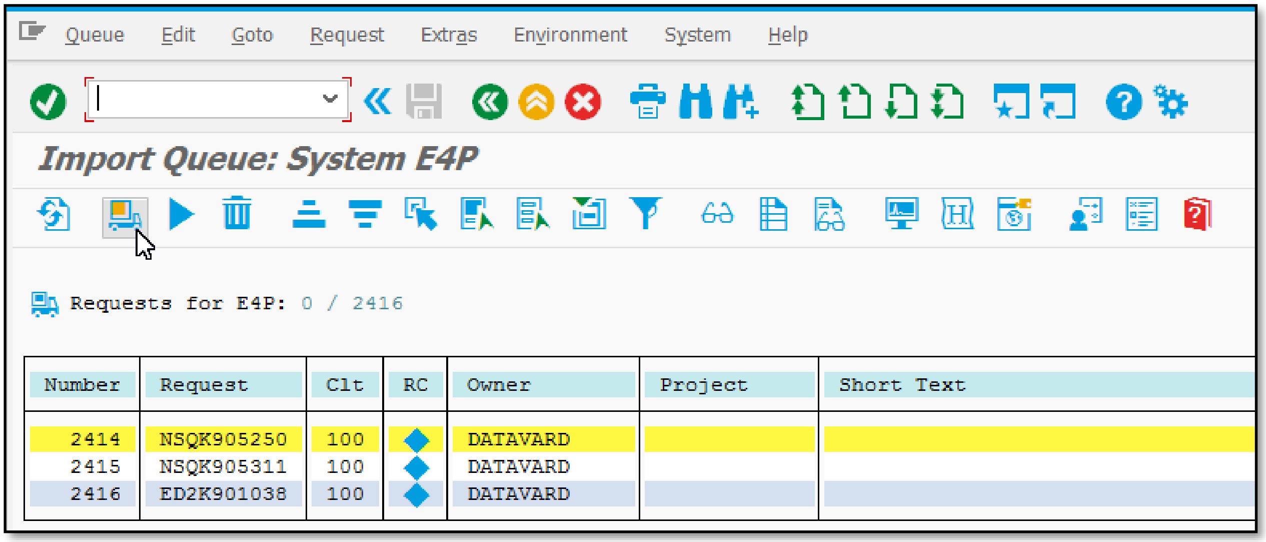
4. Select "*Extras > Other Requests > Add*" from the menu bar and add the vendor transports to the import queue one at a time in the order given above.



Requests for E4P: 0 / 2416

Number	Request	Clt	RC	Owner
2414	NSQK905250	100	◆	DATAVARD
2415	NSQK905311	100	◆	DATAVARD
2416	ED2K901038	100	◆	DATAVARD

5. Click on the first transport and hit F9 to select it. Then click the "*Import Request*" icon (small truck) on the toolbar:



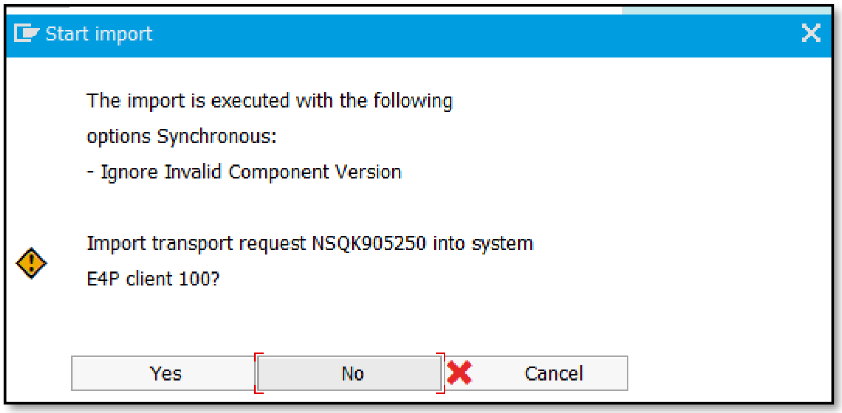
Import Queue: System E4P

Requests for E4P: 0 / 2416

Number	Request	Clt	RC	Owner	Project	Short Text
2414	NSQK905250	100	◆	DATAVARD		
2415	NSQK905311	100	◆	DATAVARD		
2416	ED2K901038	100	◆	DATAVARD		

6. Select the "*Execution*" tab in the "*Import Transport Request*" pop-up dialog box and select the option "*Synchronous*". Then select the "*Options*" tab and put a checkmark next to "*Ignore Invalid Component Version*". Finally, click the green checkmark icon in the lower right-hand corner of the dialog box.

7. In the next pop-up dialog box click "Yes" to start the import process:



The import is executed with the following options Synchronous:

- Ignore Invalid Component Version

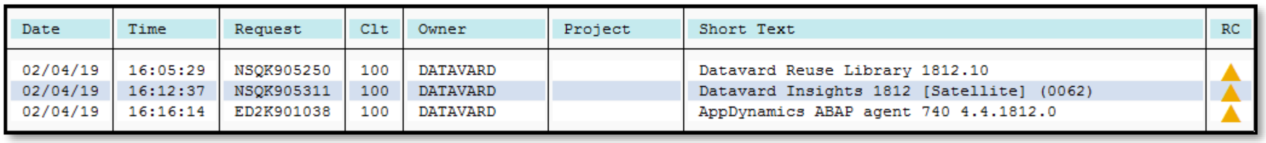
Import transport request NSQK905250 into system E4P client 100?

8. The transport will take about 4 or 5 minutes to complete. It will end with warnings, which is okay. When it is done, click the transport in the queue again and press F9 to unselect it.

9. Repeat 5 through 8 for the second transport. The second transport will also take about 4 or 5 minutes and will also end with warnings.

10. Repeat 5 through 8 for the third transport. The third transport will take about 2 minutes to complete and will also end with warnings.

11. Check the import history for the correct statuses:



Date	Time	Request	Clt	Owner	Project	Short Text	RC
02/04/19	16:05:29	NSQK905250	100	DATAVARD		Datavard Reuse Library 1812.10	⚠
02/04/19	16:12:37	NSQK905311	100	DATAVARD		Datavard Insights 1812 [Satellite] (0062)	⚠
02/04/19	16:16:14	ED2K901038	100	DATAVARD		AppDynamics ABAP agent 740 4.4.1812.0	⚠

Configure User Authorizations

There are three roles that are imported along with the three request transports. Their descriptions are below:

/DVD/APPD_ADMIN – Administration role for Datavard Insights

- Run Insights transactions
- Allow to start/stop collector jobs
- Allow to add/remove new system to/from monitoring pool
- Allow to view monitoring and analyses results
- Allow to maintain Insights customizing

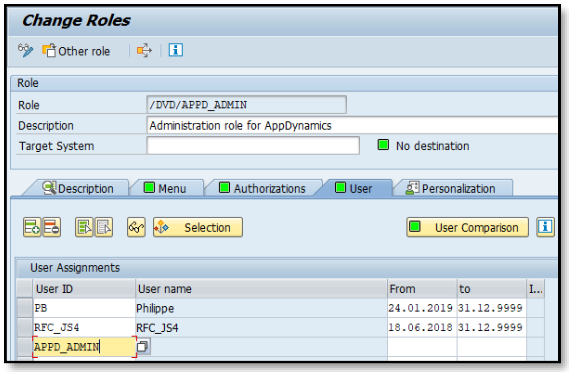
/DVD/APPD_USER – User role for AppDynamics (subset of the /DVD/APPD_ADMIN role)

- Allow execution of AppDynamics integration function modules
- Allow ABAP and SQL action execution

/DVD/MON_ADMIN – Administration role for monitoring

- Allow starting the Insights monitoring tool

Follow the steps below to configure users for the appropriate roles:



Role

Role	/DVD/APPD_ADMIN
Description	Administration role for AppDynamics
Target System	🟩 No destination

User Assignments

User ID	User name	From	to	I...
PB	Philippe	24.01.2019	31.12.9999	
RFC_JS4	RFC_JS4	18.06.2018	31.12.9999	
APPD_ADMIN				

1. Start transaction **PFCG**

2. Edit role /DVD/APPD_ADMIN

3. Add AppDynamics admin users to **/DVD/APPD_ADMIN**

4. Edit role /DVD/APPD_USER

5. Add all monitored users to **/DVD/APPD_USER**

6. Edit role /DVD/MON_ADMIN

7. Add the user to the **/DVD/MON_ADMIN** role to authorize them to start the Datavard Insights monitoring tool

Configure ABAP Agent and Install HTTP SDK

The ABAP agent will be installed on all ABAP servers. The AppDynamics admin roles must first be installed in the ABAP system and assigned to your user account before you can execute the following steps.

1. Login to the SAP system and execute transaction "*/DVD/APPD_CUST*".

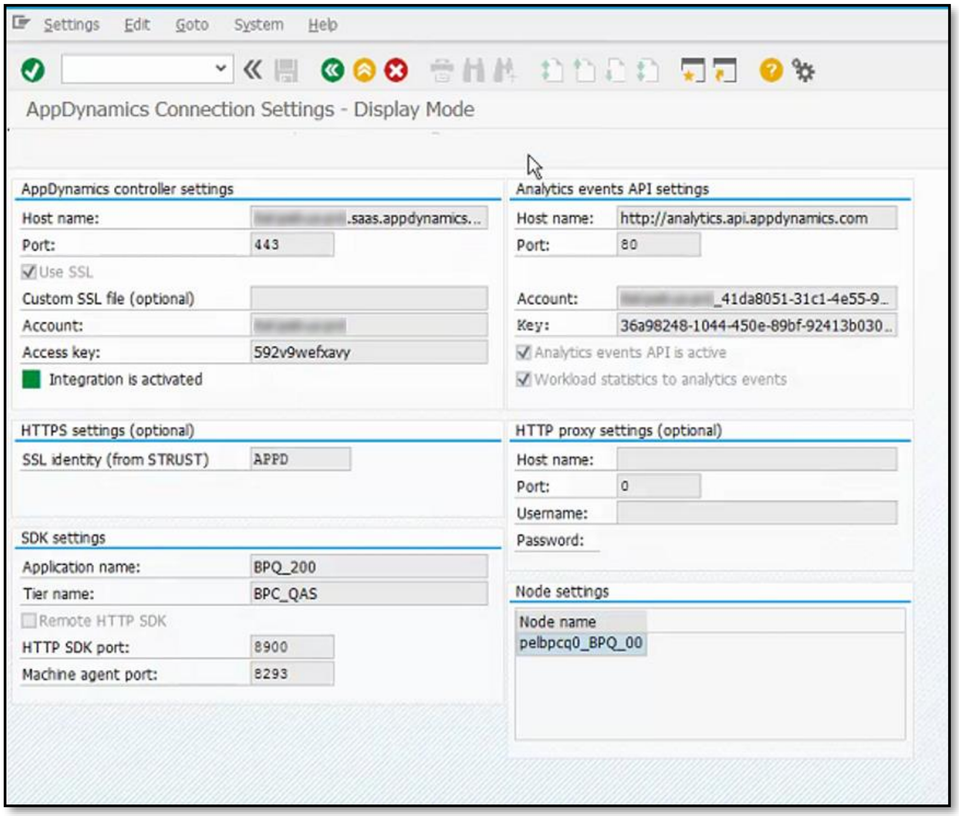
2. Switch to edit mode.

3. Fill in the property fields on the screen.

Property Name	Actual Property Value	Example Property Value	Additional Info
AppDynamics Controller Settings			
Host name:	Refer to "*AppDynamics Controller and Agent Properties*" section above "**controller-host**"	mycompany.saas.appdynamics.com	

Property Name	Actual Property Value	Example Property Value	Additional Info
Port:	Refer to “*AppDynamics Controller and Agent Properties*” section above “**controller-port**”	443	
Use SSL:	Refer to “*AppDynamics Controller and Agent Properties*” section above “**controller-ssl-enabled**”	Checked = true	
Account:	Refer to “*AppDynamics Controller and Agent Properties*” section above “**account-name**”	mycompany	
Access key:	Refer to “*AppDynamics Controller and Agent Properties*” section above “**account-access-key**”	9ayhsaoe096t	
HTTP Settings (optional)			
SSL Identity (from STRUST)	See (SSL Certificates) for details on SSL identity	APPD	
SDK Settings			
Application name:	This value starts with the name of the application followed by a hyphen and the environment name	MyApp-QA	
Tier name:	This value will start with the SAP system ID followed by a hyphen and the network zone (for example, for the X6P system which resides in Zone C, the name would be “X6P-ZoneC” without any spaces	X6P-ZoneC	
Remote HTTP SDK:	Unchecked = false	Unchecked = false	If you are installing the HTTP SDK manually (for unsupported OS), please see (HTTP SDK Manual Install)
HTTP SDK Port:	8900	8900	
Machine Agent Port:	8293	8293	

Property Name	Actual Property Value	Example Property Value	Additional Info
Analytics Events API Settings			
Host Name:	Refer to “*AppDynamics Controller and Agent Properties*” section above “**analytics-events-service-host**”	https:// analytics.api.appdynamics.com	
Port:	Refer to “*AppDynamics Controller and Agent Properties*” section above “**analytics-events-service-port**”	443	
Account:	Refer to “*AppDynamics Controller and Agent Properties*” section above “**global-account-name**”	mycompany_a88c08ac-78c4-4eea-8bbc-2ea492c04489	
Key:	See (Creating API Keys) for details on how to create an Analytics API Key with “Custom Analytics Events Permissions”	b54c03fd-54c2-7aaw-3ccv-8cn738c06731	
Analytics events API is active:	Checked = true	Checked = true	
Workload statistics to analytics events:	Checked = true	Checked = true	
HTTP Proxy Settings (optional) Use the proxy properties to configure the agent to connect to the Controller through a proxy			
Host Name:	The proxy host name or IP address	10.31.48.132	
Port:	The proxy HTTP(S) port	8183	
Username:	The name of the user that is authenticated by the proxy host	proxyusr	
Password:	The password of the user that is authenticated by the proxy host	proxypwd	
Node Settings			
Node Name:	You should see a complete list of all the app servers for this SAP system		



AppDynamics Connection Settings - Display Mode

AppDynamics controller settings

Host name: [redacted].saas.appdynamics...
Port: 443
☑ Use SSL
Custom SSL file (optional):
Account: [redacted]
Access key: 592v9wefxavy
🟩 Integration is activated

Analytics events API settings

Host name: http://analytics.api.appdynamics.com
Port: 80
Account: [redacted]_41da8051-31c1-4e55-9...
Key: 36a98248-1044-450e-89bf-92413b030...
☑ Analytics events API is active
☑ Workload statistics to analytics events

HTTPS settings (optional)

SSL identity (from STRUST) APPD

HTTP proxy settings (optional)

Host name:
Port: 0
Username:
Password:

SDK settings

Application name: BPQ_200
Tier name: BPC_QAS
□ Remote HTTP SDK
HTTP SDK port: 8900
Machine agent port: 8293

Node settings

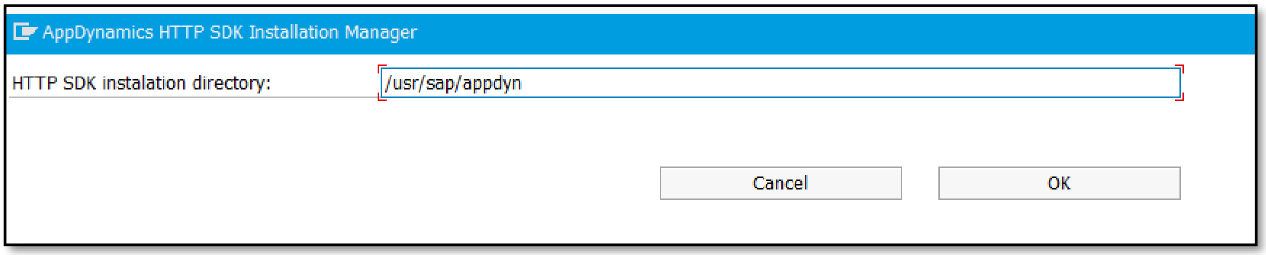
Node name
pelbpcq0_BPQ_00

4. Click the “*Activate integration*” button on the toolbar

5. Click the “*SDK Installation*” button on the toolbar. This will take you to the “*AppDynamics HTTP SDK Installation Manager*” screen

6. Select “*Edit > Change Directory*” from the menu bar

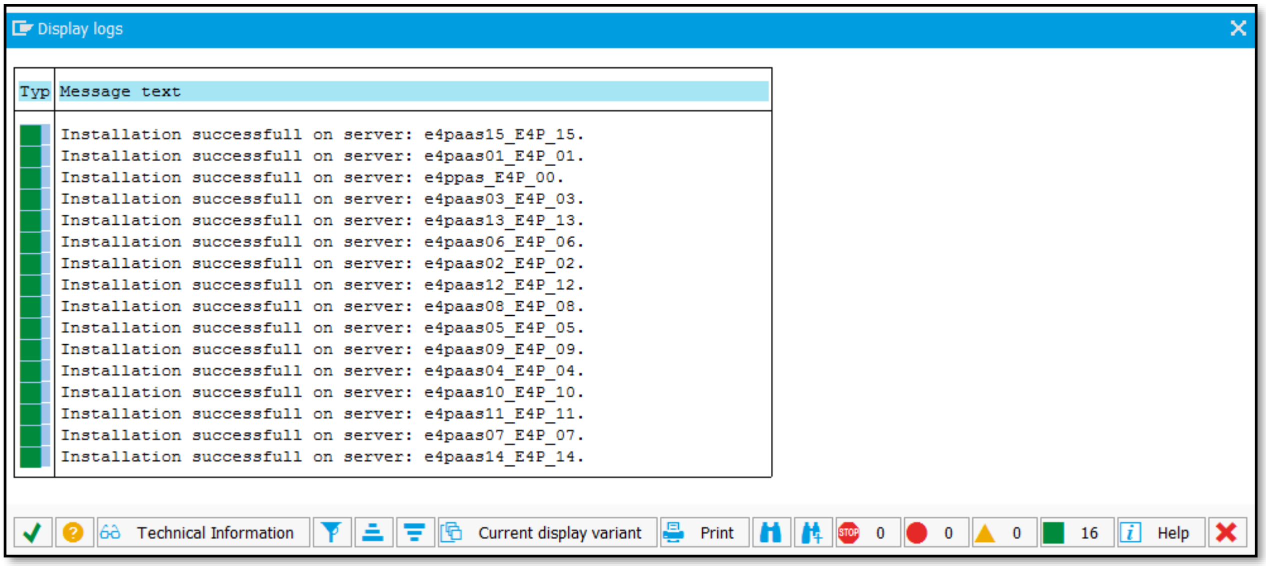
7. Enter “*/usr/sap/appdyn*” in the field displayed in the pop-up dialog box shown below, and then click OK:



AppDynamics HTTP SDK Installation Manager

HTTP SDK instalation directory: /usr/sap/appdyn

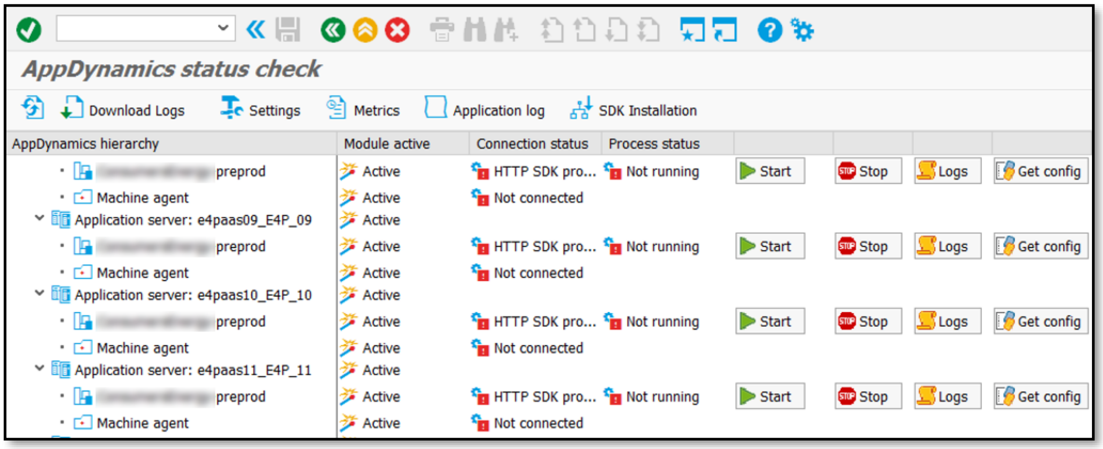
8. Click the “Install SDK” button on the toolbar. You should get a status screen as shown below:



```
Typ Message text

    Installation successfull on server: e4paas15_E4P_15.
    Installation successfull on server: e4paas01_E4P_01.
    Installation successfull on server: e4ppas_E4P_00.
    Installation successfull on server: e4paas03_E4P_03.
    Installation successfull on server: e4paas13_E4P_13.
    Installation successfull on server: e4paas06_E4P_06.
    Installation successfull on server: e4paas02_E4P_02.
    Installation successfull on server: e4paas12_E4P_12.
    Installation successfull on server: e4paas08_E4P_08.
    Installation successfull on server: e4paas05_E4P_05.
    Installation successfull on server: e4paas09_E4P_09.
    Installation successfull on server: e4paas04_E4P_04.
    Installation successfull on server: e4paas10_E4P_10.
    Installation successfull on server: e4paas11_E4P_11.
    Installation successfull on server: e4paas07_E4P_07.
    Installation successfull on server: e4paas14_E4P_14.
```

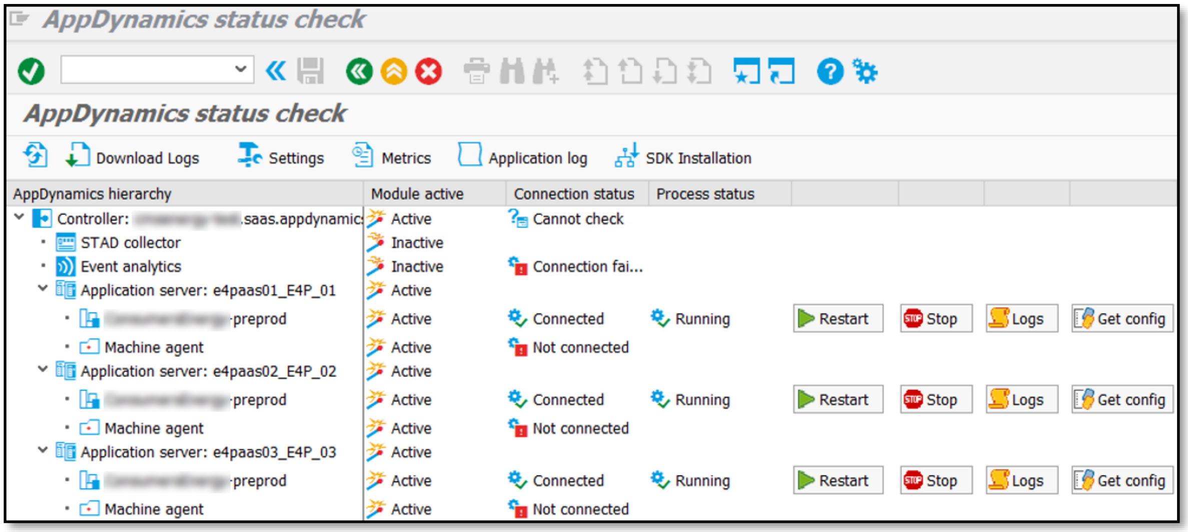
9. Click the green checkmark to exit the screen and return to the “*AppDynamics settings*” screen

10. Click the “*Status*” button on the toolbar. This will take you to the “*AppDynamics status check*” screen. As you can see below, all the application servers are showing a status of “*Not running*” for the HTTP SDK proxy:



AppDynamics hierarchy	Module active	Connection status	Process status				
• [redacted] preprod	Active	HTTP SDK pro...	Not running	Start	Stop	Logs	Get config
• Machine agent	Active	Not connected					
Application server: e4paas09_E4P_09	Active						
• [redacted] preprod	Active	HTTP SDK pro...	Not running	Start	Stop	Logs	Get config
• Machine agent	Active	Not connected					
Application server: e4paas10_E4P_10	Active						
• [redacted] preprod	Active	HTTP SDK pro...	Not running	Start	Stop	Logs	Get config
• Machine agent	Active	Not connected					
Application server: e4paas11_E4P_11	Active						
• [redacted] preprod	Active	HTTP SDK pro...	Not running	Start	Stop	Logs	Get config
• Machine agent	Active	Not connected					

11. Click each “*Start*” button to start the HTTP SDK proxy on each SAP app server:



AppDynamics status check

AppDynamics hierarchy	Module active	Connection status	Process status				
Controller: [redacted].saas.appdynamic	Active	Cannot check					
STAD collector	Inactive						
Event analytics	Inactive	Connection fai...					
Application server: e4paas01_E4P_01	Active						
[redacted]-preprod	Active	Connected	Running	Restart	Stop	Logs	Get config
Machine agent	Active	Not connected					
Application server: e4paas02_E4P_02	Active						
[redacted]-preprod	Active	Connected	Running	Restart	Stop	Logs	Get config
Machine agent	Active	Not connected					
Application server: e4paas03_E4P_03	Active						
[redacted]-preprod	Active	Connected	Running	Restart	Stop	Logs	Get config
Machine agent	Active	Not connected					

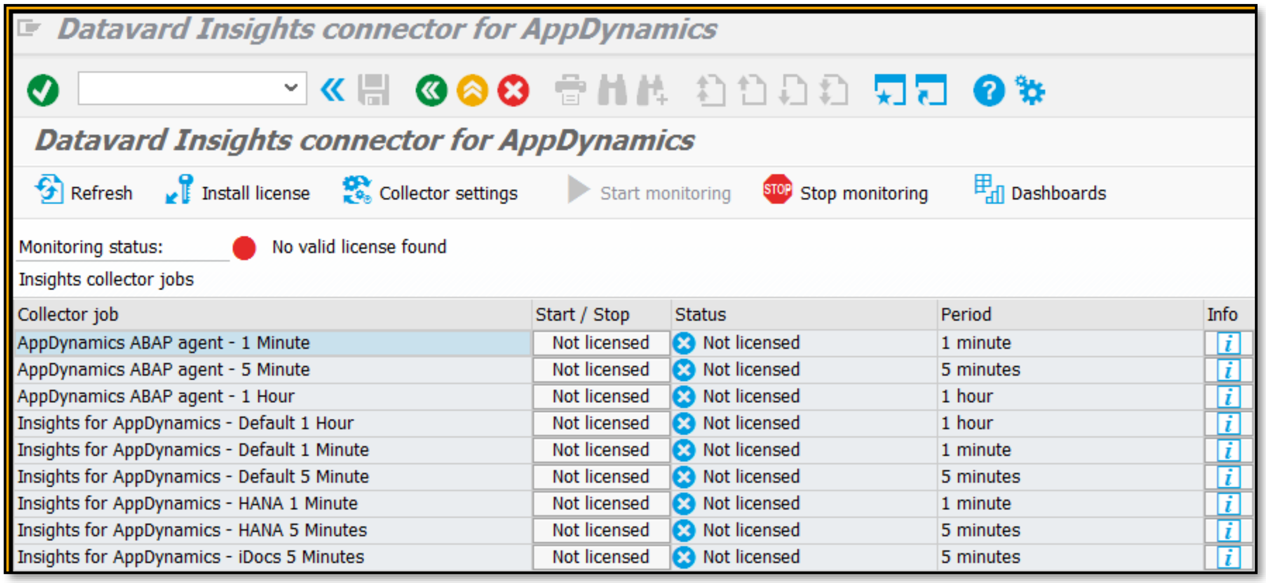
12. When the HTTP SDK is installed in this way, the directory structure that gets created on each app server has read/write permissions at the user, group, and other levels. We want to remove write privileges for other. Login to each app server as the appropriate <SID>adm user and change the permissions on the HTTP SDK directory and contents:

```
cd

. ./.profile

cd /usr/sap/appdyn

chmod -R o-w ./appdhttpsdk
```

Activate Datavard Insight Collectors

Datavard Insights can collect detailed performance data for an SAP system. It uses collector jobs that run as periodic background processes in the SAP system. These jobs must be scheduled to run.

1. Go to the "*/DVD/APPD_DI_CON*" transaction. This brings up the screen seen below.



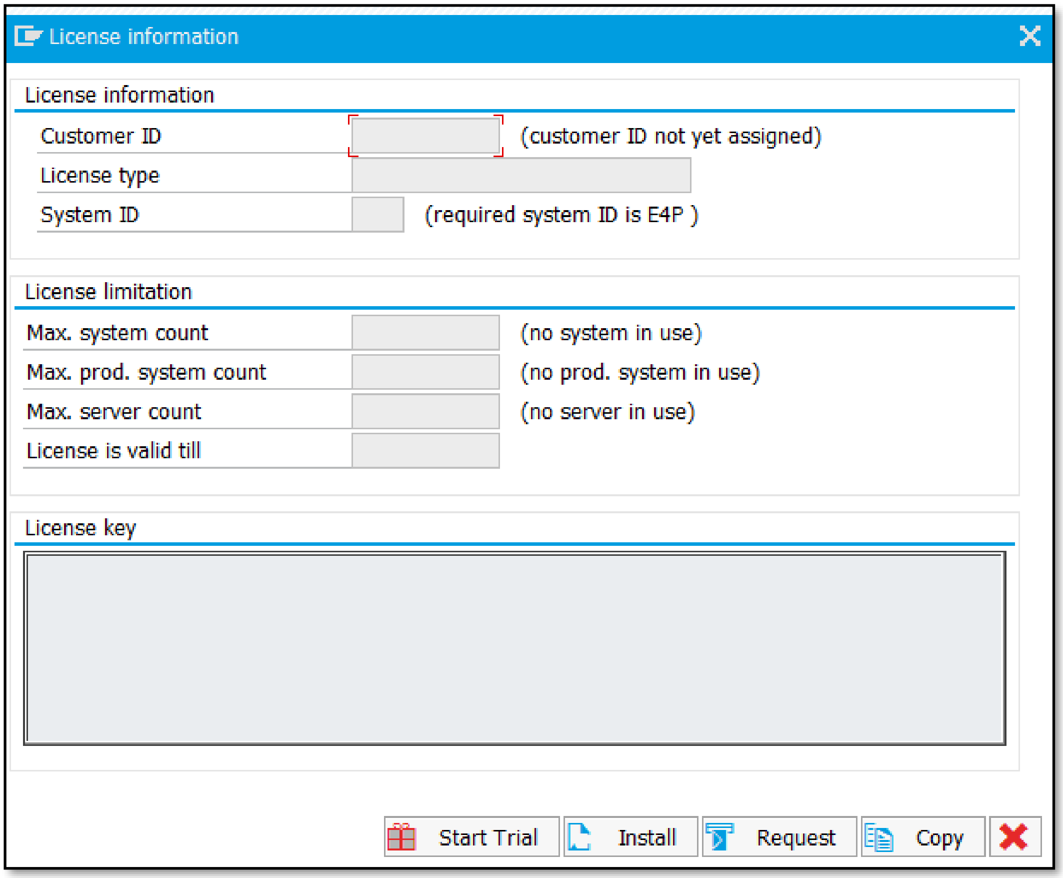
Datavard Insights connector for AppDynamics

Monitoring status: ● No valid license found

Insights collector jobs

Collector job	Start / Stop	Status	Period	Info
AppDynamics ABAP agent - 1 Minute	Not licensed	⊗ Not licensed	1 minute	i
AppDynamics ABAP agent - 5 Minute	Not licensed	⊗ Not licensed	5 minutes	i
AppDynamics ABAP agent - 1 Hour	Not licensed	⊗ Not licensed	1 hour	i
Insights for AppDynamics - Default 1 Hour	Not licensed	⊗ Not licensed	1 hour	i
Insights for AppDynamics - Default 1 Minute	Not licensed	⊗ Not licensed	1 minute	i
Insights for AppDynamics - Default 5 Minute	Not licensed	⊗ Not licensed	5 minutes	i
Insights for AppDynamics - HANA 1 Minute	Not licensed	⊗ Not licensed	1 minute	i
Insights for AppDynamics - HANA 5 Minutes	Not licensed	⊗ Not licensed	5 minutes	i
Insights for AppDynamics - iDocs 5 Minutes	Not licensed	⊗ Not licensed	5 minutes	i

2. Currently there is no license. Click the "*Install license*" button on the toolbar. You will get a pop-up saying the license isn't found. Click "*OK*" and a new pop-up box appears:



License information

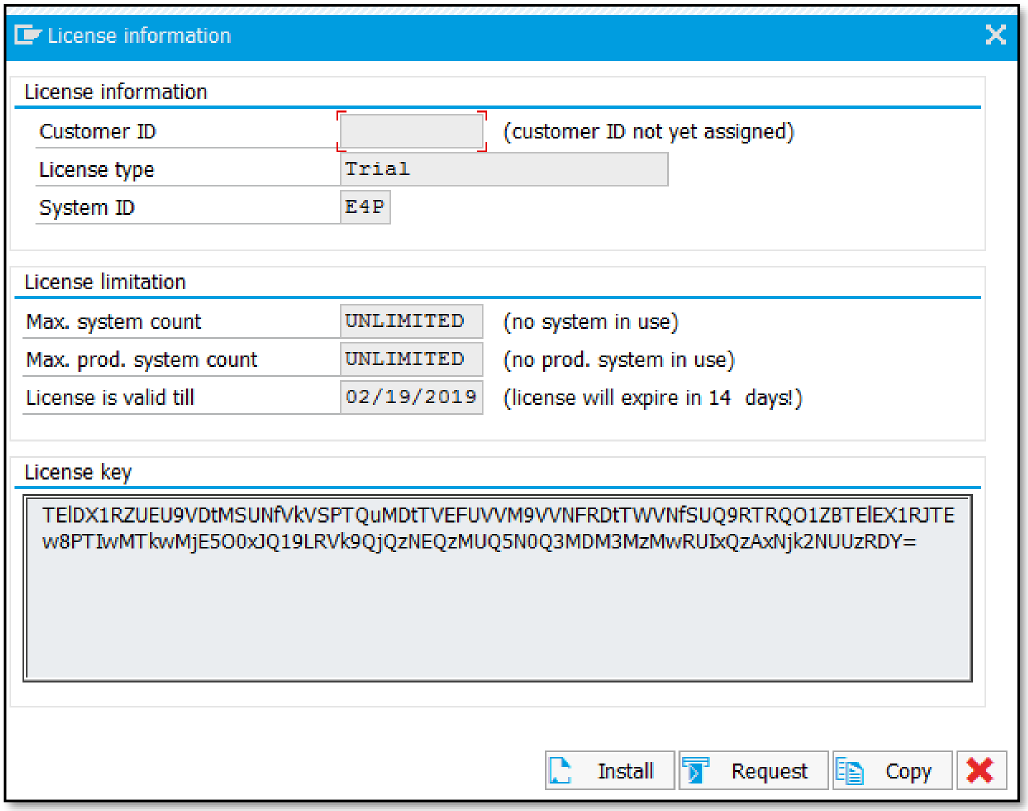
Customer ID		(customer ID not yet assigned)
License type		
System ID		(required system ID is E4P)

License limitation

Max. system count		(no system in use)
Max. prod. system count		(no prod. system in use)
Max. server count		(no server in use)
License is valid till		

License key

3. Click the "*Start Trial*" button. The following fields will get filled in automatically:



License information

License information

Customer ID		(customer ID not yet assigned)
License type	Trial	
System ID	E4P	

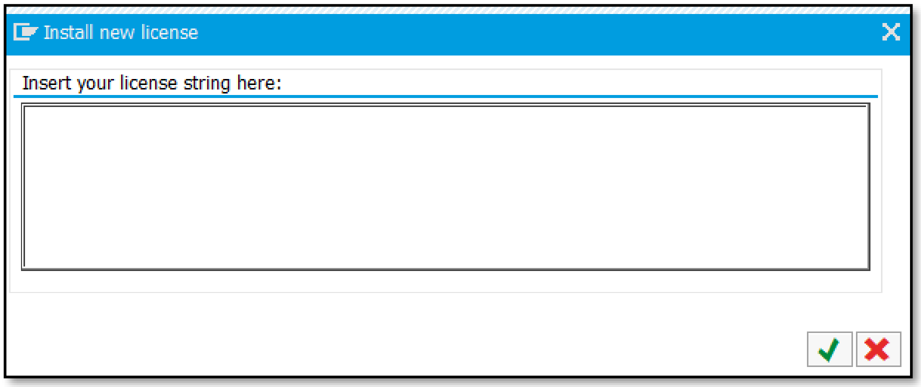
License limitation

Max. system count	UNLIMITED	(no system in use)
Max. prod. system count	UNLIMITED	(no prod. system in use)
License is valid till	02/19/2019	(license will expire in 14 days!)

License key

TElDX1RZUEU9VDtMSUNfVkVSPTQuMDtTVEFUVVM9VVNFRDtTWVNfSUQ9RTRQO1ZBTElEX1RJTE
w8PTIwMTkwMjE5O0xJQ19LRVk9QjQzNEQzMUQ5N0Q3MDM3MzMwRUIxQzAxNjk2NUUzRDY=

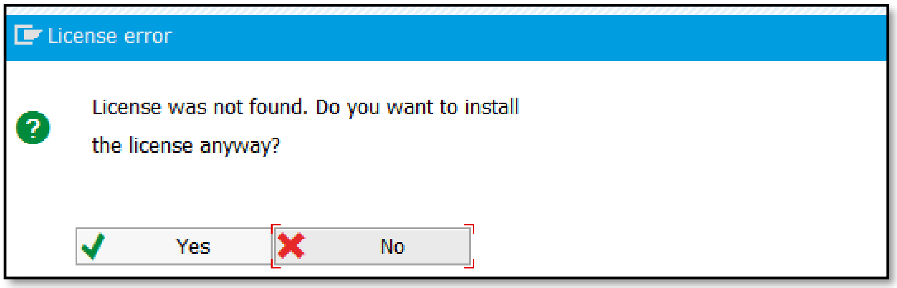
4. Copy the contents of the “*License key*” field to the clipboard. Click the “*Install*” button. You will get another pop-up:



Install new license

Insert your license string here:

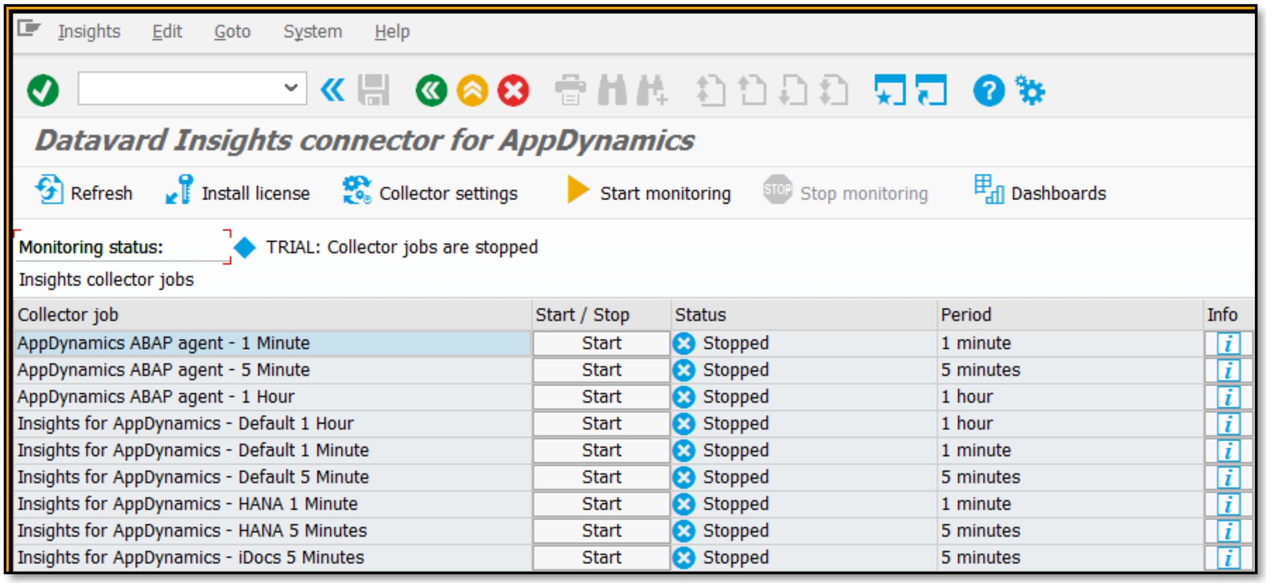
5. Paste the key that was copied from the previous screen into the empty box. Then click the green checkmark. Another pop-up appears:



License error

License was not found. Do you want to install the license anyway?

6. Click "*Yes*" to install the license. You will then need to click the red "X" to close the "*License Information*" screen. The temporary license now takes effect and will be valid for two weeks:



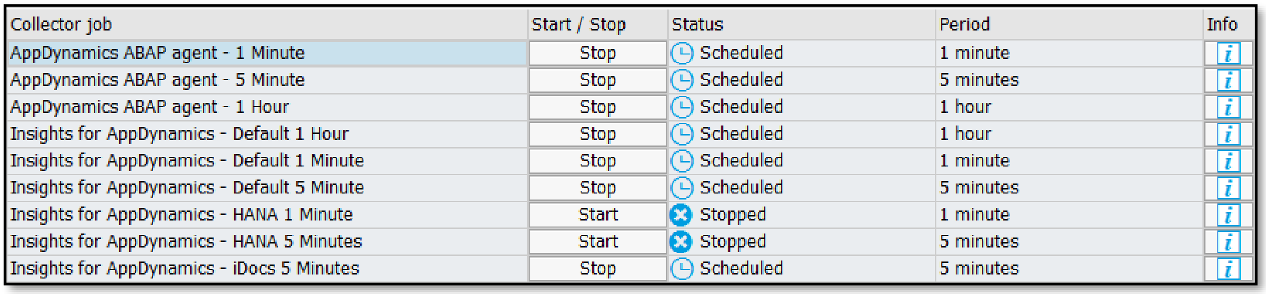
Datavard Insights connector for AppDynamics

Monitoring status: TRIAL: Collector jobs are stopped

Insights collector jobs

Collector job	Start / Stop	Status	Period	Info
AppDynamics ABAP agent - 1 Minute	Start	Stopped	1 minute	i
AppDynamics ABAP agent - 5 Minute	Start	Stopped	5 minutes	i
AppDynamics ABAP agent - 1 Hour	Start	Stopped	1 hour	i
Insights for AppDynamics - Default 1 Hour	Start	Stopped	1 hour	i
Insights for AppDynamics - Default 1 Minute	Start	Stopped	1 minute	i
Insights for AppDynamics - Default 5 Minute	Start	Stopped	5 minutes	i
Insights for AppDynamics - HANA 1 Minute	Start	Stopped	1 minute	i
Insights for AppDynamics - HANA 5 Minutes	Start	Stopped	5 minutes	i
Insights for AppDynamics - iDocs 5 Minutes	Start	Stopped	5 minutes	i

7. Click the "*Start monitoring*" button

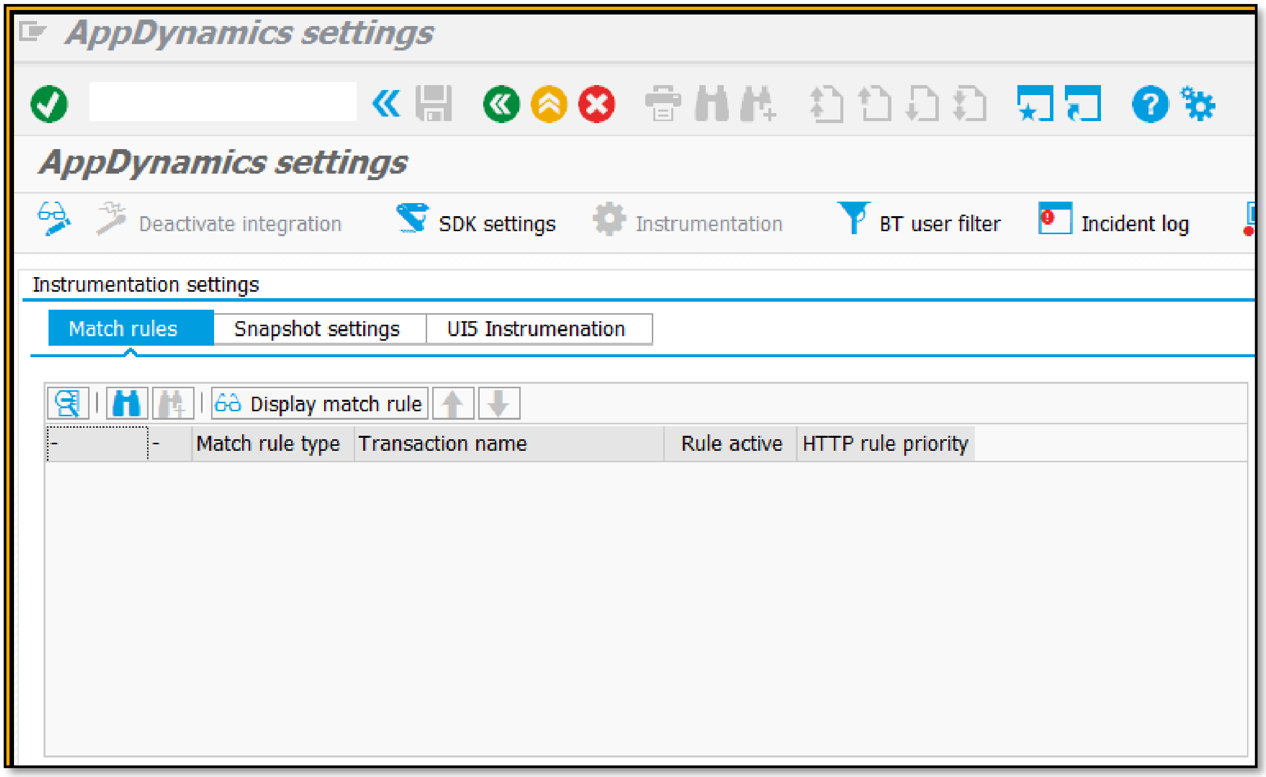


Collector job	Start / Stop	Status	Period	Info
AppDynamics ABAP agent - 1 Minute	Stop	Scheduled	1 minute	i
AppDynamics ABAP agent - 5 Minute	Stop	Scheduled	5 minutes	i
AppDynamics ABAP agent - 1 Hour	Stop	Scheduled	1 hour	i
Insights for AppDynamics - Default 1 Hour	Stop	Scheduled	1 hour	i
Insights for AppDynamics - Default 1 Minute	Stop	Scheduled	1 minute	i
Insights for AppDynamics - Default 5 Minute	Stop	Scheduled	5 minutes	i
Insights for AppDynamics - HANA 1 Minute	Start	Stopped	1 minute	i
Insights for AppDynamics - HANA 5 Minutes	Start	Stopped	5 minutes	i
Insights for AppDynamics - iDocs 5 Minutes	Stop	Scheduled	5 minutes	i

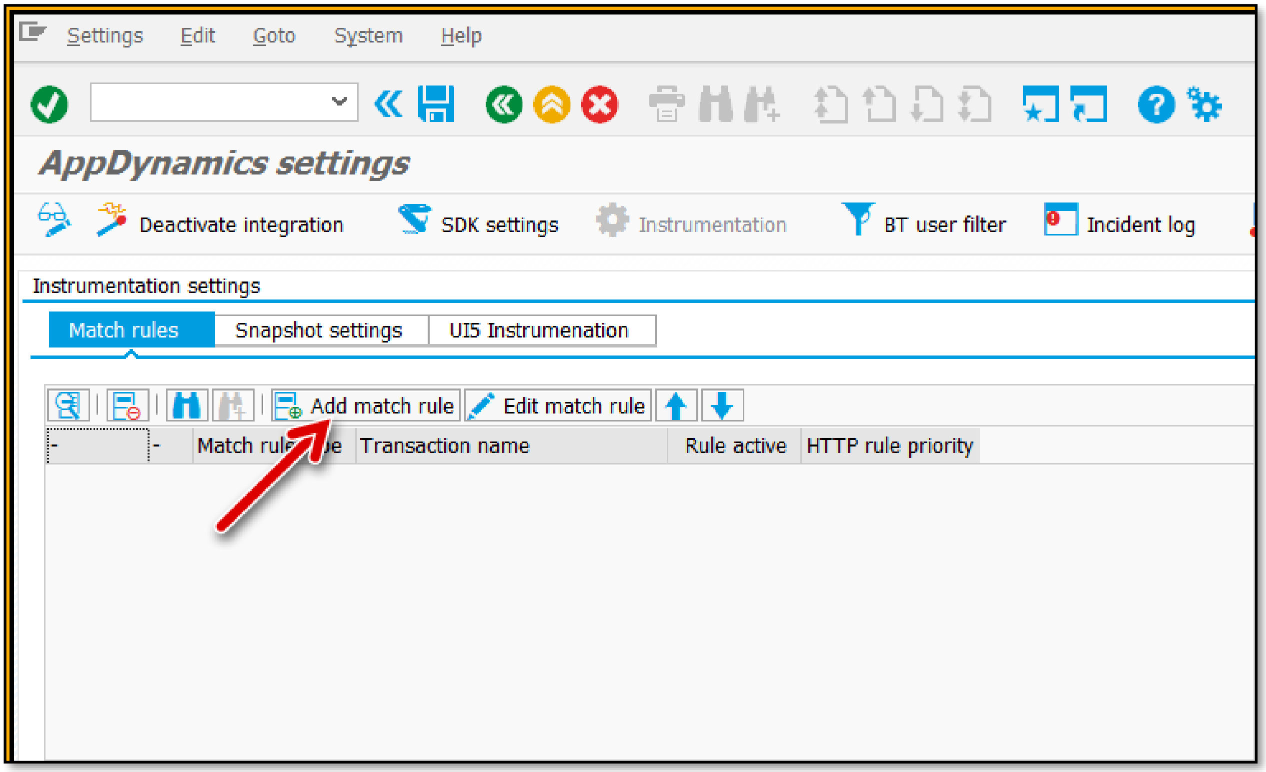
Configure Generic Instrumentation

Now we are ready to start configuring the instrumentation settings. Until this is done you will not see any transactional data displayed in the AppDynamics controller.

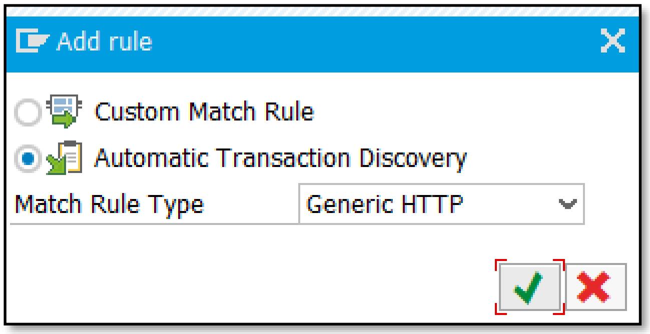
1. Go to transaction "*/DVD/APPD_CUST*" and click the "*Instrumentation*" button on the toolbar



2. Switch to edit mode and click the "*Add match rule*" button:



3. Select "*Automatic Transaction Discovery*" and select "*Generic HTTP*" from the drop-down. (Automatic transaction discovery will be used for all the match roles we are going to set up below.)



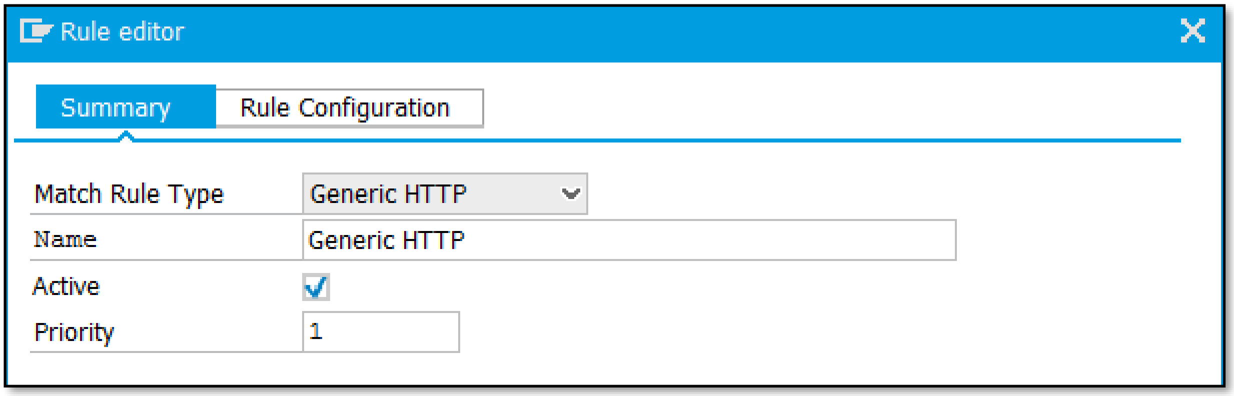
Add rule

○ Custom Match Rule

● Automatic Transaction Discovery

Match Rule Type: Generic HTTP

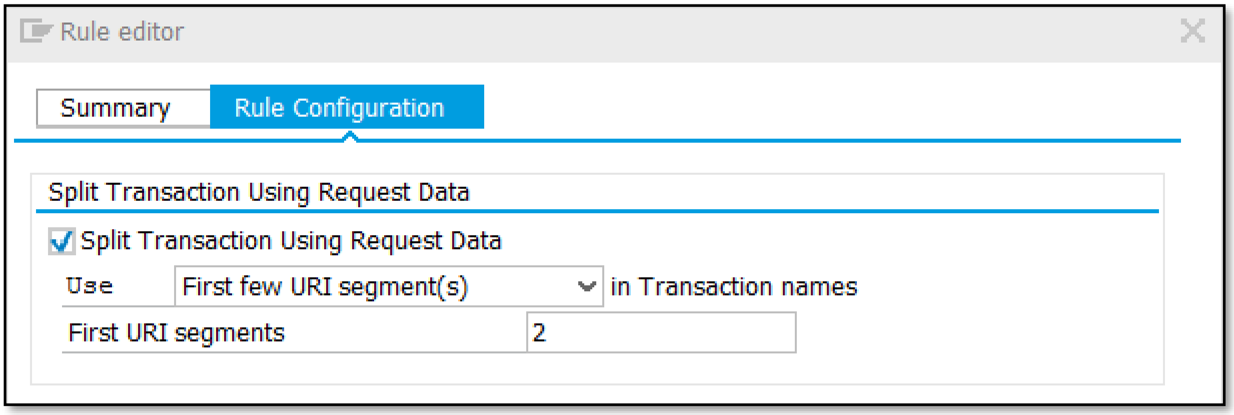
4. Click the green checkmark and then fill in the following details on the next pop-up screen:



Rule editor

Match Rule Type	Generic HTTP
Name	Generic HTTP
Active	☑
Priority	1

5. On the "*Rule Configuration*" tab accept the default settings. Click the green checkmark button to exit the screen:



Rule editor

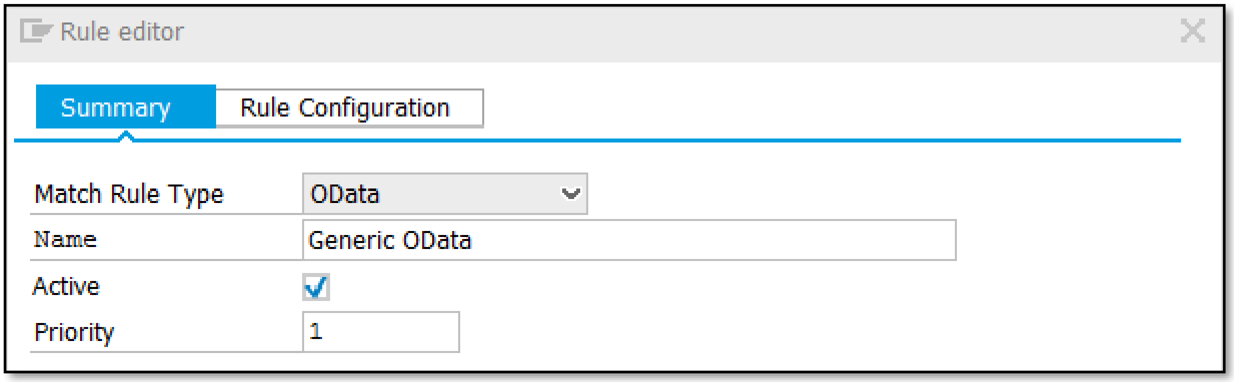
Split Transaction Using Request Data

☑ Split Transaction Using Request Data

Use First few URI segment(s) in Transaction names

First URI segments: 2

6. Click the "*Add match rule*" button again and this time select "*OData*" from the drop-down list. Fill in the fields as shown below:



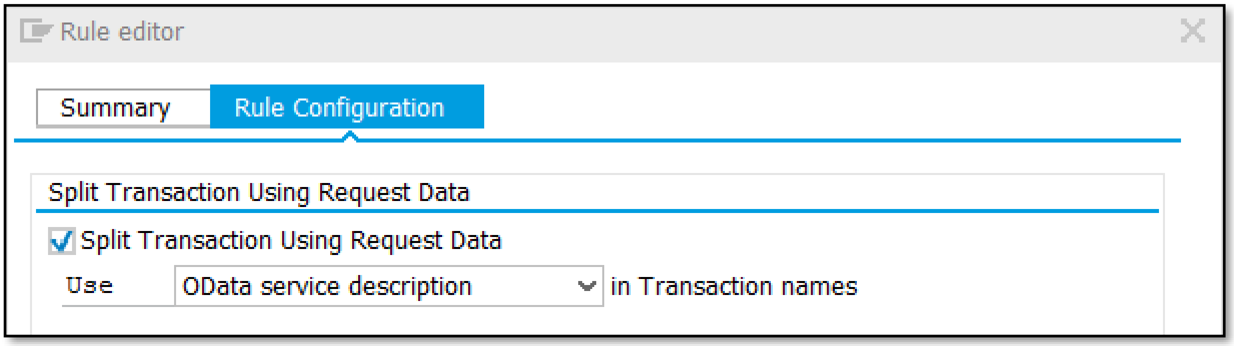
Match Rule Type: OData

Name: Generic OData

Active: ☑

Priority: 1

7. On the "*Rule Configuration*" tab accept the default settings. Click the green checkmark button to exit the screen:

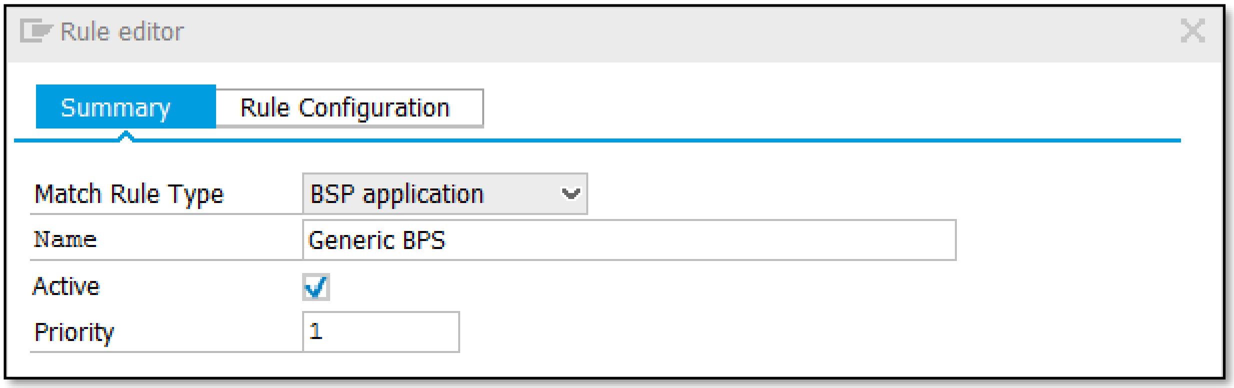


Split Transaction Using Request Data

☑ Split Transaction Using Request Data

Use OData service description in Transaction names

8. Click the "*Add match rule*" button again and this time select "*BSP application*" from the drop-down list. Fill in the fields as shown below:



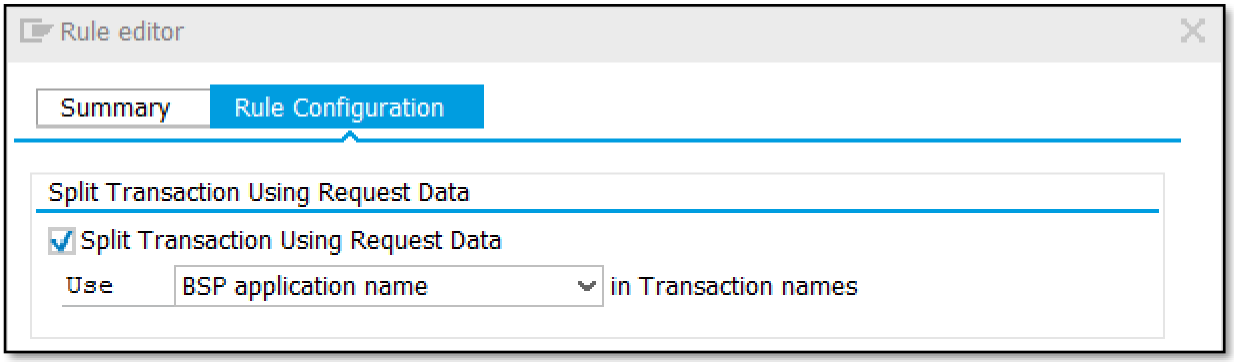
Match Rule Type: BSP application

Name: Generic BPS

Active: ☑

Priority: 1

9. On the "*Rule Configuration*" tab accept the default settings. Click the green checkmark button to exit the screen:

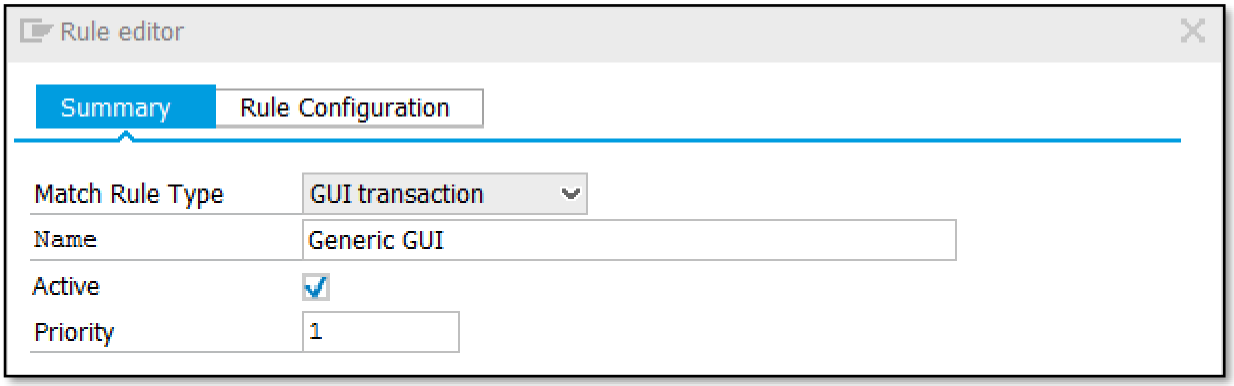


Split Transaction Using Request Data

☑ Split Transaction Using Request Data

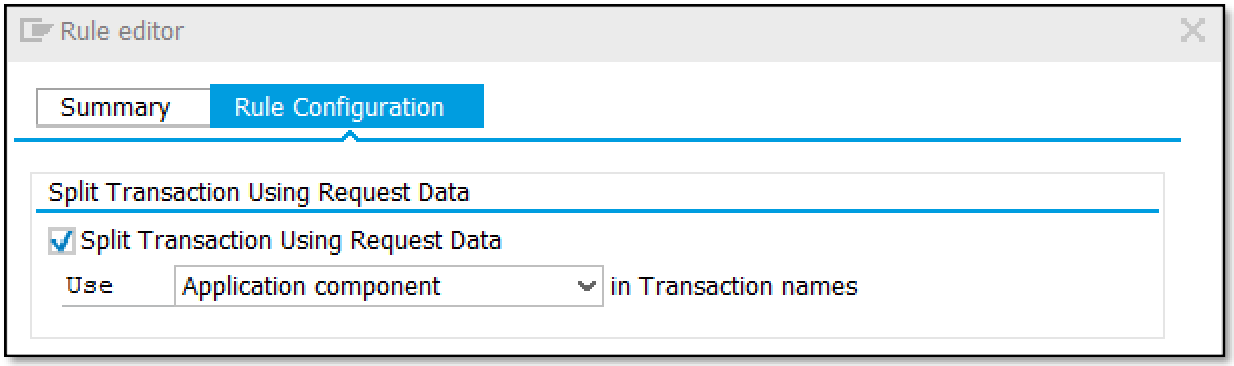
Use BSP application name in Transaction names

10. Click the "*Add match rule*" button again and this time select "*GUI transaction*" from the drop-down list. Fill in the fields as shown below:



Match Rule Type	GUI transaction
Name	Generic GUI
Active	☑
Priority	1

11. On the "*Rule Configuration*" tab accept the default settings. Click the green checkmark button to exit the screen:

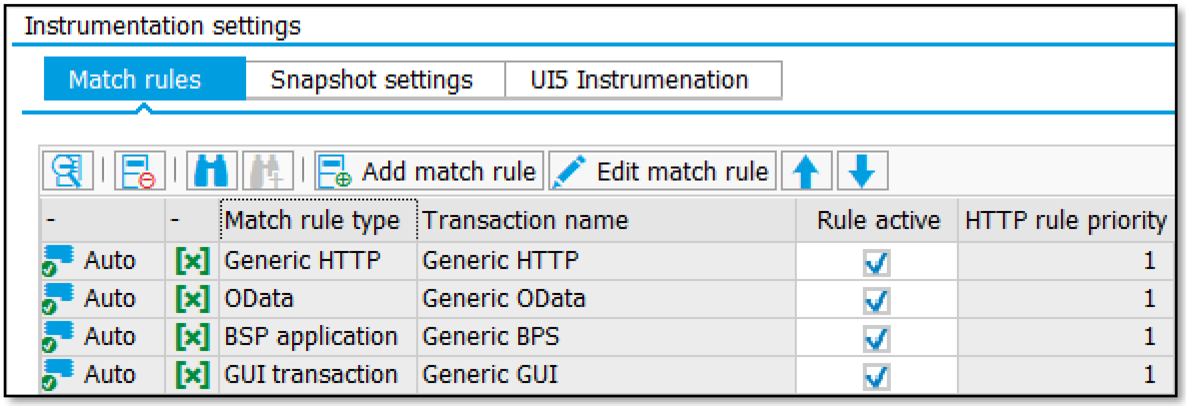


Split Transaction Using Request Data

☑ Split Transaction Using Request Data

Use Application component in Transaction names

12. When you are done, the "Instrumentation settings" screen should look like this:



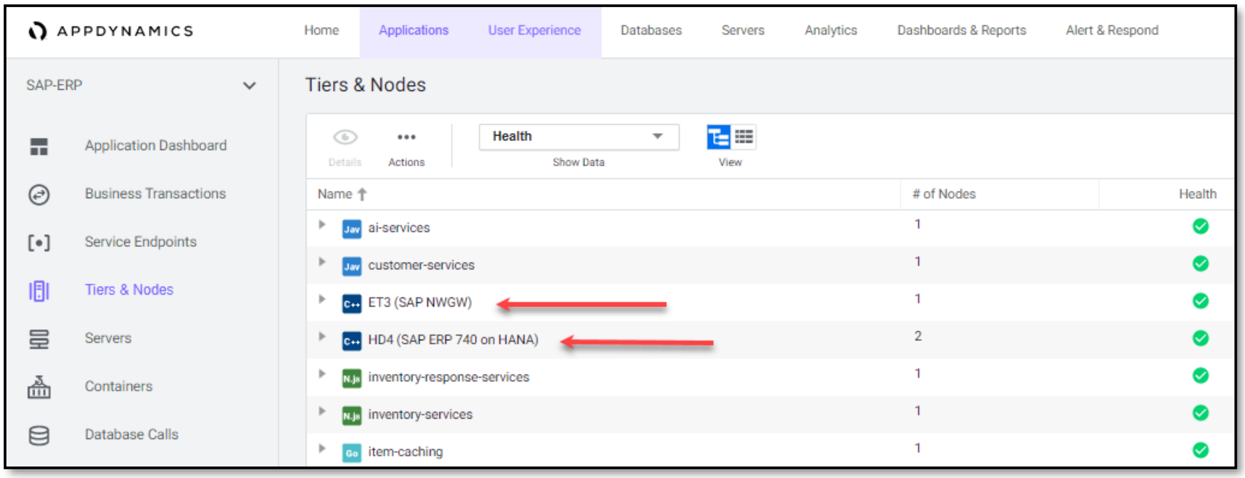
-	-	Match rule type	Transaction name	Rule active	HTTP rule priority
Auto	[x]	Generic HTTP	Generic HTTP	✓	1
Auto	[x]	OData	Generic OData	✓	1
Auto	[x]	BSP application	Generic BPS	✓	1
Auto	[x]	GUI transaction	Generic GUI	✓	1

13. Be sure to click the "*Save*" icon on the top toolbar after adding the match rules. Otherwise your changes will be lost when you exit the transaction.

Validate the ABAP Agent Installation

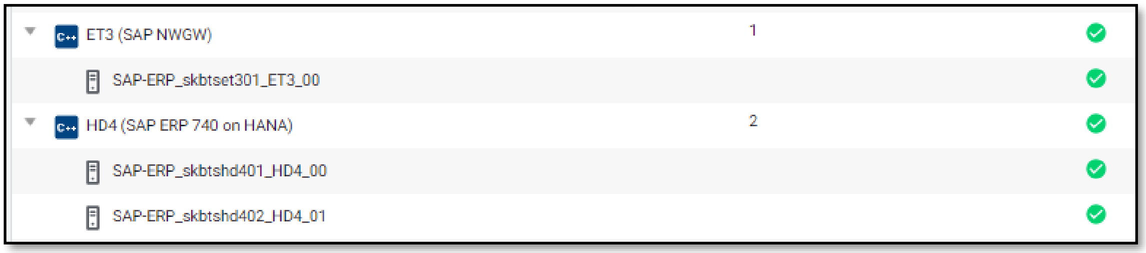
Now we can validate the changes we made for the ABAP agent in the AppDynamics controller.

1. Login to the AppDynamics controller and select the "*Applications*" tab. Then select "*Tiers & Nodes*" from the navigation pane on the left. You should see the SAP applications in the list:



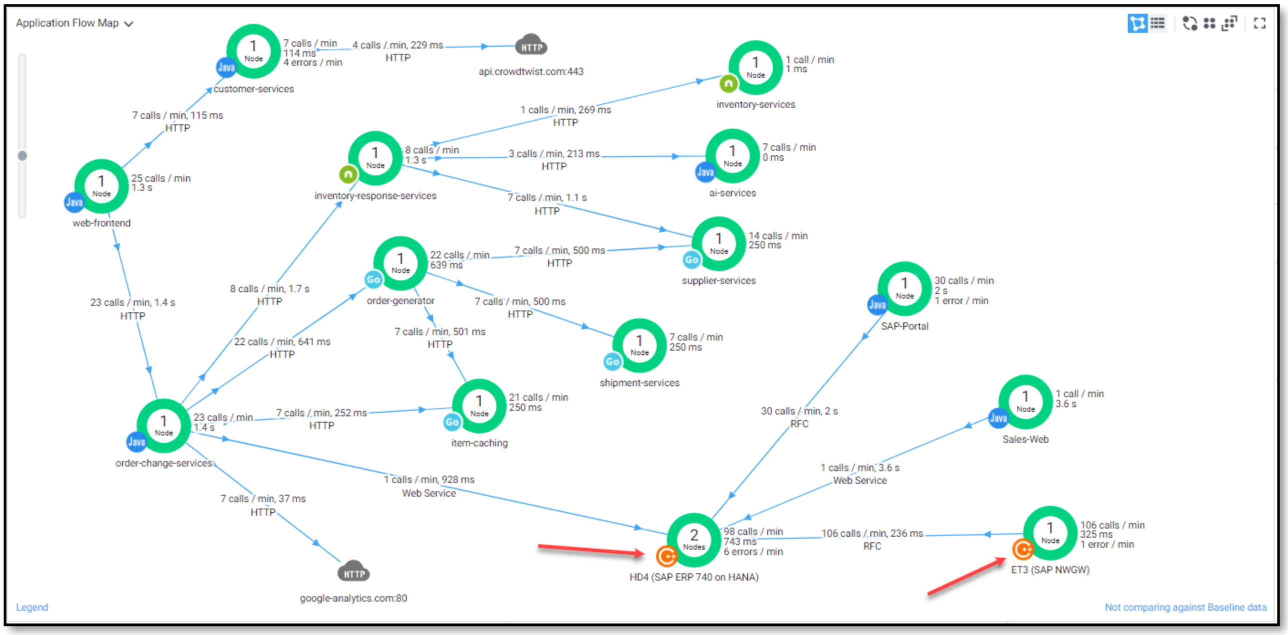
Name	# of Nodes	Health
ai-services	1	✓
customer-services	1	✓
ET3 (SAP NWGW)	1	✓
HD4 (SAP ERP 740 on HANA)	2	✓
inventory-response-services	1	✓
inventory-services	1	✓
item-caching	1	✓

2. If you expand the node, you should see all the SAP app servers listed with a green status showing in the "Health" column:



ET3 (SAP NWGW)	1	✓
SAP-ERP_skbtset301_ET3_00		✓
HD4 (SAP ERP 740 on HANA)	2	✓
SAP-ERP_skbtshd401_HD4_00		✓
SAP-ERP_skbtshd402_HD4_01		✓

3. After some transaction activity takes place in the system, you should also be able to see the system node in the flow map on the "*Application Dashboard*" screen:



Install Machine / Server Visibility Agent

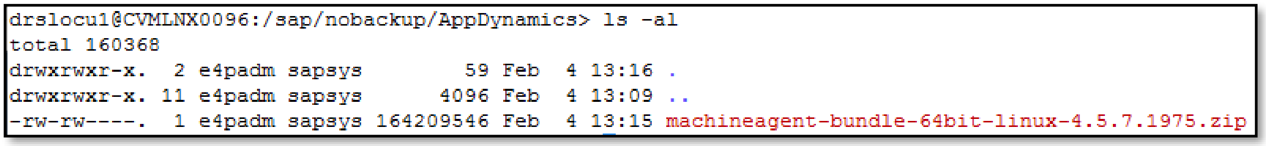
The Machine / Server Visibility Agent must be installed on every application server and central services server that will be monitored. The agent binary ZIP file will be named like "*machineagent-bundle-64bit-linux-4.5.7.1975.zip*".

4. Download the Machine / Server Visibility Agent from the location mentioned earlier in this document (see the section titled "*Download AppDynamics Agents*")

5. Save the agent zip file to a temporary location on the server, such as "*/sap/nobackup/AppDynamics*"

   Be sure to give the "*sidadm*" user full read/write/execute access to this directory.

   You should login as the "*sidadm*" user and then create the directory so that the "*sidadm*" user becomes the owner of the directory.



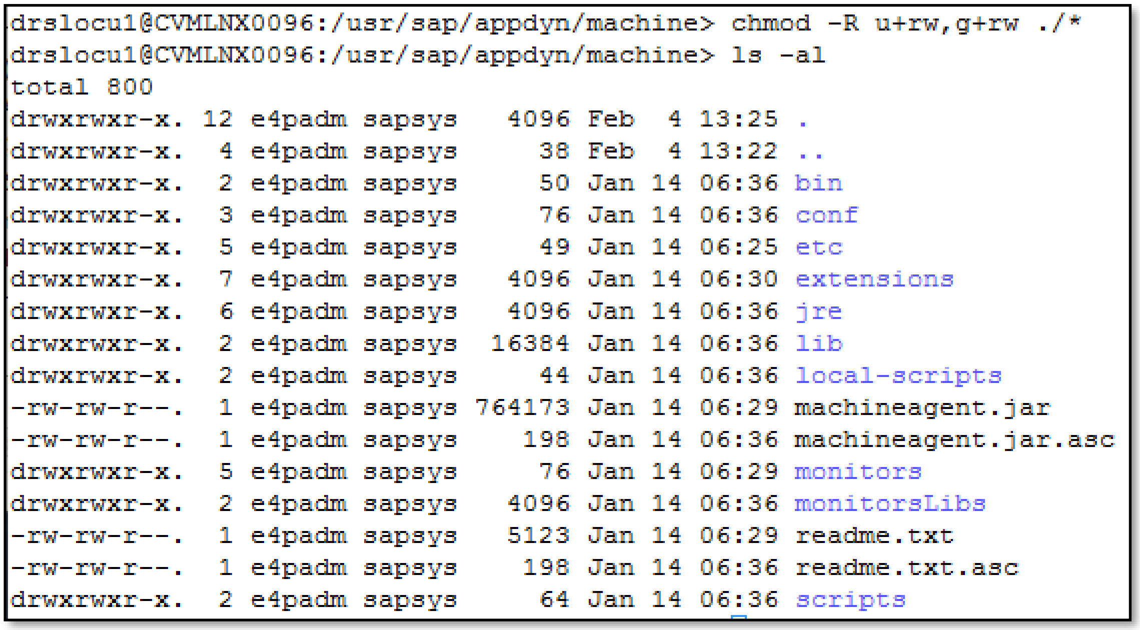
```
drslocu1@CVMLNX0096:/sap/nobackup/AppDynamics> ls -al
total 160368
drwxrwxr-x.  2 e4padm sapsys         59 Feb  4 13:16 .
drwxrwxr-x. 11 e4padm sapsys       4096 Feb  4 13:09 ..
-rw-rw----.  1 e4padm sapsys 164209546 Feb  4 13:15 machineagent-bundle-64bit-linux-4.5.7.1975.zip
```

6. Unzip the machine agent zip file into the "*/usr/sap/appdyn/machine*" directory:

```
unzip machineagent-bundle-64bit-linux-4.5.7.1975.zip -d /usr/sap/appdyn/machine
```

7. Change the permissions on the contents of the "*/usr/sap/appdyn/machine*" directory to give full read/write privileges at the owner and group levels:

```
cd /usr/sap/appdyn/machine

chmod -R u+rw,g+rw ./*
```

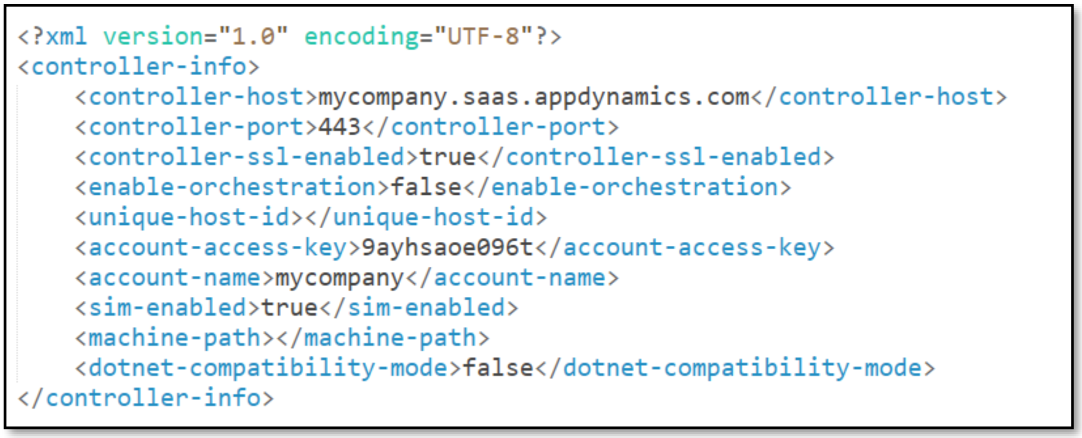


```
drslocu1@CVMLNX0096:/usr/sap/appdyn/machine> chmod -R u+rw,g+rw ./*
drslocu1@CVMLNX0096:/usr/sap/appdyn/machine> ls -al
total 800
drwxrwxr-x. 12 e4padm sapsys   4096 Feb  4 13:25 .
drwxrwxr-x.  4 e4padm sapsys     38 Feb  4 13:22 ..
drwxrwxr-x.  2 e4padm sapsys     50 Jan 14 06:36 bin
drwxrwxr-x.  3 e4padm sapsys     76 Jan 14 06:36 conf
drwxrwxr-x.  5 e4padm sapsys     49 Jan 14 06:25 etc
drwxrwxr-x.  7 e4padm sapsys   4096 Jan 14 06:30 extensions
drwxrwxr-x.  6 e4padm sapsys   4096 Jan 14 06:36 jre
drwxrwxr-x.  2 e4padm sapsys  16384 Jan 14 06:36 lib
drwxrwxr-x.  2 e4padm sapsys     44 Jan 14 06:36 local-scripts
-rw-rw-r--.  1 e4padm sapsys 764173 Jan 14 06:29 machineagent.jar
-rw-rw-r--.  1 e4padm sapsys    198 Jan 14 06:36 machineagent.jar.asc
drwxrwxr-x.  5 e4padm sapsys     76 Jan 14 06:29 monitors
drwxrwxr-x.  2 e4padm sapsys   4096 Jan 14 06:36 monitorsLibs
-rw-rw-r--.  1 e4padm sapsys   5123 Jan 14 06:29 readme.txt
-rw-rw-r--.  1 e4padm sapsys    198 Jan 14 06:36 readme.txt.asc
drwxrwxr-x.  2 e4padm sapsys     64 Jan 14 06:36 scripts
```

8. Edit the "*controller-info.xml*" file located in the "*/usr/sap/appdyn/machine/conf*" directory. Set the following elements in the xml file.

Property Name	Actual Property Value	Example Property Value	Additional Info
controller-host	Refer to "*AppDynamics Controller and Agent Properties*" section above	mycompany.saas.appdynamics.com	
controller-port	Refer to "*AppDynamics Controller and Agent Properties*" section above	443	
controller-ssl-enabled	Refer to "*AppDynamics Controller and Agent Properties*" section above	true	
account-access-key	Refer to "*AppDynamics Controller and Agent Properties*" section above	9ayhsaoe096t	
account-name	Refer to "*AppDynamics Controller and Agent Properties*" section above	mycompany	
sim-enabled	true	true	

The rest of the properties can be left as-is. This is what an example completed file looks like (all the comment blocks have been edited out to make it easier to read):

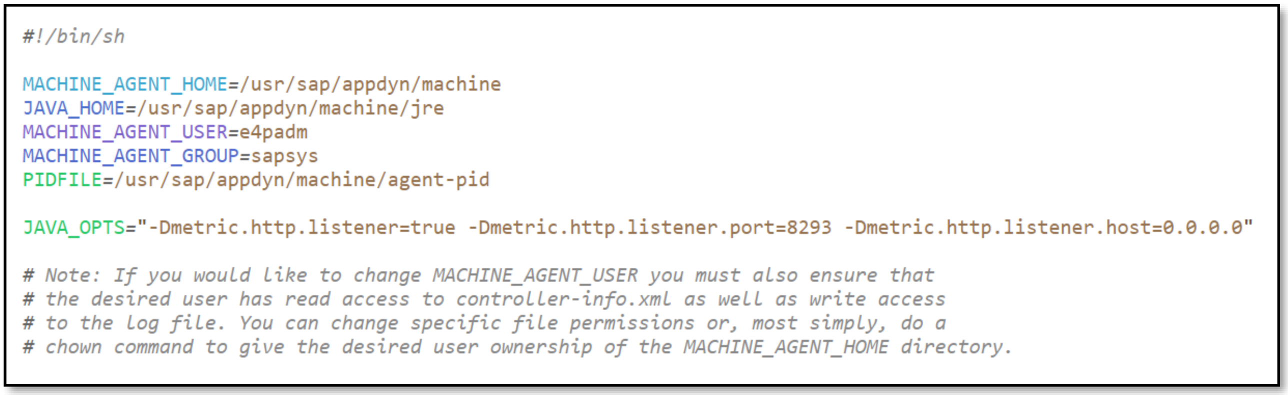


```xml
<?xml version="1.0" encoding="UTF-8"?>
<controller-info>
    <controller-host>mycompany.saas.appdynamics.com</controller-host>
    <controller-port>443</controller-port>
    <controller-ssl-enabled>true</controller-ssl-enabled>
    <enable-orchestration>false</enable-orchestration>
    <unique-host-id></unique-host-id>
    <account-access-key>9ayhsaoe096t</account-access-key>
    <account-name>mycompany</account-name>
    <sim-enabled>true</sim-enabled>
    <machine-path></machine-path>
    <dotnet-compatibility-mode>false</dotnet-compatibility-mode>
</controller-info>
```

9. Modify the following property values in the "*appdynamics-machine-agent*" file in the "*/usr/sap/appdyn/machine/etc/sysconfig*" directory:

Property Name	Actual Property Value	Example Property Value	Additional Info
MACHINE_AGENT_HOME	/usr/sap/appdyn/machine	/usr/sap/appdyn/machine	
JAVA_HOME	/usr/sap/appdyn/machine/jre	/usr/sap/appdyn/machine/jre	
MACHINE_AGENT_USER	set to the "sidadm" user	e4padm	
MACHINE_AGENT_GROUP	set to the group for the "sidadm" user	sapsys	
PIDFILE	/usr/sap/appdyn/machine/agent-pid	/usr/sap/appdyn/machine/agent-pid	
JAVA_OPTS	-Dmetric.http.listener=true -Dmetric.http.listener.port=8293 -Dmetric.http.listener.host=0.0.0.0	-Dmetric.http.listener=true -Dmetric.http.listener.port=8293 -Dmetric.http.listener.host=0.0.0.0	This is needed only if your installing the server monitoring agent on an ABAP app server

This is what an example completed file looks like:



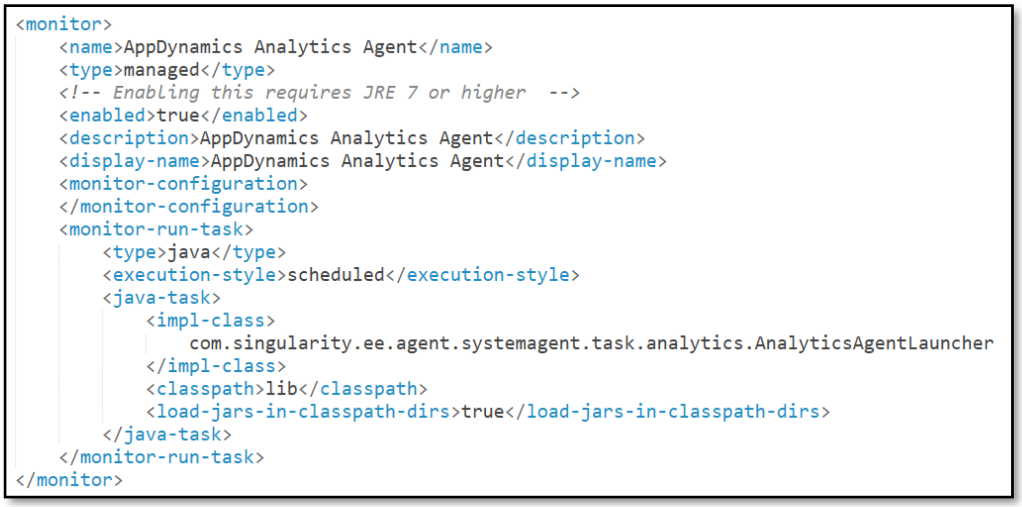
```
#!/bin/sh

MACHINE_AGENT_HOME=/usr/sap/appdyn/machine
JAVA_HOME=/usr/sap/appdyn/machine/jre
MACHINE_AGENT_USER=e4padm
MACHINE_AGENT_GROUP=sapsys
PIDFILE=/usr/sap/appdyn/machine/agent-pid

JAVA_OPTS="-Dmetric.http.listener=true -Dmetric.http.listener.port=8293 -Dmetric.http.listener.host=0.0.0.0"

# Note: If you would like to change MACHINE_AGENT_USER you must also ensure that
# the desired user has read access to controller-info.xml as well as write access
# to the log file. You can change specific file permissions or, most simply, do a
# chown command to give the desired user ownership of the MACHINE_AGENT_HOME directory.
```

10. Modify the following value in the “*monitor.xml*” file in the “*/usr/sap/appdyn/machine/monitors/analytics-agent*” directory:

11. Set to the “*enabled*” tag to “*true*”

12. <enabled>*true*</enabled>

    The rest of the properties can be left as-is. This is what an example completed file looks like:



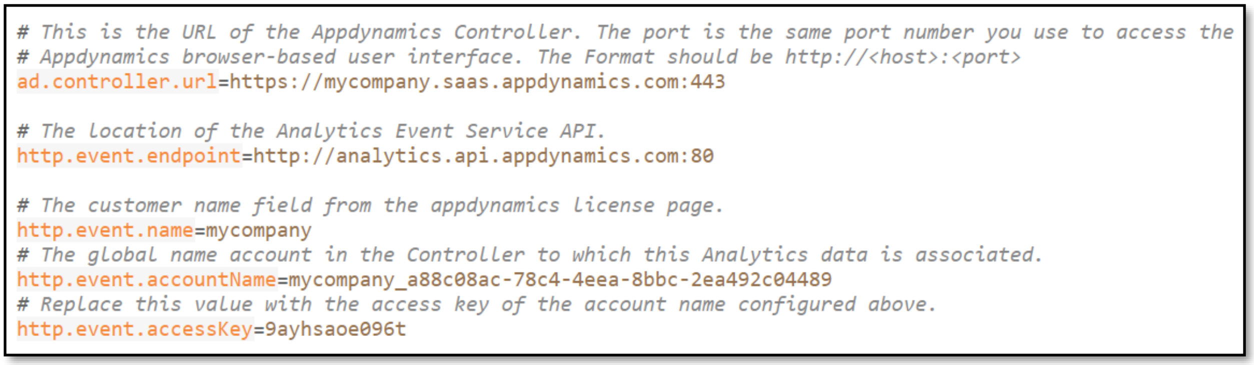
```
<monitor>
    <name>AppDynamics Analytics Agent</name>
    <type>managed</type>
    <!-- Enabling this requires JRE 7 or higher  -->
    <enabled>true</enabled>
    <description>AppDynamics Analytics Agent</description>
    <display-name>AppDynamics Analytics Agent</display-name>
    <monitor-configuration>
    </monitor-configuration>
    <monitor-run-task>
        <type>java</type>
        <execution-style>scheduled</execution-style>
        <java-task>
            <impl-class>
                com.singularity.ee.agent.systemagent.task.analytics.AnalyticsAgentLauncher
            </impl-class>
            <classpath>lib</classpath>
            <load-jars-in-classpath-dirs>true</load-jars-in-classpath-dirs>
        </java-task>
    </monitor-run-task>
</monitor>
```

13. Modify the following values in the “analytics-agent.properties” file in the “/usr/sap/appdyn/machine/monitors/analytics-agent/conf” directory:

Property Name	Actual Property Value	Example Property Value	Additional Info
ad.controller.url	Refer to “*AppDynamics Controller and Agent Properties*” section above “**controller-host**” “**controller-port**”	https://mycompany.saas.appdynamics.com:443	The Format should be http(s)://<host>:<port>

Property Name	Actual Property Value	Example Property Value	Additional Info
http.event.endpoint	Refer to "*AppDynamics Controller and Agent Properties*" section above "**analytics-events-service-host**" "**analytics-events-service-port**"	http://analytics.api.appdynamics.com:80	The Format should be http(s)://<host>:<port>
http.event.name	Refer to "*AppDynamics Controller and Agent Properties*" section above "**account-name**"	mycompany	
http.event.accountName	Refer to "*AppDynamics Controller and Agent Properties*" section above "**global-account-name**"	mycompany_a88c08ac-78c4-4eea-8bbc-2ea492c04489	
http.event.accessKey	Refer to "*AppDynamics Controller and Agent Properties*" section above "**account-access-key**"	9ayhsaoe096t	

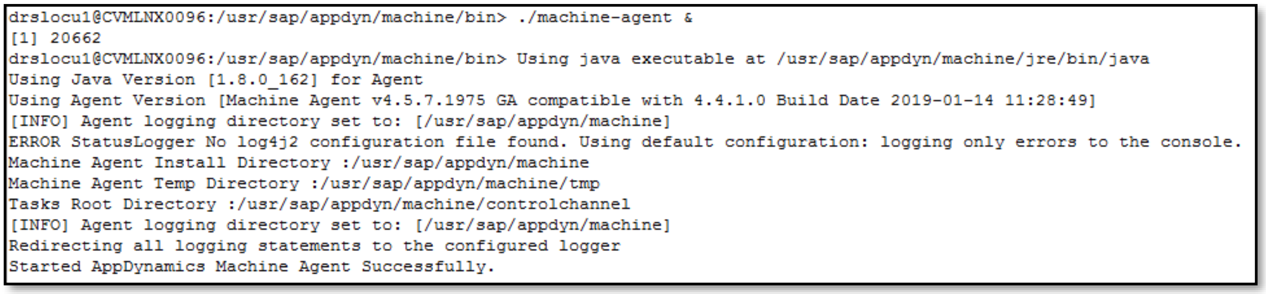
The rest of the properties can be left as-is. This is what an example completed file looks like (only properties that required modification are shown):



```
# This is the URL of the Appdynamics Controller. The port is the same port number you use to access the
# Appdynamics browser-based user interface. The Format should be http://<host>:<port>
ad.controller.url=https://mycompany.saas.appdynamics.com:443

# The location of the Analytics Event Service API.
http.event.endpoint=http://analytics.api.appdynamics.com:80

# The customer name field from the appdynamics license page.
http.event.name=mycompany
# The global name account in the Controller to which this Analytics data is associated.
http.event.accountName=mycompany_a88c08ac-78c4-4eea-8bbc-2ea492c04489
# Replace this value with the access key of the account name configured above.
http.event.accessKey=9ayhsaoe096t
```

14. Start the machine agent by going to the "bin" directory and executing the "machine-agent" script:

```
cd /usr/sap/appdyn/machine/bin

./machine-agent &
```



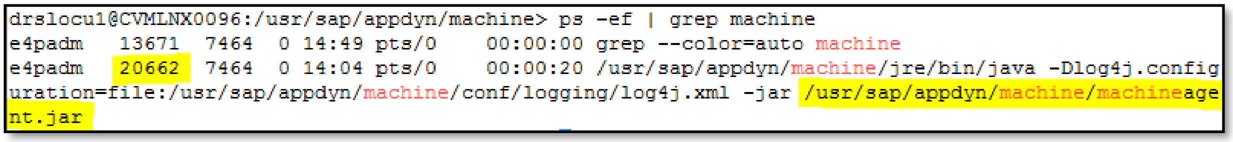
```
drslocu1@CVMLNX0096:/usr/sap/appdyn/machine/bin> ./machine-agent &
[1] 20662
drslocu1@CVMLNX0096:/usr/sap/appdyn/machine/bin> Using java executable at /usr/sap/appdyn/machine/jre/bin/java
Using Java Version [1.8.0_162] for Agent
Using Agent Version [Machine Agent v4.5.7.1975 GA compatible with 4.4.1.0 Build Date 2019-01-14 11:28:49]
[INFO] Agent logging directory set to: [/usr/sap/appdyn/machine]
ERROR StatusLogger No log4j2 configuration file found. Using default configuration: logging only errors to the console.
Machine Agent Install Directory :/usr/sap/appdyn/machine
Machine Agent Temp Directory :/usr/sap/appdyn/machine/tmp
Tasks Root Directory :/usr/sap/appdyn/machine/controlchannel
[INFO] Agent logging directory set to: [/usr/sap/appdyn/machine]
Redirecting all logging statements to the configured logger
Started AppDynamics Machine Agent Successfully.
```

Verify the Machine / Server Visibility Agent Installation

After following the steps in the previous section, you can verify the successful installation of the server visibility agent as follows:

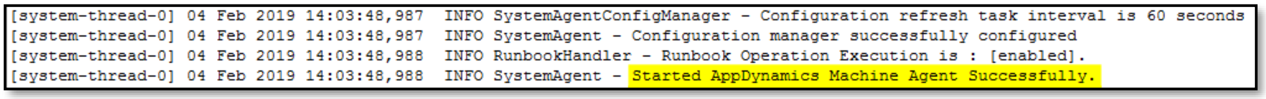
1. Check that the server visibility agent process is running on the server:

```
ps -ef | grep machine
```



```
drslocu1@CVMLNX0096:/usr/sap/appdyn/machine> ps -ef | grep machine
e4padm    13671  7464  0 14:49 pts/0    00:00:00 grep --color=auto machine
e4padm    20662  7464  0 14:04 pts/0    00:00:20 /usr/sap/appdyn/machine/jre/bin/java -Dlog4j.config
uration=file:/usr/sap/appdyn/machine/conf/logging/log4j.xml -jar /usr/sap/appdyn/machine/machineage
nt.jar
```

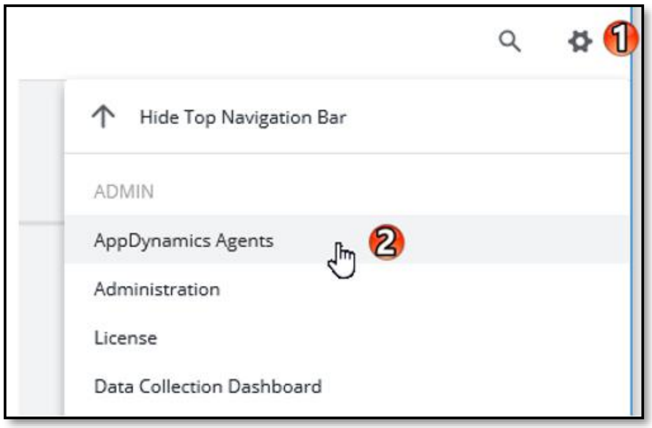
2. Check the agent log (*machine-agent.log*) located at “*/usr/sap/appdyn/machine/logs*”. You should see messages like the following:



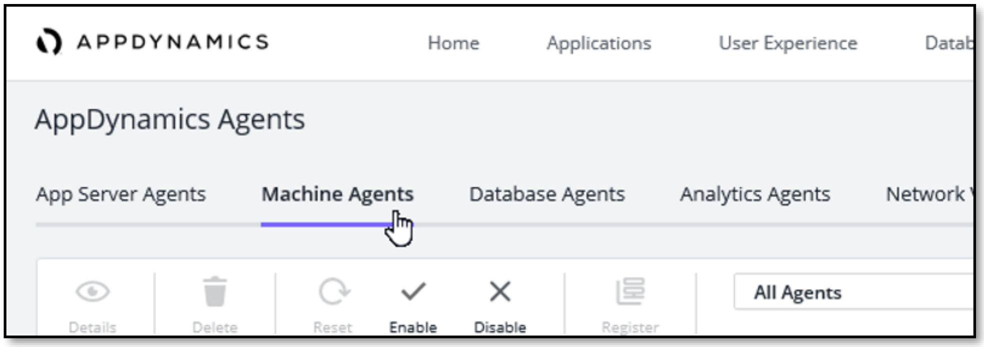
```
[system-thread-0] 04 Feb 2019 14:03:48,987  INFO SystemAgentConfigManager - Configuration refresh task interval is 60 seconds
[system-thread-0] 04 Feb 2019 14:03:48,987  INFO SystemAgent - Configuration manager successfully configured
[system-thread-0] 04 Feb 2019 14:03:48,988  INFO RunbookHandler - Runbook Operation Execution is : [enabled].
[system-thread-0] 04 Feb 2019 14:03:48,988  INFO SystemAgent - Started AppDynamics Machine Agent Successfully.
```

3. Log into the Controller UI as an administrator.

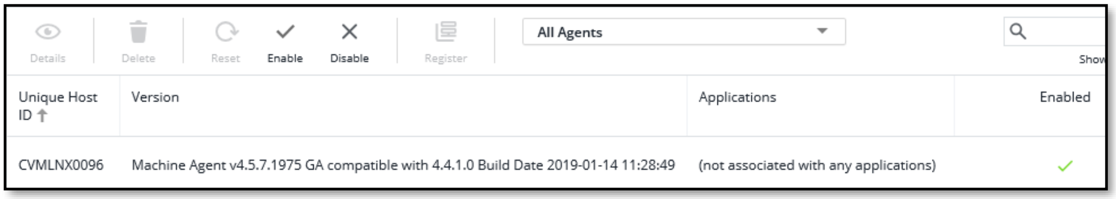
4. Click the gear icon at the top right of the screen and then select “AppDynamics Agents” from the drop-down list:



5. Select the “*Machine Agents*” tab:

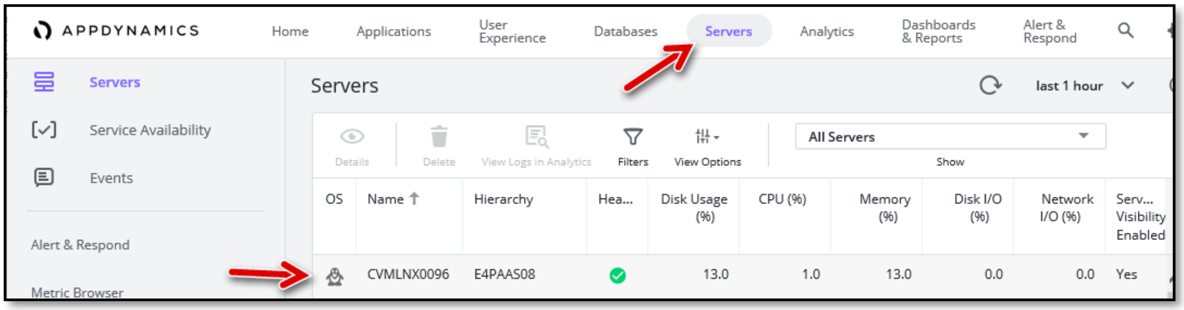


6. You should be able to find the agent that was just installed in the list of machine agents. The “*Enabled*” column should have a green checkmark.



Unique Host ID ↑	Version	Applications	Enabled
CVMLNX0096	Machine Agent v4.5.7.1975 GA compatible with 4.4.1.0 Build Date 2019-01-14 11:28:49	(not associated with any applications)	✓

7. If you have set “*sim-enabled*” to “*true*” then you should also see the server details on the “*Servers*” tab in the controller:



OS	Name ↑	Hierarchy	Hea...	Disk Usage (%)	CPU (%)	Memory (%)	Disk I/O (%)	Network I/O (%)	Serv... Visibility Enabled
	CVMLNX0096	E4PAAS08	✓	13.0	1.0	13.0	0.0	0.0	Yes