Workshop: Tips and Tricks for Implementing the Progress Application Server (PAS) for OpenEdge

Version 1.5 September 26, 2019

Notices

These materials and all Progress® software products are copyrighted and all rights are reserved by Progress Software Corporation. The information in these materials is subject to change without notice, and Progress Software Corporation assumes no responsibility for any errors that may appear therein. The references in these materials to specific platforms supported are subject to change.

Business Making Progress, Corticon, DataDirect (and design), DataDirect Cloud, DataDirect Connect, DataDirect Connect64, DataDirect XML Converters, DataDirect XQuery, Deliver More Than Expected, Easyl, Fathom, Icenium, Kendo UI, Making Software Work Together, OpenEdge, Powered by Progress, Progress, Progress Control Tower, Progress RPM, Progress Software Business Making Progress, Progress Software Developers Network, Rollbase, RulesCloud, RulesWorld, SequeLink, Sitefinity (and Design), SpeedScript, Stylus Studio, TeamPulse, Telerik, (Telerik (and Design), Test Studio, and WebSpeed are registered trademarks of Progress Software Corporation or one of its affiliates or subsidiaries in the U.S. and/or other countries. AccelEvent, AppsAlive, AppServer, BravePoint, BusinessEdge, DataDirect Spy, DataDirect SupportLink, , Future Proof, High Performance Integration, Modulus, NativeScript, OpenAccess, Pacific, ProDataSet, Progress Arcade, Progress Pacific, Progress Profiles, Progress Results, Progress RFID, Progress Software, ProVision, PSE Pro, SectorAlliance, Sitefinity, SmartBrowser, SmartComponent, SmartDataBrowser, SmartDataObjects, SmartDataView, SmartDialog, SmartFolder, SmartFrame, SmartObjects, SmartPanel, SmartQuery, SmartViewer, SmartWindow, WebClient, and Who Makes Progress are trademarks or service marks of Progress Software Corporation and/or its subsidiaries or affiliates in the U.S. and other countries. Java is a registered trademark of Oracle and/or its affiliates. Any other marks contained herein may be trademarks of their respective owners.

Please refer to the Release Notes applicable to the particular Progress product release for any third-party acknowledgements required to be provided in the documentation associated with the Progress product.

The Release Notes can be found in the OpenEdge installation directory and online at: https://community.progress.com/technicalusers/w/openedgegeneral/1329.openedge-product-documentation-overview.aspx.

For the latest documentation updates see OpenEdge Product Documentation on Progress Communities: (https://community.progress.com/technicalusers/w/openedgegeneral/1329.openedge-product-documentation-overview.aspx).

April 2019

Last updated with new content: Release 12.0.0

Contents

Tips and Tricks for Implementing the OpenEdge Application Server

LAB 1

Understanding PAS for OpenEdge

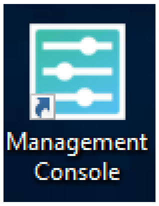
Overview

Some daily administration tasks for Progress Application Server (PAS) for OpenEdge.

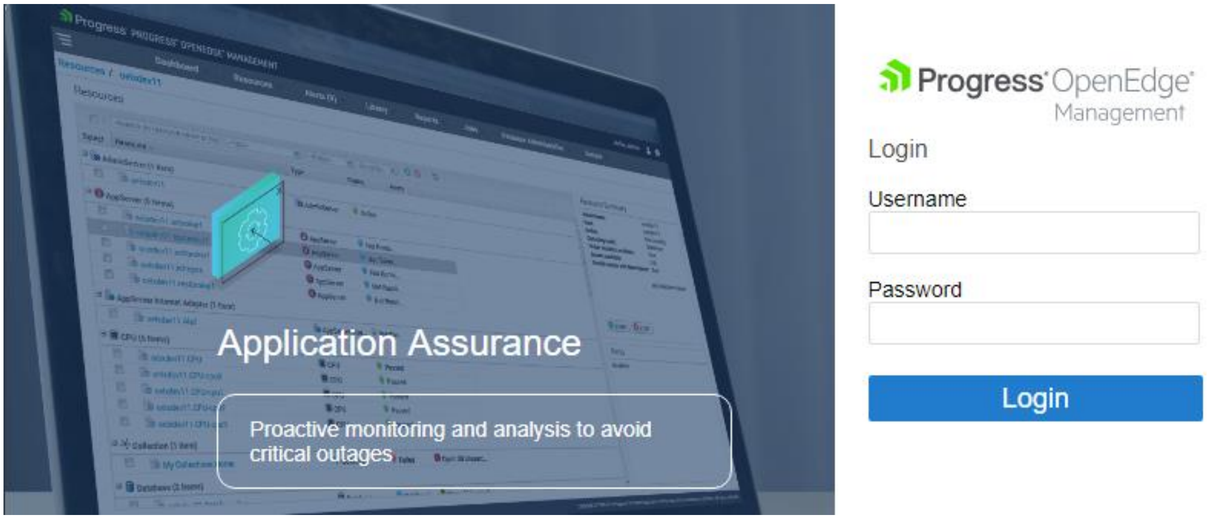
- Understand using OpenEdge Explorer/Management
- Understand using "tcman" command line interface
- Understand msagent connections in shared memory database connection
- Finding the running processes
- Using a browser to verify PAS for OpenEdge is running
- Using a browser to get metrics

Starting PAS for OpenEdge with OpenEdge Explorer/Management

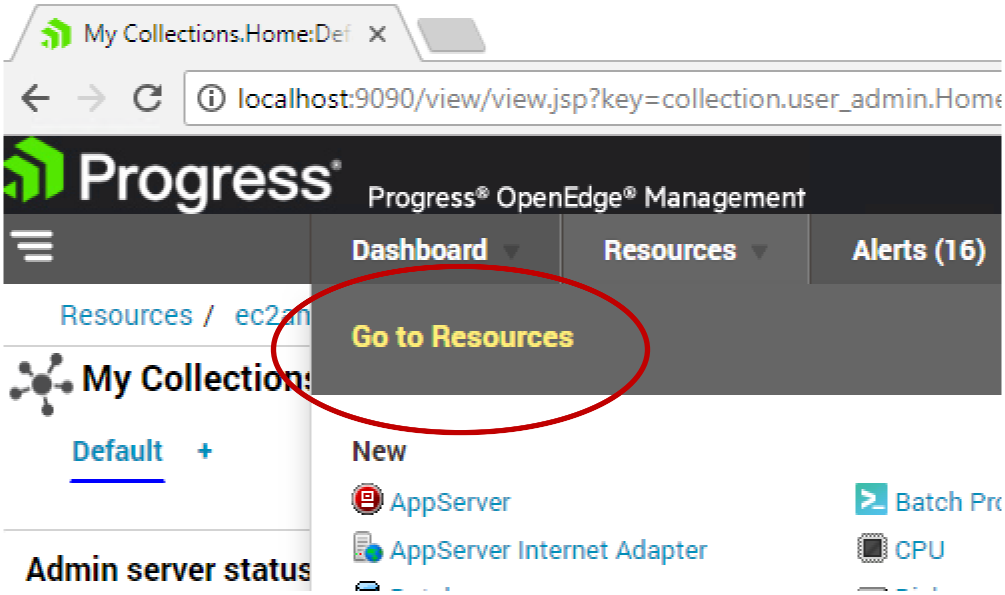
1. To start OpenEdge Explorer/Management, double-click on the desktop icon.



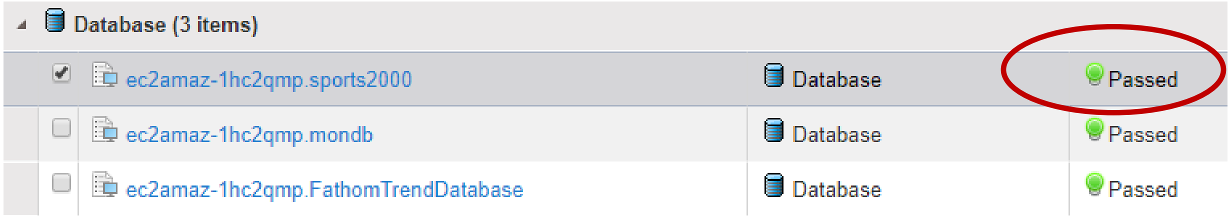
2. Login using these credentials:
   Username: admin
   Password: 4admin


Application Assurance
Proactive monitoring and analysis to avoid critical outages
Progress OpenEdge Management
Login
Username
Password
Login

3. Select Tab “Resources” and then select “Go to Resources”.



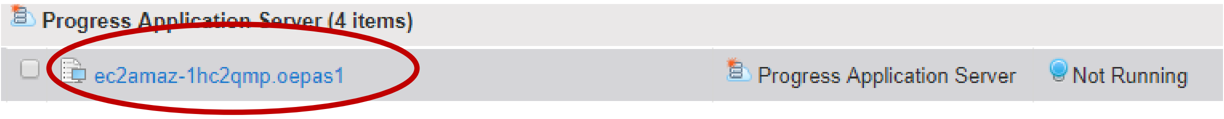
4. On the Resources page, scroll down to Database “sports2000” and make sure it is running.



Database (3 items)

☑	ec2amaz-1hc2qmp.sports2000	Database	Passed
□	ec2amaz-1hc2qmp.mondb	Database	Passed
□	ec2amaz-1hc2qmp.FathomTrendDatabase	Database	Passed

5. Scroll down to the Progress Application Server resources and click the “oepas1” link to open the oepas1 server instance page.

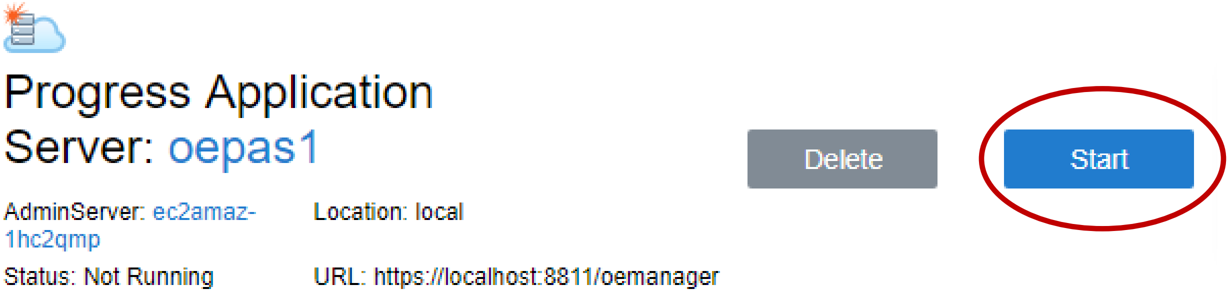


Progress Application Server (4 items)

□	ec2amaz-1hc2qmp.oepas1	Progress Application Server	Not Running

NOTE: selecting the link directs you to the detailed Resource home page

6. On the oepas1 Resource page, click the start button to start the server instance.



Progress Application Server: oepas1

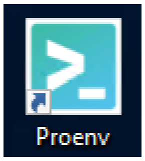
AdminServer: ec2amaz-1hc2qmp

Location: local

Status: Not Running

URL: https://localhost:8811/oemanager

7. On your desktop, double-click on the PROENV icon to access the command line utilities.



```
proenv> cd
```

```
proenv>cd
C:\OpenEdge\WRK
```

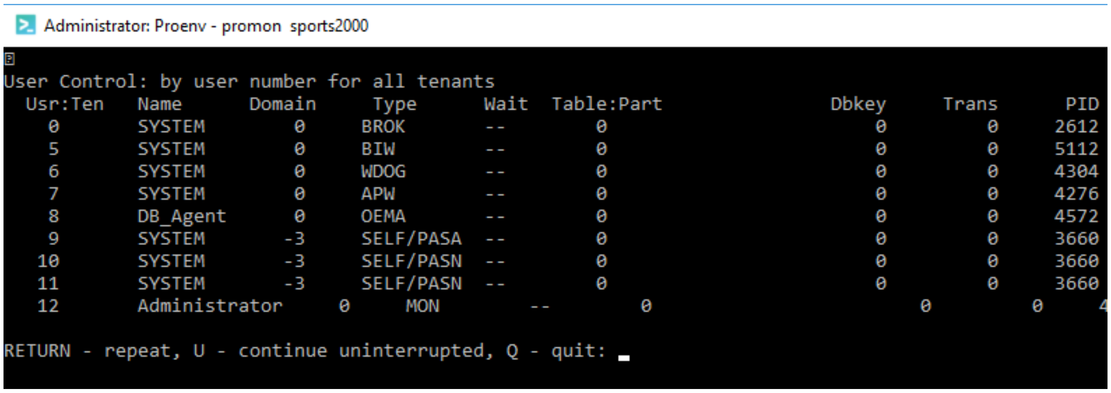
8. Start “promon” to view database user connections.

```
proenv> promon sports2000
proenv>promon sports2000
OpenEdge Release 12.0 as of Fri Fri Feb 22 19:02:39 EST
2019

  OpenEdge MONITOR Release 12

  Database: C:\OpenEdge\WRK\db\sports2000

  1.  User Control
  2.  Locking and Waiting Statistics
  3.  Block Access
  4.  Record Locking Table
  5.  Activity
  6.  Shared Resources
  7.  Database Status
  8.  Shut Down Database
  9.  Currently Connected Tenants

 R&D.  Advanced options
  T.  2PC Transactions Control
  L.  Resolve 2PC Limbo Transactions
  C.  2PC Coordinator Information

  J.  Resolve JTA Transactions

  M.  Modify Defaults
  Q.  Quit

  Enter your selection:
```

9. Enter “1” for User Control.

```
  Enter your selection: 1

  1.  Display all entries
  2.  Match a user number
  3.  Match a range of user numbers

  5.  Change list sorting
  6.  Match a tenant or tenants

  Q.  Return to main menu

  Enter your selection:
```

10. Enter “1” for Display all entries.



```
User Control: by user number for all tenants
  Usr:Ten    Name       Domain     Type       Wait  Table:Part              Dbkey     Trans      PID
    0        SYSTEM          0     BROK       --         0                      0         0     2612
    5        SYSTEM          0     BIW        --         0                      0         0     5112
    6        SYSTEM          0     WDOG       --         0                      0         0     4304
    7        SYSTEM          0     APW        --         0                      0         0     4276
    8        DB_Agent        0     OEMA       --         0                      0         0     4572
    9        SYSTEM         -3     SELF/PASA  --         0                      0         0     3660
   10        SYSTEM         -3     SELF/PASN  --         0                      0         0     3660
   11        SYSTEM         -3     SELF/PASN  --         0                      0         0     3660
   12        Administrator      0      MON        --         0                      0         0     4

RETURN - repeat, U - continue uninterrupted, Q - quit:
```

NOTICE: ***The PASA user, this is the Agents administration thread***
The PASN users, these are the ABL sessions started initially

Type “q” enter and then “q” enter again to quit and exit promon.

11. Run “tcman env” to see environment and status of “oepas1”.

```
proenv> cd %WRKDIR%\oepas1
proenv> bin\tcman.bat env
```

```
proenv>bin\tcman.bat env
catalina home:     C:\Progress\OpenEdge\servers\pasoe
catalina base:     C:\OpenEdge\WRK\oepas1
catalina tmpdir    C:\OpenEdge\WRK\oepas1\temp
catalina pid:      C:\OpenEdge\WRK\oepas1\logs\catalina-oepas1.pid
java home:         C:\Progress\OpenEdge\jdk
jre home:
manager http port: 8810
manager https port:8811
manager shut port: 8812
manager URL:       http://localhost:8810/manager
config type:       instance
config alias:      oepas1
config parent:     C:/Progress/OpenEdge/servers/pasoe
server running:    1
instance tracking: True
instance file:
C:\Progress\OpenEdge\servers\pasoe\conf\instances.windows
server process-id: 4944
window title:      PAS--55920
security model:    developer
service:           false
```

12. Run “tcman plist” to see processes for “oepas1”.

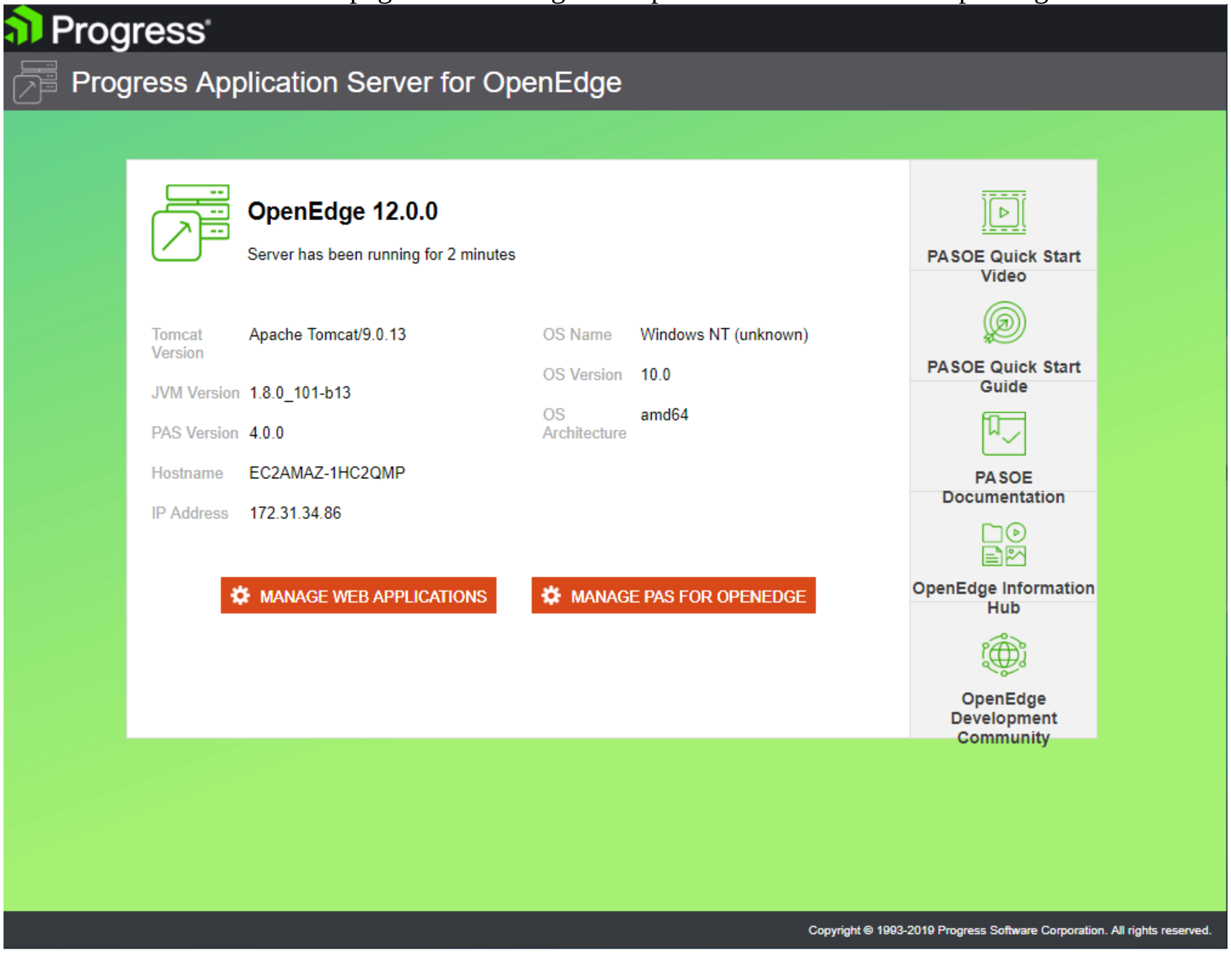
```
proenv> cd %WRKDIR%\oepas1
proenv> bin\tcman.bat plist -f
```

```
proenv>bin\tcman.bat plist -f
4944
+3660
```

4944 is the example Tomcat process ID
+3660 is the example msagent process ID

13. Verify PAS for OpenEdge is running in a browser by entering:
http://localhost:8810

This is the home page for a running development licensed PAS for OpenEdge.



Progress Application Server for OpenEdge

OpenEdge 12.0.0

Server has been running for 2 minutes

Tomcat Version: Apache Tomcat/9.0.13
JVM Version: 1.8.0_101-b13
PAS Version: 4.0.0
Hostname: EC2AMAZ-1HC2QMP
IP Address: 172.31.34.86

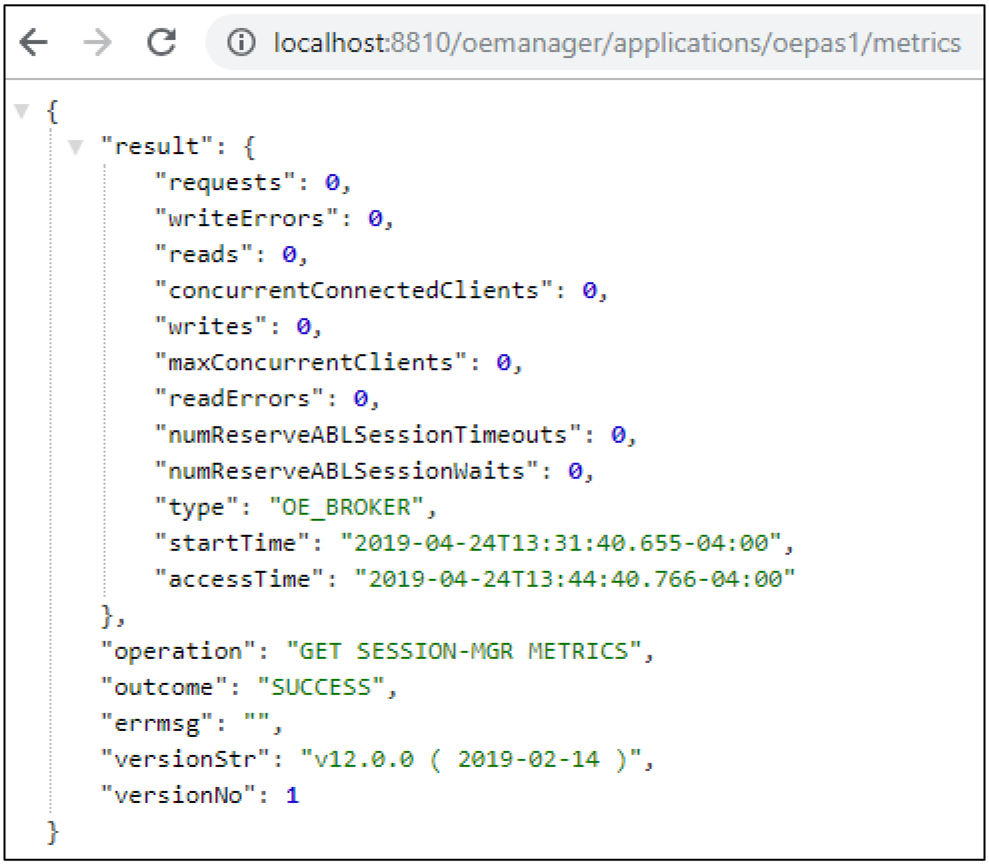
OS Name: Windows NT (unknown)
OS Version: 10.0
OS Architecture: amd64

14. Access the oemanager REST APIs by entering this URL in your browser:
http://localhost:8810/oemanager/applications/oepas1/metrics

UserName:	tomcat
Password:	tomcat

NOTE: be sure to change this password in your production environments!

This is like the Classic AppServer's "asbman -query" command.



```
{
    "result": {
        "requests": 0,
        "writeErrors": 0,
        "reads": 0,
        "concurrentConnectedClients": 0,
        "writes": 0,
        "maxConcurrentClients": 0,
        "readErrors": 0,
        "numReserveABLSessionTimeouts": 0,
        "numReserveABLSessionWaits": 0,
        "type": "OE_BROKER",
        "startTime": "2019-04-24T13:31:40.655-04:00",
        "accessTime": "2019-04-24T13:44:40.766-04:00"
    },
    "operation": "GET SESSION-MGR METRICS",
    "outcome": "SUCCESS",
    "errmsg": "",
    "versionStr": "v12.0.0 ( 2019-02-14 )",
    "versionNo": 1
}
```

NOTE: Nicely formatted because JSON Formatter extension added to chrome.

Section summary

PAS for OpenEdge has many different ways that you can configure, manage, and monitor it. Choose what works best for you.

Tips and Tricks for Implementing the OpenEdge Application Server

LAB 2

Configure PAS for OpenEdge for your application

Overview

This section covers the steps to configure your PAS for OpenEdge application based on your Classic AppServer information.

Manually configure your PAS for OpenEdge based on a sample Broker query and its Broker settings in the ubroker.properties file.

Use paspropconv to configure your PAS for OpenEdge.

Manually configure your PAS for OpenEdge

Review a sample asbman -query and sample ubroker.properties provided below to configure PAS for OpenEdge properties used by your new instance in step #3.

1. This is a sample asbman query output (don’t run the command).

   What would be the most useful numbers for setting you maxConnectionsPerAgent?

   ```
   Broker Name                    : MyASBroker
   Operating Mode                 : Stateless
   Broker Status                  :  ACTIVE
   Broker Port                    : 13500
   Broker PID                     : 42745
   Active Agents                  : 10
   Busy Agents                    : 1
   Locked Agents                  : 0
   Available Agents               : 9
   Active Clients (now, peak)     : (0, 84)
   Client Queue Depth (cur, max)  : (0, 80)
   Total Requests                 : 6396050
   Rq Wait (max, avg)             : (7620 ms, 3 ms)
   Rq Duration (max, avg)         : (7620 ms, 3 ms)
   ```

2. This is a sample section of a ubroker.properties for MyASBroker.

   ```
   [UBroker.WS.MyASBroker]
       appserviceNameList=MyASBroker
       controllingNameServer=NS1
       maxSrvrInstance=80
       minSrvrInstance=3
       portNumber=19100
   ```

3. Fill in the openedge.properties values based on the sample values provided in steps #1 and #2.

```
[AppServer.SessMgr.myoepas]

   maxABLSessionsPerAgent=___________

   maxAgents=___________

   maxConnectionsPerAgent=___________

   numInitialAgents=___________

   minAgents=___________

   agentStartLimit=___________

[AppServer.Agent.myoepas]

   numInitialSessions=___________
```

NOTE: the *psc.as.executor.maxthreads=300* should not need to be changed but remember to change it if you have more than 300 concurrent client connections.

4. Use the new "paspropconv" utility to convert a Classic AppServer configuration to PAS for OpenEdge.

5. Open the 11.7.4 proenv window by double-clicking the icon.

6. Run "paspropconv" on the ubroker.properties to get a merge file.

```
proenv> cd %WRKDIR%
proenv> paspropconv.bat --ubrokerPropsFile
%DLC%\properties\ubroker.properties --ubrokerName
UBroker.AS.MyASBroker --pasoeAppName mypasoe
```

7. After the command executes, the console does not output any message. The command outputs the following files to the %WRKDIR%:
   a. mypasoe.MyASBroker.oemerge – merge file for PAS for OpenEdge
   b. MyASBroker_setenv.bat – environment file for Windows (we won't use this file in this lab)
   c. MyASBroker_setenv.sh – environment file for UNIX (we won't use this file in this lab)
   d. paspropconv.log – log file for property conversion

```
proenv>dir *MyASBroker*
```

```
proenv>dir *MyASBroker*
 Volume in drive C has no label.
 Volume Serial Number is AAD5-F33A

 Directory of C:\OpenEdge\WRK

05/20/2018  01:10 AM                21 MyASBroker_setenv.bat
05/20/2018  01:10 AM                35 MyASBroker_setenv.sh
05/20/2018  01:10 AM            15,308 mypasoe.MyASBroker.oemerge
               3 File(s)         15,364 bytes
               0 Dir(s)  60,294,086,656 bytes free
```

8. CLOSE THE 11.7.4 PROENV WINDOW!

9. Create a new PAS for OpenEdge instance in 12.0 to merge the MyASBroker configuration.

10. Open the 12.0 proenv window again (if you closed it before)

   a. We will use "pasman" this time, just to shake things up!
   b. We will use the -v (verbose) command because its more interesting.
   c. And the -f will copy the manager.war and oemanager.war files from %DLC%\servers\pasoe\webapps.

```
proenv> cd %WRKDIR%
proenv> pasman create -v -f -p 19100 -P 19101 -s 19102 mypasoe
proenv>pasman create -v -f -p 19100 -P 19101 -s 19102 mypasoe
info: TCManager 4.0.0
info: executing create Tomcat instance command
info: creating new server instance at: C:\OpenEdge\117\WRK\mypasoe
info: creating instance directory structure: C:\OpenEdge\117\WRK\mypasoe
info: populating CATALINA_HOME files into instance directory structure:
C:\OpenEdge\117\WRK\mypasoe
info: copying CATALINA_HOME ROOT application into instance directory structure
info: copying CATALINA_HOME web application archives into instance directory structure
info: copying CATALINA_HOME conf files into instance directory structure
info: copying CATALINA_HOME bin files into instance directory structure
info: copying CATALINA_HOME bin\*_tlr tailoring files into instance directory structure
info: copying CATALINA_HOME bin\*_setenv files into instance directory structure
info: copying CATALINA_HOME bin\*_startup files into instance directory structure
info: copying CATALINA_HOME bin\*_shutdown files into instance directory structure
info: executing CATALINA_HOME product specific scripts
info: executing oe_manager create
info: executing create instance command
info: verifying source PASOE bin files
info: copying PASOE bin files into directory structure
info: cpyfile: C:\OpenEdge\WRK\mypasoe\bin\deployREST.sh
info: cpyfile: C:\OpenEdge\WRK\mypasoe\bin\deploySOAP.sh
info: cpyfile: C:\OpenEdge\WRK\mypasoe\bin\oeprop.sh
info: cpyfile: C:\OpenEdge\WRK\mypasoe\bin\deploysvc.sh
info: cpyfile: C:\OpenEdge\WRK\mypasoe\bin\oeabl_tlr.sh
info: cpyfile: C:\OpenEdge\WRK\mypasoe\bin\wscgi_tlr.sh
info: cpyfile: C:\OpenEdge\WRK\mypasoe\bin\oeproxy_tlr.sh
info: cpyfile: C:\OpenEdge\WRK\mypasoe\bin\oests_tlr.sh
info: cpyfile: C:\OpenEdge\WRK\mypasoe\bin\oedbg_tlr.sh
info: cpyfile: C:\OpenEdge\WRK\mypasoe\bin\oedbg_tlr.sh
info: cpyfile: C:\OpenEdge\WRK\mypasoe\bin\oejmx.sh
info: cpyfile: C:\OpenEdge\WRK\mypasoe\bin\oewatcher.sh
info: updating CATALINA_BASE in instance directory tlr files
info: updating CATALINA_BASE in C:\OpenEdge\WRK\mypasoe\bin\oeabl_tlr.bat to
C:\OpenEdge\WRK\mypasoe
info: updating CATALINA_BASE in C:\OpenEdge\WRK\mypasoe\bin\wscgi_tlr.bat to
C:\OpenEdge\WRK\mypasoe
info: updating CATALINA_BASE in C:\OpenEdge\WRK\mypasoe\bin\oeproxy_tlr.bat to
C:\OpenEdge\WRK\mypasoe
info: updating CATALINA_BASE in C:\OpenEdge\WRK\mypasoe\bin\oests_tlr.bat to
C:\OpenEdge\WRK\mypasoe
info: updating CATALINA_BASE in C:\OpenEdge\WRK\mypasoe\bin\oedbg_tlr.bat to
C:\OpenEdge\WRK\mypasoe
info: copying CATALINA_HOME openedge folders into instance directory structure
info: running oeabl tailoring script
info: begin tailoring ABL for the Progress Application Server
info: creating instance with application name mypasoe
info: Creating context.xml for context name ROOT
info: adding properties to openedge.properties for ABL instance mypasoe
info: setting application to mypasoe in openedge.properties
info: adding ROOT to webapps in openedge.properties
info: security model is developer
info: enabling server status pages for development server
info: setting intitial sessions to 2 for development server
info: enabling APSV transport for development server
info: enabling SOAP transport for development server
info: enabling REST transport for development server
info: enabling WEB transport for development server
info: setting default WEB handler to OpenEdge.Web.CompatibilityHandler
info: updating PROPATH with webapp directory
info: updated PROPATH to ${CATALINA_BASE}/webapps/ROOT/WEB-
```

11. Change directory to mypasoe and merge the OEMERGE file.

    ```
    proenv>cd mypasoe
    proenv>bin\oeprop.bat -f
    C:\OpenEdge\117\WRK\mypasoe.MyASBroker.oemerge
    ```

12. Open notepad++ and view the C:\OpenEdge\WRK\mypasoe\conf\openedge.properties.

NOTE: The oemerge file and now the openedge.properties file is full of comments that explain what was done! Take the time to read the comments!

13. Do your manual changes match the changes done by “paspropconv”?

14. EXTRA CREDIT – start mypasoe using “tcman start” and verify it is running.

    Hints: to verify tcman started: “tcman env”, “tcman plist”, use a browser and hit the home page.

Section summary

Taking the time to configure your PAS for OpenEdge instance before testing can save you many problems down the road. Using “paspropconv” makes it easier, but you can also manually make the conversion.

Tips and Tricks for Implementing the OpenEdge Application Server

LAB 3

Monitoring your production PAS for OpenEdge

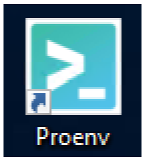
Overview

Review of the different ways to monitor your PAS for OpenEdge instance.

- OE Explorer and OE Management
- oemanager REST API
- oejmx script

Monitoring your production PAS for OpenEdge with OE Explorer

1. Open a **proenv** command prompt from the icon on the Windows desktop display of your workshop machine.



2. Start the “monpasoe” instance. The database monpasoe connects to is auto-started through the AdminServer. Let’s use the pasman command with pasoestart this time to start the instance.

```
proenv> cd %WRKDIR%
proenv> pasman pasoestart -I monpasoe -v
```

```
proenv>pasman pasoestart -I monpasoe -v
info: TCManager 4.0.0
info: Loading PASOE instance process IDs 0
Starting stopped PASOE instance monpasoe
info: Using startup timeout of 120 seconds
info: Cleaning previous log files
info: executing clean Tomcat logs command
info: Java starting server instance at C:\OpenEdge\WRK\monpasoe
info: waiting for window 'PAS-monpasoe-33091' ( 2988 ) to start
info: setting window title ( 459648 ) to PAS-monpasoe-33091
info: Waiting for PASOE instance startup to complete...
.............
```

3. Run “clientLoop.p” to make 100 requests to “monpasoe”.

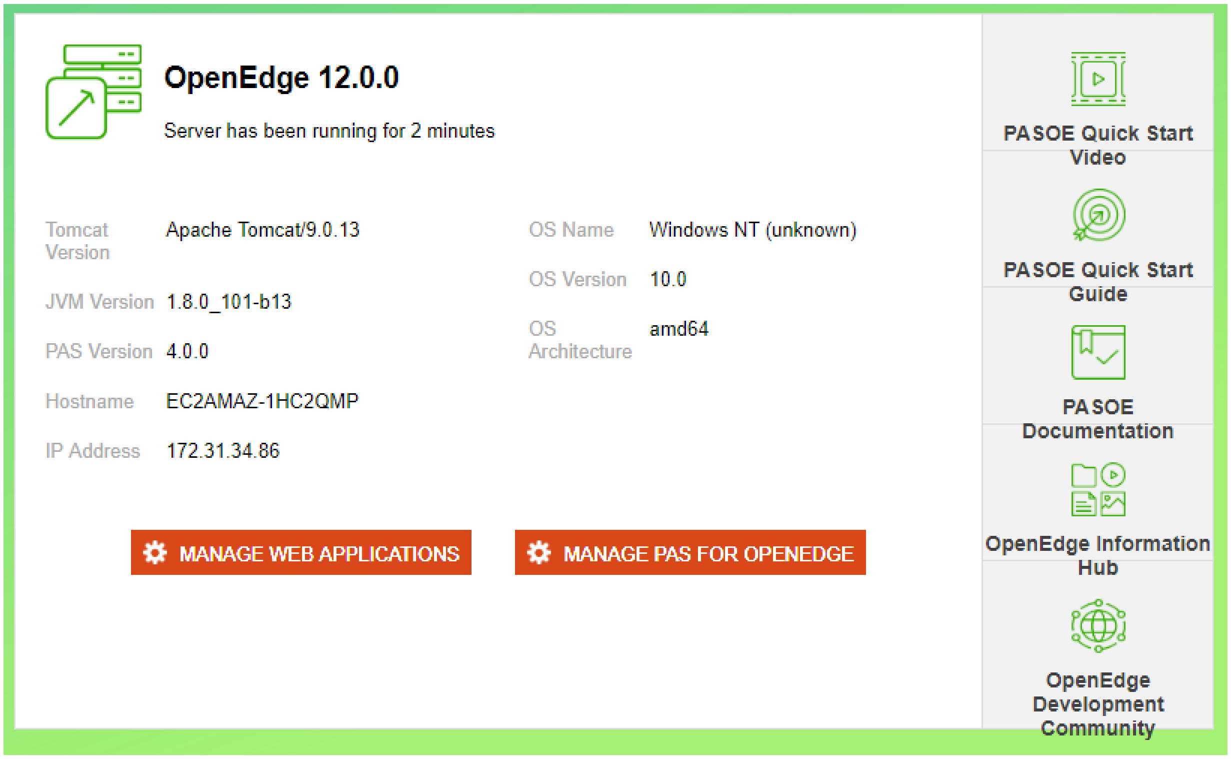
```
proenv> cd C:\OpenEdge\WRK
proenv> prowin -b -p clientLoop.p
```

This will run 100 requests to “monpasoe”.

4. Start the Chrome browser and hit the home page for monpasoe.

   `http://localhost:20000`

5. Select the “Manage Web Applications” button.

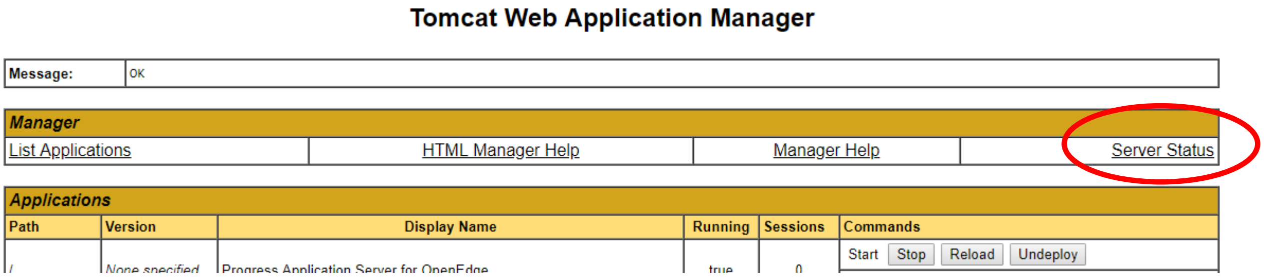


OpenEdge 12.0.0

Server has been running for 2 minutes

Tomcat Version	Apache Tomcat/9.0.13	OS Name	Windows NT (unknown)
JVM Version	1.8.0_101-b13	OS Version	10.0
PAS Version	4.0.0	OS Architecture	amd64
Hostname	EC2AMAZ-1HC2QMP		
IP Address	172.31.34.86		

6. Enter the Username and Password (tomcat/tomcat).
   NOTE: Change this password in your production environments!
7. This will bring to you the Tomcat Manager application (can also get there with http://hostname:port/manager). Select the “Server Status” link.



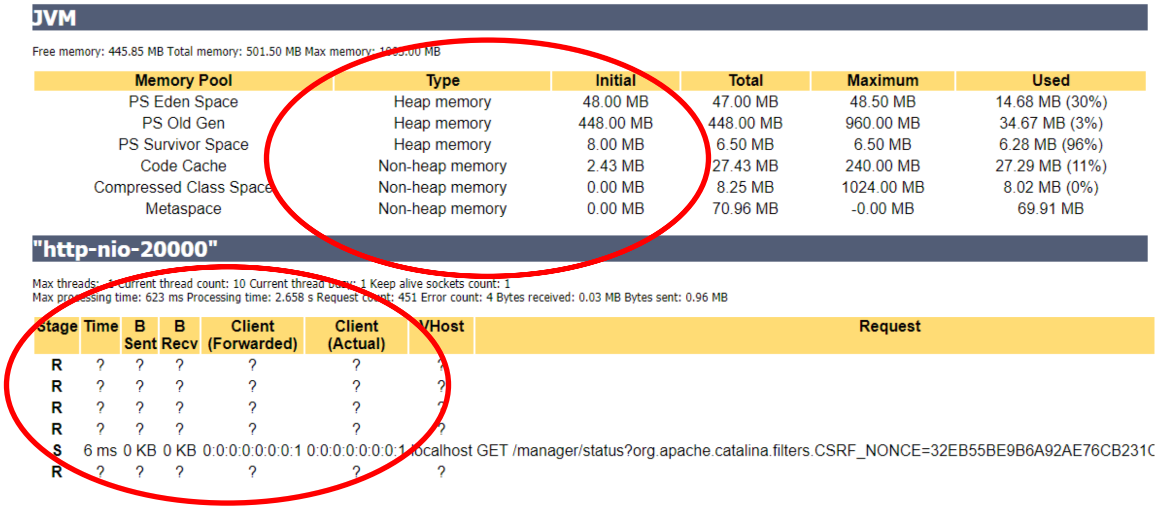
Tomcat Web Application Manager

Message:	OK

Manager			
List Applications	HTML Manager Help	Manager Help	Server Status

Applications					
Path	Version	Display Name	Running	Sessions	Commands
/	None specified	Progress Application Server for OpenEdge	true	0	Start Stop Reload Undeploy

8. Notice the memory information and current threads.



JVM

Free memory: 445.85 MB Total memory: 501.50 MB Max memory: 1003.00 MB

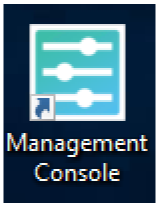
Memory Pool	Type	Initial	Total	Maximum	Used
PS Eden Space	Heap memory	48.00 MB	47.00 MB	48.50 MB	14.68 MB (30%)
PS Old Gen	Heap memory	448.00 MB	448.00 MB	960.00 MB	34.67 MB (3%)
PS Survivor Space	Heap memory	8.00 MB	6.50 MB	6.50 MB	6.28 MB (96%)
Code Cache	Non-heap memory	2.43 MB	27.43 MB	240.00 MB	27.29 MB (11%)
Compressed Class Space	Non-heap memory	0.00 MB	8.25 MB	1024.00 MB	8.02 MB (0%)
Metaspace	Non-heap memory	0.00 MB	70.96 MB	-0.00 MB	69.91 MB

"http-nio-20000"

Max threads: -1 Current thread count: 10 Current thread busy: 1 Keep alive sockets count: 1
Max processing time: 623 ms Processing time: 2.658 s Request count: 451 Error count: 4 Bytes received: 0.03 MB Bytes sent: 0.96 MB

Stage	Time	B Sent	B Recv	Client (Forwarded)	Client (Actual)	VHost	Request
R	?	?	?	?	?	?	
R	?	?	?	?	?	?	
R	?	?	?	?	?	?	
R	?	?	?	?	?	?	
S	6 ms	0 KB	0 KB	0:0:0:0:0:0:0:1	0:0:0:0:0:0:0:1	localhost	GET /manager/status?org.apache.catalina.filters.CSRF_NONCE=32EB55BE9B6A92AE76CB231C
R	?	?	?	?	?	?	

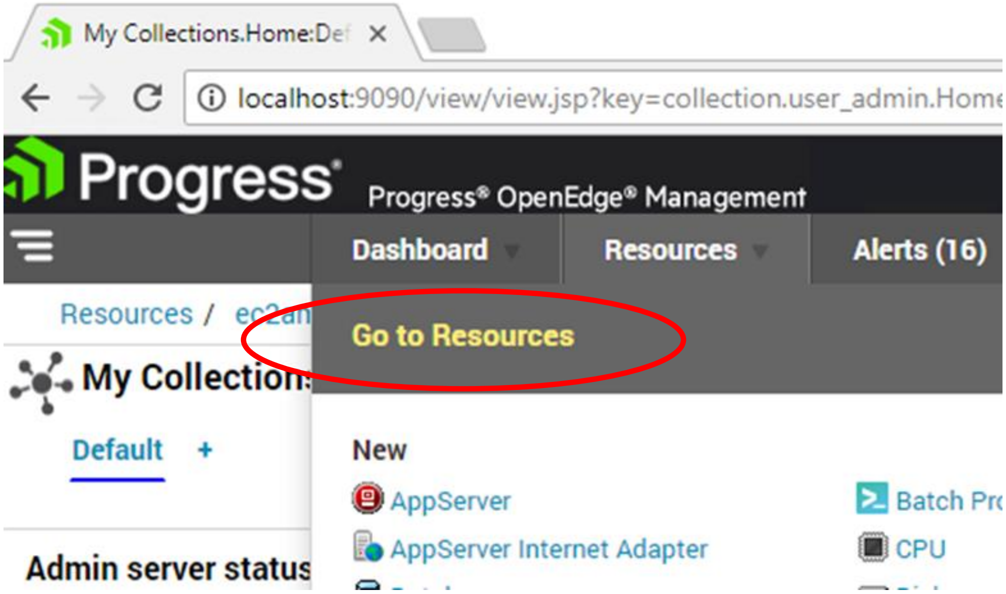
9. Start Management Console by double-clicking the icon on the desktop.



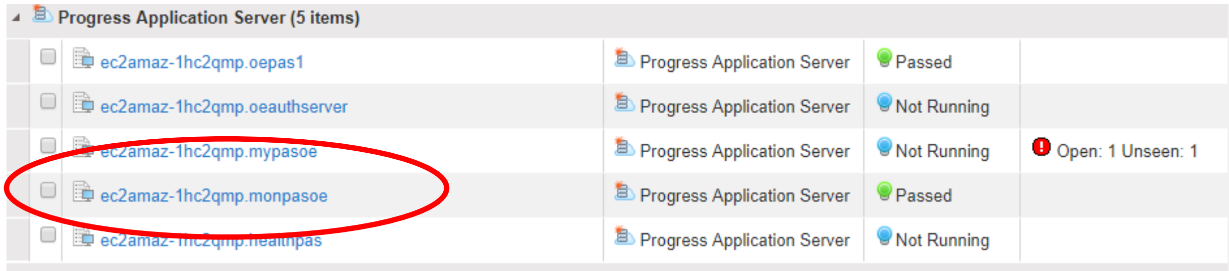
10. Login.

Username: admin
Password: 4admin

11. Select the “Resources” tab and then select “Go to Resources”.



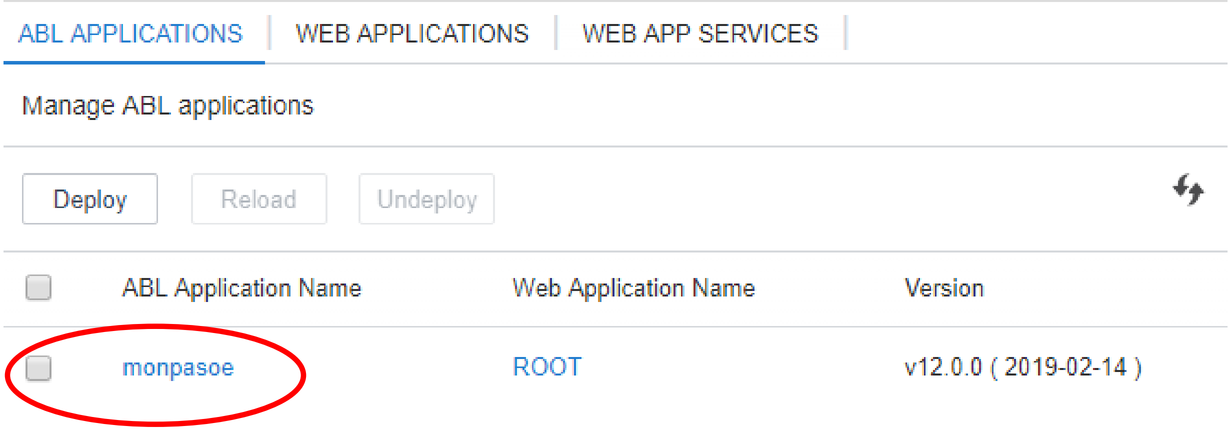
12. Scroll down in the resources and click on the “monpasoe” link.



Progress Application Server (5 items)

□	ec2amaz-1hc2qmp.oepas1	Progress Application Server	Passed	
□	ec2amaz-1hc2qmp.oeauthserver	Progress Application Server	Not Running	
□	ec2amaz-1hc2qmp.mypasoe	Progress Application Server	Not Running	Open: 1 Unseen: 1
□	ec2amaz-1hc2qmp.monpasoe	Progress Application Server	Passed	
□	ec2amaz-1hc2qmp.healthpas	Progress Application Server	Not Running	

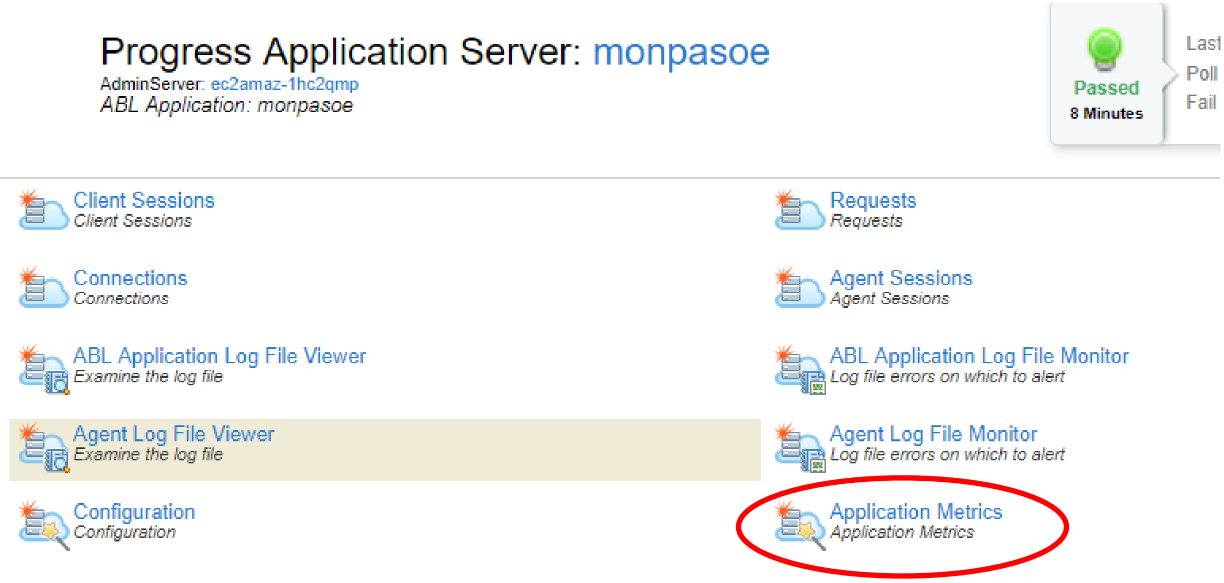
13. Select the “monpasoe” from the ABL Applications section.



Manage ABL applications

□	ABL Application Name	Web Application Name	Version
□	monpasoe	ROOT	v12.0.0 (2019-02-14)

14. Select the “Application Metrics”.



Progress Application Server: monpasoe
AdminServer: ec2amaz-1hc2qmp
ABL Application: monpasoe

Passed
8 Minutes

Last
Poll
Fail

Client Sessions
Client Sessions

Requests
Requests

Connections
Connections

Agent Sessions
Agent Sessions

ABL Application Log File Viewer
Examine the log file

ABL Application Log File Monitor
Log file errors on which to alert

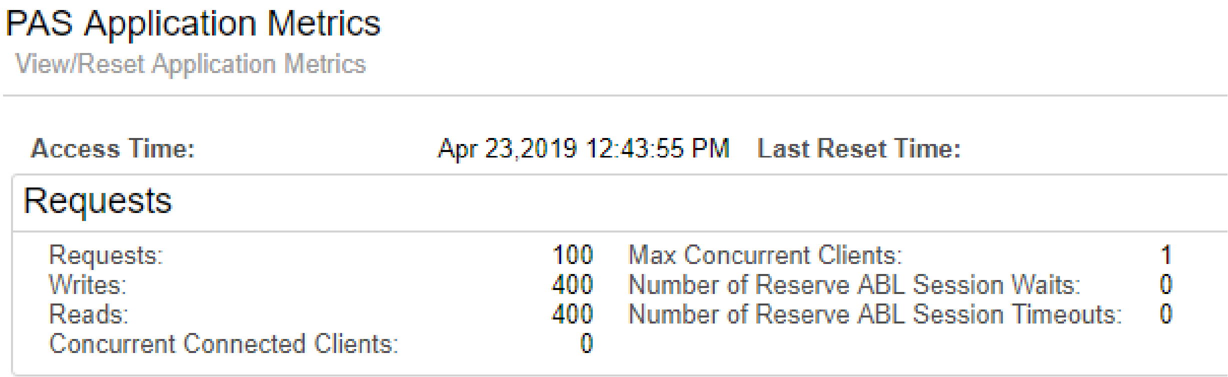
Agent Log File Viewer
Examine the log file

Agent Log File Monitor
Log file errors on which to alert

Configuration
Configuration

Application Metrics
Application Metrics

15. Be sure to notice the Requests amount (should be 100).



PAS Application Metrics
View/Reset Application Metrics

Access Time: Apr 23,2019 12:43:55 PM Last Reset Time:

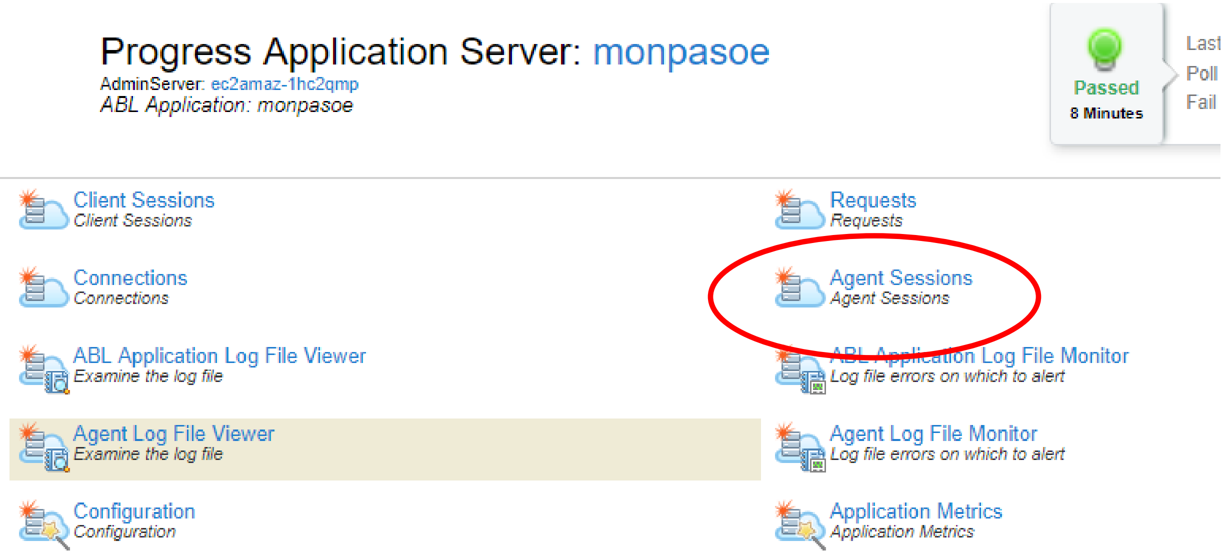
Requests

Requests:	100	Max Concurrent Clients:	1
Writes:	400	Number of Reserve ABL Session Waits:	0
Reads:	400	Number of Reserve ABL Session Timeouts:	0
Concurrent Connected Clients:	0		

16. Scroll down and view the whole page.

NOTICE: The RESET STATISTICS button and the graphs on the bottom.

17. Click the back button and then select the “Agent Sessions” link.



Progress Application Server: monpasoe

AdminServer: ec2amaz-1hc2qmp

ABL Application: monpasoe

Client Sessions
Client Sessions

Requests
Requests

Connections
Connections

Agent Sessions
Agent Sessions

ABL Application Log File Viewer
Examine the log file

ABL Application Log File Monitor
Log file errors on which to alert

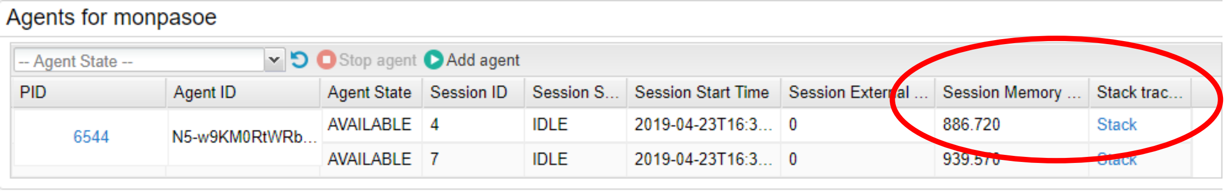
Agent Log File Viewer
Examine the log file

Agent Log File Monitor
Log file errors on which to alert

Configuration
Configuration

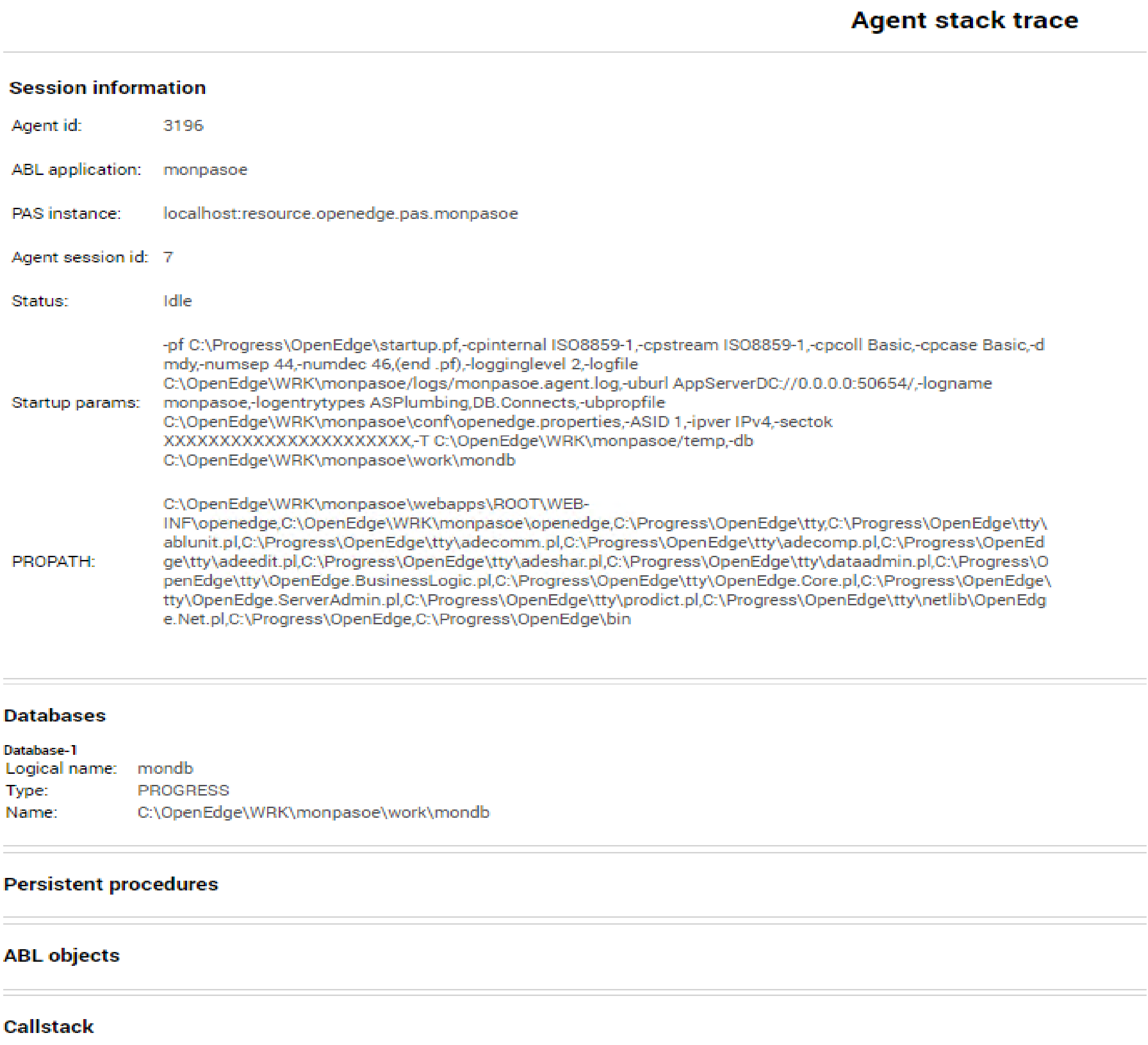
Application Metrics
Application Metrics

18. Select “Stack” trace link and view.



Agents for monpasoe

PID	Agent ID	Agent State	Session ID	Session S...	Session Start Time	Session External ...	Session Memory ...	Stack trac...
6544	N5-w9KM0RtWRb...	AVAILABLE	4	IDLE	2019-04-23T16:3...	0	886.720	Stack
		AVAILABLE	7	IDLE	2019-04-23T16:3...	0	939.570	Stack



Agent stack trace

Session information

Agent id:	3196
ABL application:	monpasoe
PAS instance:	localhost:resource.openedge.pas.monpasoe
Agent session id:	7
Status:	Idle
Startup params:	-pf C:\Progress\OpenEdge\startup.pf,-cpinternal ISO8859-1,-cpstream ISO8859-1,-cpcoll Basic,-cpcase Basic,-dmdy,-numsep 44,-numdec 46,(end .pf),-logginglevel 2,-logfile C:\OpenEdge\WRK\monpasoe/logs/monpasoe.agent.log,-uburl AppServerDC://0.0.0.0:50654/,-logname monpasoe,-logentrytypes ASPlumbing,DB.Connects,-ubpropfile C:\OpenEdge\WRK\monpasoe\conf\openedge.properties,-ASID 1,-ipver IPv4,-sectok XXXXXXXXXXXXXXXXXXXXXX,-T C:\OpenEdge\WRK\monpasoe/temp,-db C:\OpenEdge\WRK\monpasoe\work\mondb
PROPATH:	C:\OpenEdge\WRK\monpasoe\webapps\ROOT\WEB-INF\openedge,C:\OpenEdge\WRK\monpasoe\openedge,C:\Progress\OpenEdge\tty,C:\Progress\OpenEdge\tty\ablunit.pl,C:\Progress\OpenEdge\tty\adecomm.pl,C:\Progress\OpenEdge\tty\adecomp.pl,C:\Progress\OpenEdge\tty\adeedit.pl,C:\Progress\OpenEdge\tty\adeshar.pl,C:\Progress\OpenEdge\tty\dataadmin.pl,C:\Progress\OpenEdge\tty\OpenEdge.BusinessLogic.pl,C:\Progress\OpenEdge\tty\OpenEdge.Core.pl,C:\Progress\OpenEdge\tty\OpenEdge.ServerAdmin.pl,C:\Progress\OpenEdge\tty\prodict.pl,C:\Progress\OpenEdge\tty\netlib\OpenEdge.Net.pl,C:\Progress\OpenEdge,C:\Progress\OpenEdge\bin

Databases

Database-1

Logical name:	mondb
Type:	PROGRESS
Name:	C:\OpenEdge\WRK\monpasoe\work\mondb

Persistent procedures

ABL objects

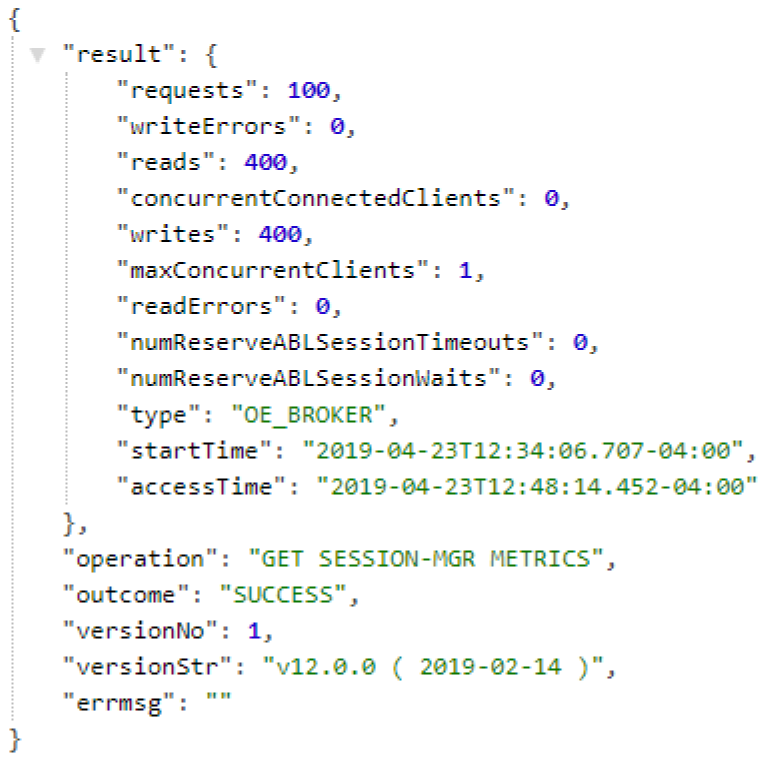
Callstack

Monitoring with oemanager REST APIs

1. Now open a new Chrome tab in the browser to use the oemanager REST APIs.

2. Use the oemanager REST API (username/password = tomcat/tomcat).

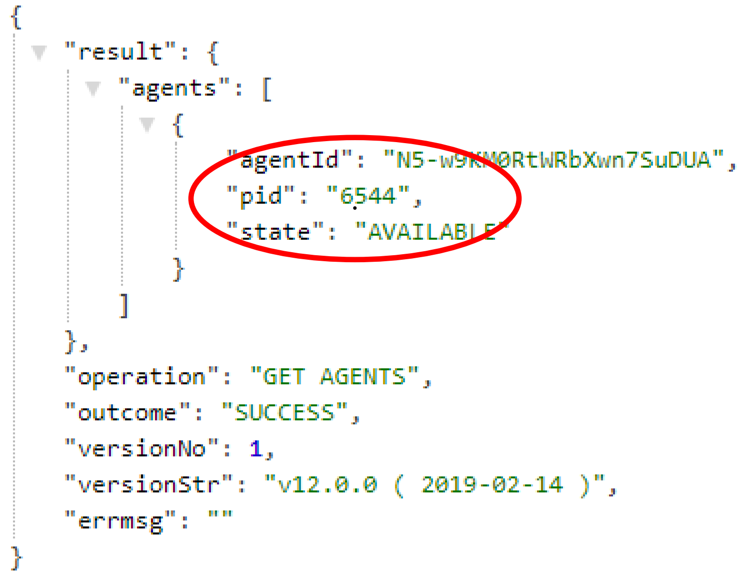
http://localhost:20000/oemanager/applications/monpasoe/metrics



```
{
    "result": {
        "requests": 100,
        "writeErrors": 0,
        "reads": 400,
        "concurrentConnectedClients": 0,
        "writes": 400,
        "maxConcurrentClients": 1,
        "readErrors": 0,
        "numReserveABLSessionTimeouts": 0,
        "numReserveABLSessionWaits": 0,
        "type": "OE_BROKER",
        "startTime": "2019-04-23T12:34:06.707-04:00",
        "accessTime": "2019-04-23T12:48:14.452-04:00"
    },
    "operation": "GET SESSION-MGR METRICS",
    "outcome": "SUCCESS",
    "versionNo": 1,
    "versionStr": "v12.0.0 ( 2019-02-14 )",
    "errmsg": ""
}
```

3. Now get the “Agent” PID or “agentId” (save PID for later, you will need it).

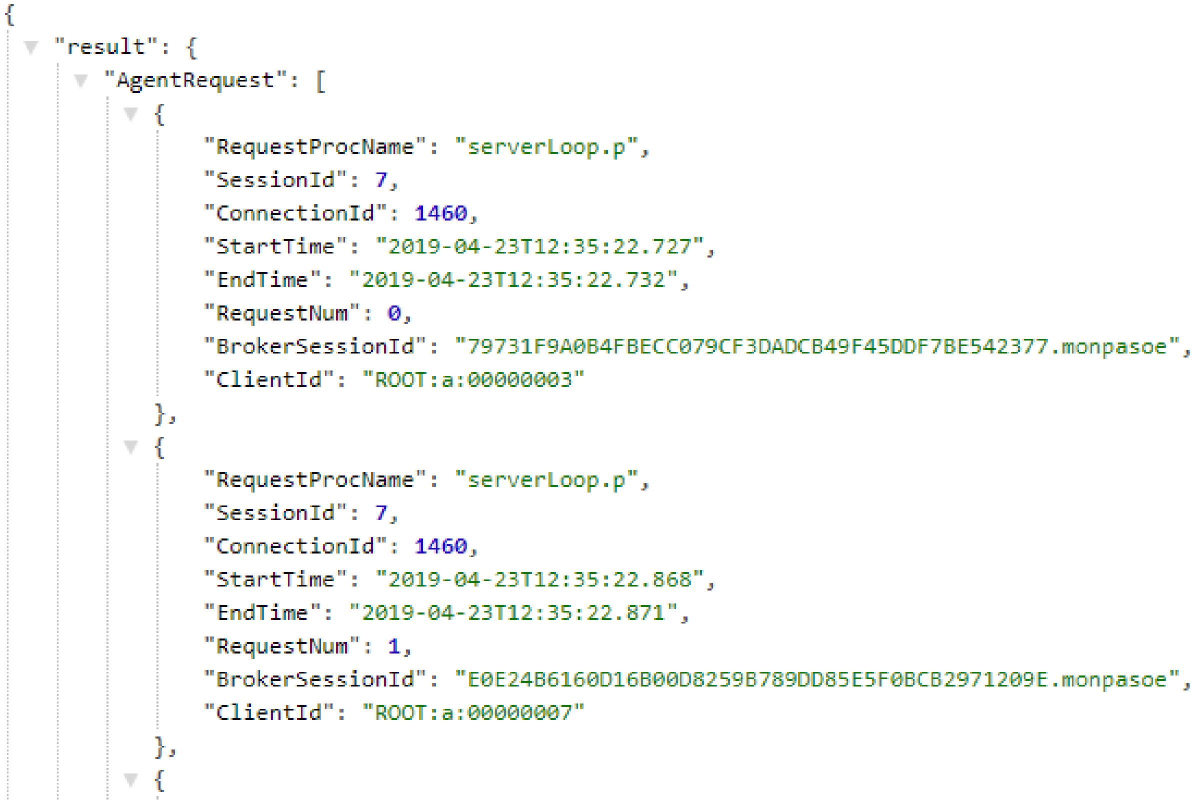
http://localhost:20000/oemanager/applications/monpasoe/agents



```
{
  "result": {
    "agents": [
      {
        "agentId": "N5-w9KM0RtWRbXwn7SuDUA",
        "pid": "6544",
        "state": "AVAILABLE"
      }
    ]
  },
  "operation": "GET AGENTS",
  "outcome": "SUCCESS",
  "versionNo": 1,
  "versionStr": "v12.0.0 ( 2019-02-14 )",
  "errmsg": ""
}
```

4. Now get requests using the agentId or pid value (use your PID from #3).

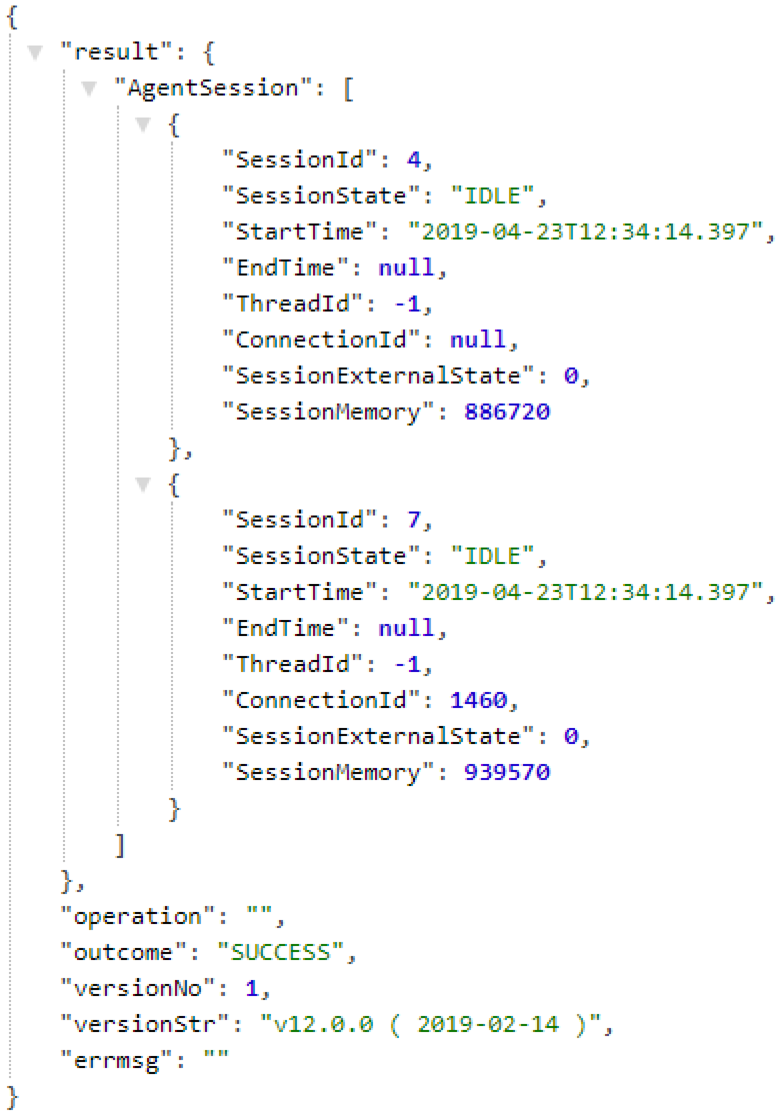
http://localhost:20000/oemanager/applications/monpasoe/agents/{PID}/requests



```
{
  "result": {
    "AgentRequest": [
      {
        "RequestProcName": "serverLoop.p",
        "SessionId": 7,
        "ConnectionId": 1460,
        "StartTime": "2019-04-23T12:35:22.727",
        "EndTime": "2019-04-23T12:35:22.732",
        "RequestNum": 0,
        "BrokerSessionId": "79731F9A0B4FBECC079CF3DADCB49F45DDF7BE542377.monpasoe",
        "ClientId": "ROOT:a:00000003"
      },
      {
        "RequestProcName": "serverLoop.p",
        "SessionId": 7,
        "ConnectionId": 1460,
        "StartTime": "2019-04-23T12:35:22.868",
        "EndTime": "2019-04-23T12:35:22.871",
        "RequestNum": 1,
        "BrokerSessionId": "E0E24B6160D16B00D8259B789DD85E5F0BCB2971209E.monpasoe",
        "ClientId": "ROOT:a:00000007"
      },
      {
```

5. Now get the agent’s sessions.

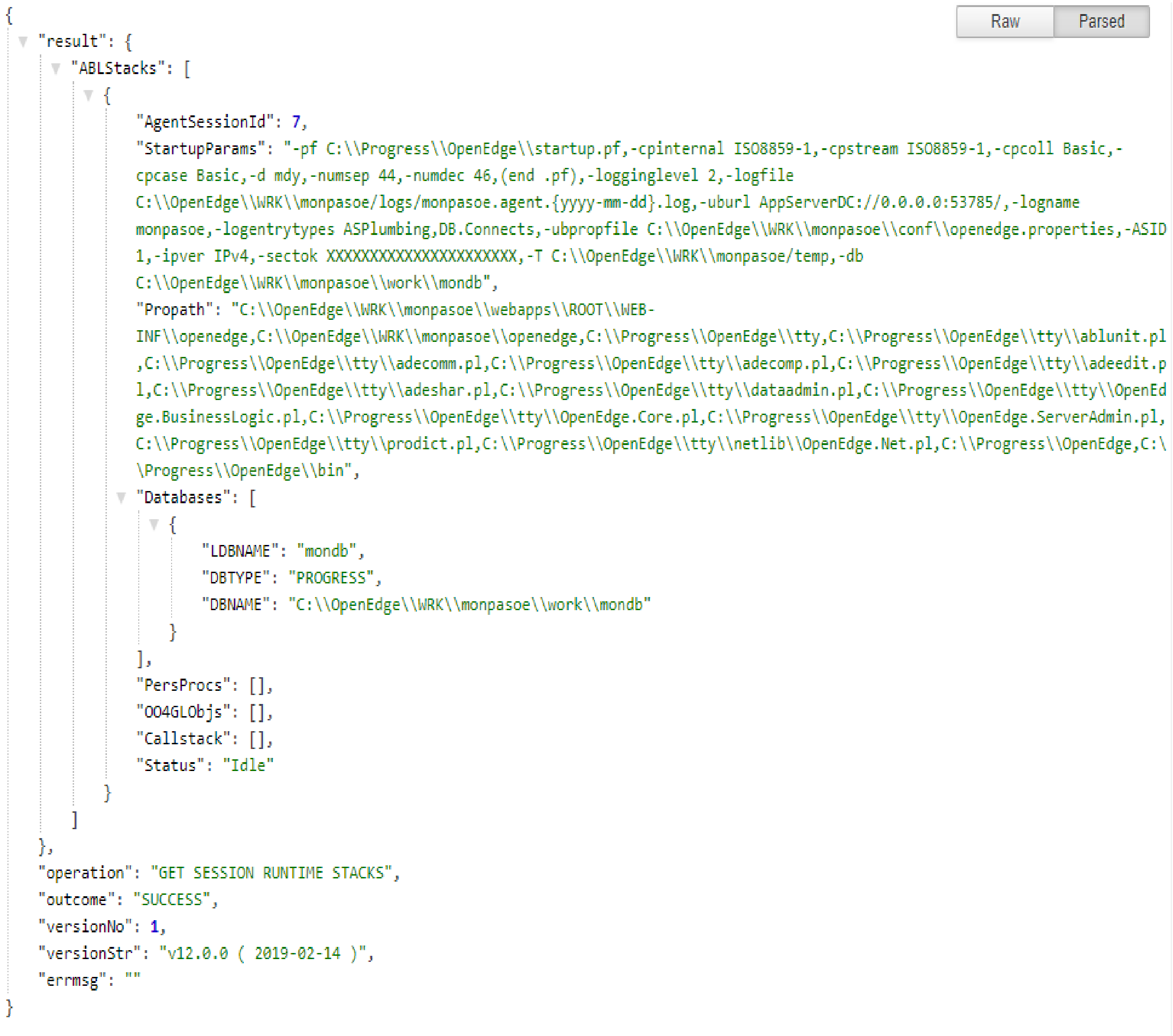
http://localhost:20000/oemanager/applications/monpasoe/agents/{PID}/sessions



```
{
    "result": {
        "AgentSession": [
            {
                "SessionId": 4,
                "SessionState": "IDLE",
                "StartTime": "2019-04-23T12:34:14.397",
                "EndTime": null,
                "ThreadId": -1,
                "ConnectionId": null,
                "SessionExternalState": 0,
                "SessionMemory": 886720
            },
            {
                "SessionId": 7,
                "SessionState": "IDLE",
                "StartTime": "2019-04-23T12:34:14.397",
                "EndTime": null,
                "ThreadId": -1,
                "ConnectionId": 1460,
                "SessionExternalState": 0,
                "SessionMemory": 939570
            }
        ]
    },
    "operation": "",
    "outcome": "SUCCESS",
    "versionNo": 1,
    "versionStr": "v12.0.0 ( 2019-02-14 )",
    "errmsg": ""
}
```

6. Now get a stack trace from a session (session 7 in the example below).

http://localhost:20000/oemanager/applications/monpasoe/agents/{PID}/sessions/7/stacks



```
{
    "result": {
        "ABLStacks": [
            {
                "AgentSessionId": 7,
                "StartupParams": "-pf C:\\Progress\\OpenEdge\\startup.pf,-cpinternal ISO8859-1,-cpstream ISO8859-1,-cpcoll Basic,-
                cpcase Basic,-d mdy,-numsep 44,-numdec 46,(end .pf),-logginglevel 2,-logfile
                C:\\OpenEdge\\WRK\\monpasoe/logs/monpasoe.agent.{yyyy-mm-dd}.log,-uburl AppServerDC://0.0.0.0:53785/,-logname
                monpasoe,-logentrytypes ASPlumbing,DB.Connects,-ubpropfile C:\\OpenEdge\\WRK\\monpasoe\\conf\\openedge.properties,-ASID
                1,-ipver IPv4,-sectok XXXXXXXXXXXXXXXXXXXXXX,-T C:\\OpenEdge\\WRK\\monpasoe/temp,-db
                C:\\OpenEdge\\WRK\\monpasoe\\work\\mondb",
                "Propath": "C:\\OpenEdge\\WRK\\monpasoe\\webapps\\ROOT\\WEB-
                INF\\openedge,C:\\OpenEdge\\WRK\\monpasoe\\openedge,C:\\Progress\\OpenEdge\\tty,C:\\Progress\\OpenEdge\\tty\\ablunit.pl
                ,C:\\Progress\\OpenEdge\\tty\\adecomm.pl,C:\\Progress\\OpenEdge\\tty\\adecomp.pl,C:\\Progress\\OpenEdge\\tty\\adeedit.p
                l,C:\\Progress\\OpenEdge\\tty\\adeshar.pl,C:\\Progress\\OpenEdge\\tty\\dataadmin.pl,C:\\Progress\\OpenEdge\\tty\\OpenEd
                ge.BusinessLogic.pl,C:\\Progress\\OpenEdge\\tty\\OpenEdge.Core.pl,C:\\Progress\\OpenEdge\\tty\\OpenEdge.ServerAdmin.pl,
                C:\\Progress\\OpenEdge\\tty\\prodict.pl,C:\\Progress\\OpenEdge\\tty\\netlib\\OpenEdge.Net.pl,C:\\Progress\\OpenEdge,C:\
                \Progress\\OpenEdge\\bin",
                "Databases": [
                    {
                        "LDBNAME": "mondb",
                        "DBTYPE": "PROGRESS",
                        "DBNAME": "C:\\OpenEdge\\WRK\\monpasoe\\work\\mondb"
                    }
                ],
                "PersProcs": [],
                "OO4GLObjs": [],
                "Callstack": [],
                "Status": "Idle"
            }
        ]
    },
    "operation": "GET SESSION RUNTIME STACKS",
    "outcome": "SUCCESS",
    "versionNo": 1,
    "versionStr": "v12.0.0 ( 2019-02-14 )",
    "errmsg": ""
}
```

Monitoring your production PAS for OpenEdge with oejmx

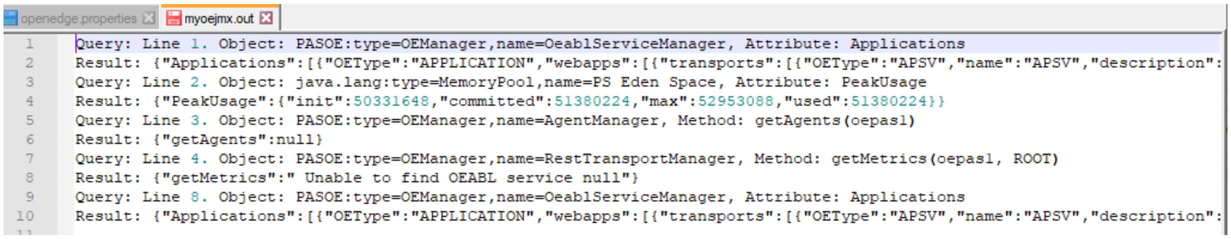
1. The oejmx utility is in your instance bin directory.

```
proenv> C:\OpenEdge\WRK\monpasoe\bin\oejmx.bat
```

2. Use the default query file for this example.

```
proenv> cd C:\OpenEdge\WRK\monpasoe
proenv> bin\oejmx.bat -Q bin\jmxqueries\default.qry -o
work\myoejmx.out
```

3. Open the C:\OpenEdge\WRK\monpasoe\work\myoejmx.out with notepad ++.



```
Query: Line 1. Object: PASOE:type=OEManager,name=OeablServiceManager, Attribute: Applications
Result: {"Applications":[{"OEType":"APPLICATION","webapps":[{"transports":[{"OEType":"APSV","name":"APSV","description":
Query: Line 2. Object: java.lang:type=MemoryPool,name=PS Eden Space, Attribute: PeakUsage
Result: {"PeakUsage":{"init":50331648,"committed":51380224,"max":52953088,"used":51380224}}
Query: Line 3. Object: PASOE:type=OEManager,name=AgentManager, Method: getAgents(oepas1)
Result: {"getAgents":null}
Query: Line 4. Object: PASOE:type=OEManager,name=RestTransportManager, Method: getMetrics(oepas1, ROOT)
Result: {"getMetrics":" Unable to find OEABL service null"}
Query: Line 8. Object: PASOE:type=OEManager,name=OeablServiceManager, Attribute: Applications
Result: {"Applications":[{"OEType":"APPLICATION","webapps":[{"transports":[{"OEType":"APSV","name":"APSV","description":
```

Query: Line 1 → Tomcat Heap Memory Usage
Query: Line 2 → Tomcat Total Memory Usage
Query: Line 3 → Gets agents
Query: Line 4 → Gets metrics for the REST transport
Query: Line 8 → Lists the states of all the transports

NOTE: in real life, you may need to make several calls to get the information you want. You may need to make a call to get the agentId or PID, then modify the file_name.qry with that value, then run to get the agent data.

For more information about OEJMX, see the OpenEdge documentation: https://documentation.progress.com/output/oe117sp/index.html#page/gssp4/use-oejmx-to-manage-and-monitor-an-instance.html

---

Section summary

There are several ways to monitor PAS for OpenEdge.

The point is to monitor PAS for OpenEdge!

Tips and Tricks for Implementing the OpenEdge Application Server

LAB 4

Looking for Memory Leaks

Overview

This section covers looking for dynamic objects in ABL code and reclaiming system memory.

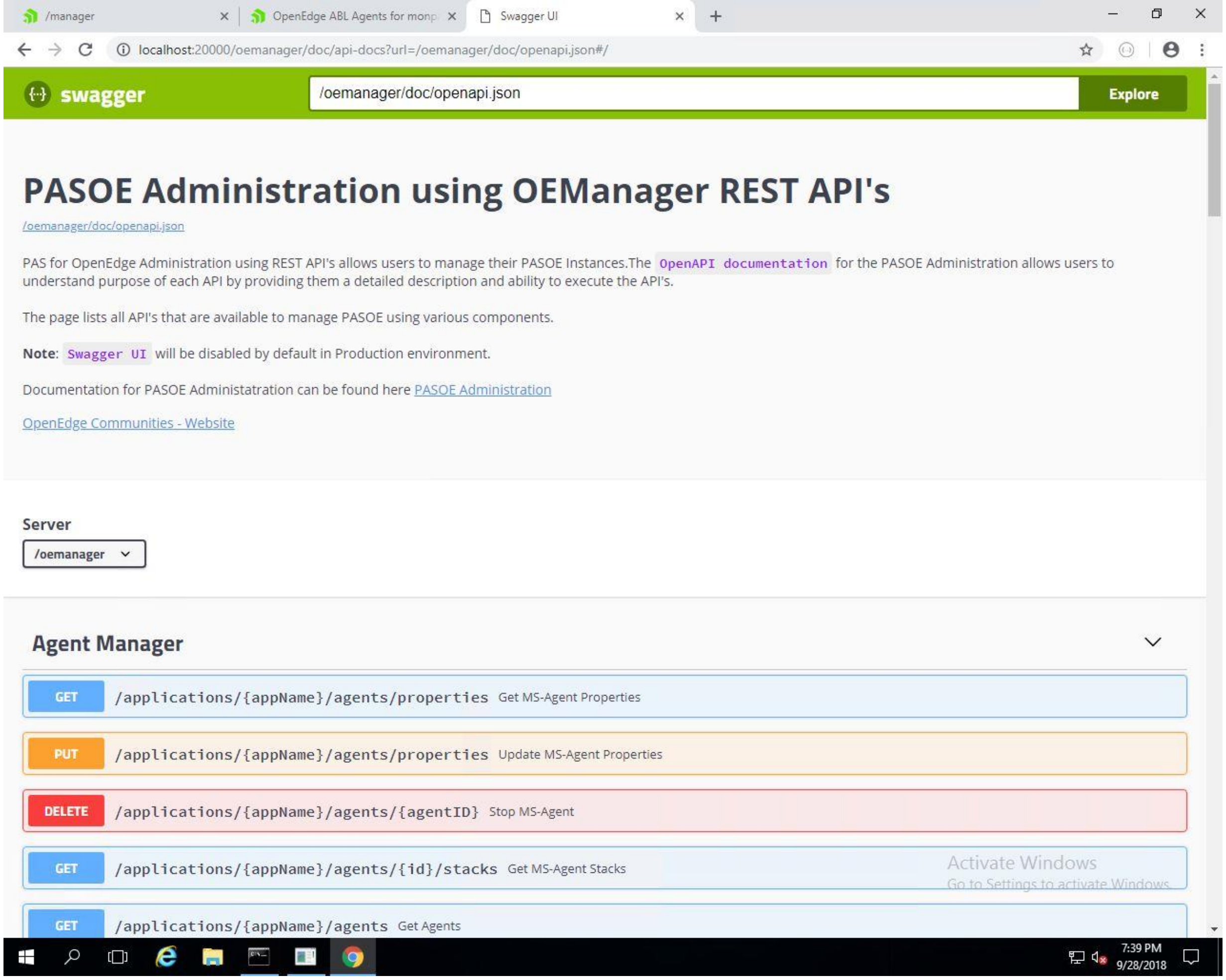
Running ABLObject Tracking

Use “monpasoe” to gather leaking memory objects. For this exercise we will use the new Swagger interface.

1. In the chrome browser, go to the main oemanager URL

http://localhost:20000/oemanager/

This will show the new Swagger administration page


/manager
OpenEdge ABL Agents for monp
Swagger UI
localhost:20000/oemanager/doc/api-docs?url=/oemanager/doc/openapi.json#/
swagger
/oemanager/doc/openapi.json
Explore

PASOE Administration using OEManager REST API's

/oemanager/doc/openapi.json

PAS for OpenEdge Administration using REST API's allows users to manage their PASOE Instances.The `OpenAPI documentation` for the PASOE Administration allows users to understand purpose of each API by providing them a detailed description and ability to execute the API's.

The page lists all API's that are available to manage PASOE using various components.

Note: `Swagger UI` will be disabled by default in Production environment.

Documentation for PASOE Administatration can be found here PASOE Administration

OpenEdge Communities - Website

Server

/oemanager

Agent Manager

GET `/applications/{appName}/agents/properties` Get MS-Agent Properties

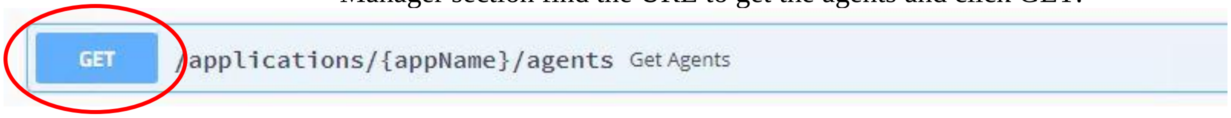
PUT `/applications/{appName}/agents/properties` Update MS-Agent Properties

DELETE `/applications/{appName}/agents/{agentID}` Stop MS-Agent

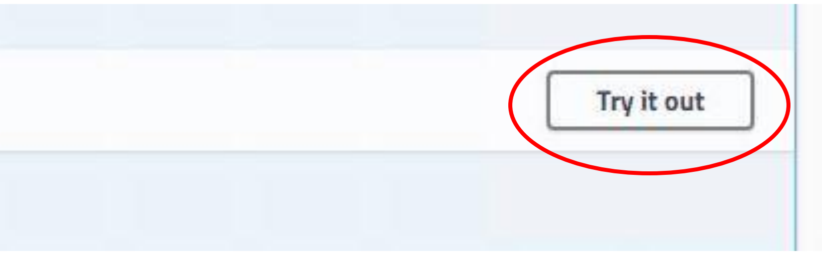
GET `/applications/{appName}/agents/{id}/stacks` Get MS-Agent Stacks

GET `/applications/{appName}/agents` Get Agents

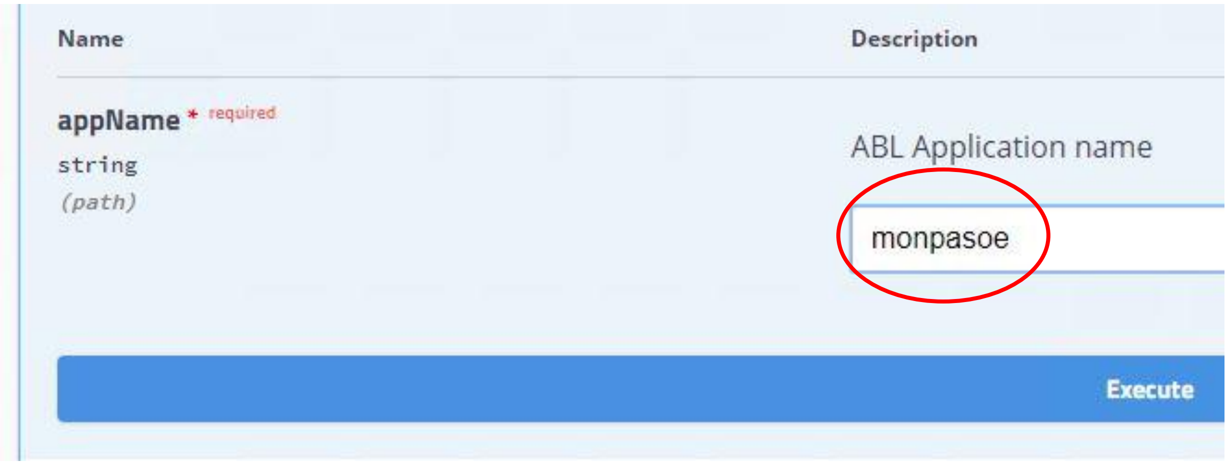
2. Get the agent ID (if you've forgotten it from before). Under the Agent Manager section find the URL to get the agents and click GET.



Then click "Try it out."

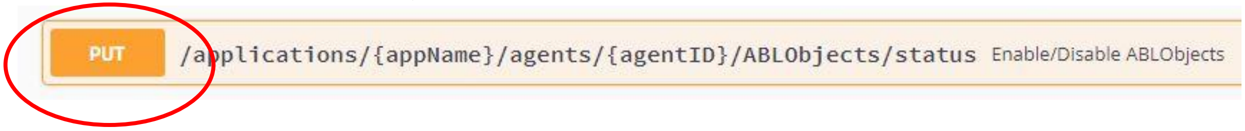


Then add the application name "monpasoe" and click "Execute."

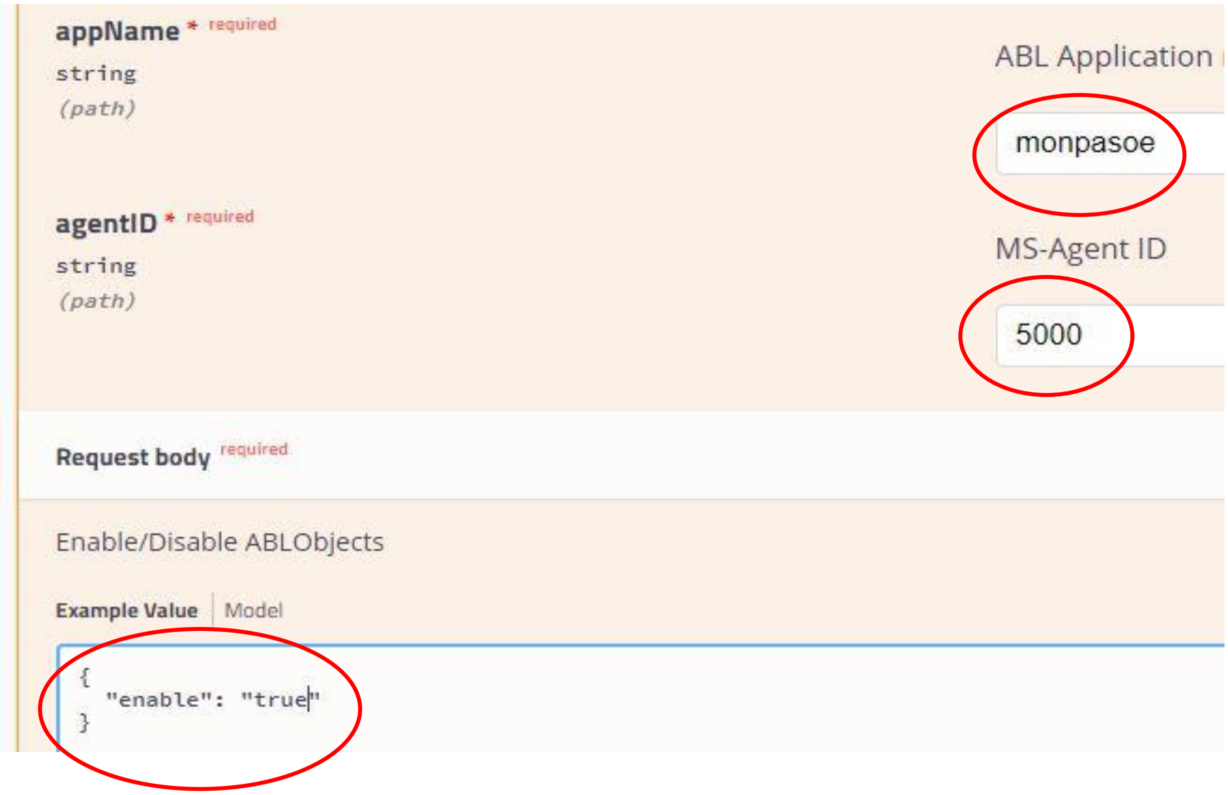


You should see your agent PID in the "Response body" section.

3. Turn on ABLObject tracking (use the PID from #2). Under the Agent Manager section, find the URL to turn on ABL object tracking and click PUT.


PUT
/applications/{appName}/agents/{agentID}/ABLObjects/status
Enable/Disable ABLObjects

Next click “Try it out.”
Then add the application name “monpasoe”, fill in the agent ID, add the request body { “enable” : “true” } and click “Execute.”



appName * required
string
(path)

ABL Application

monpasoe

agentID * required
string
(path)

MS-Agent ID

5000

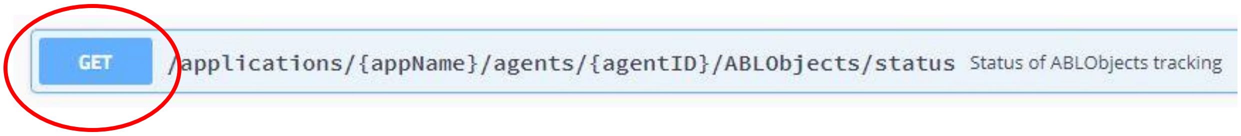
Request body required

Enable/Disable ABLObjects

Example Value | Model

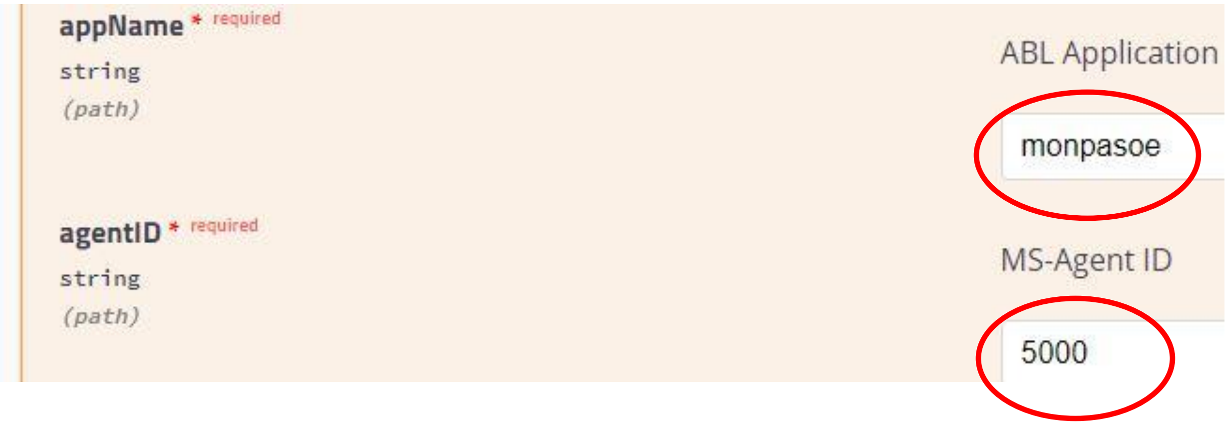
```
{
  "enable": "true"
}
```

4. Verify that we are tracking ABLObjects. Under the Agent Manager section, find the URL to get ABL Object tracking status and click GET.



GET /applications/{appName}/agents/{agentID}/ABLObjects/status Status of ABLObjects tracking

Next click “Try it out.”
As above, add the application name “monpasoe” and fill in the agent ID.
Click “Execute.”



appName * required
string
(path)

ABL Application

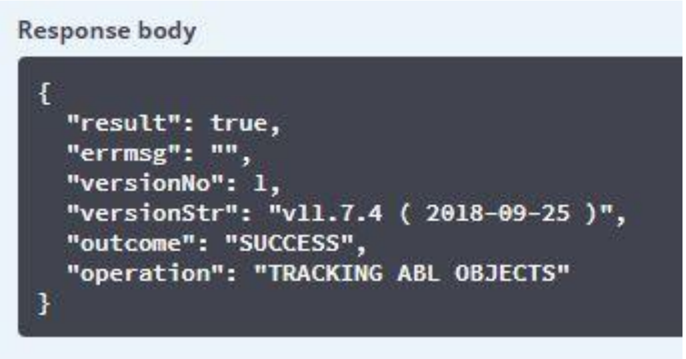
monpasoe

agentID * required
string
(path)

MS-Agent ID

5000

You should see a SUCCESS response.

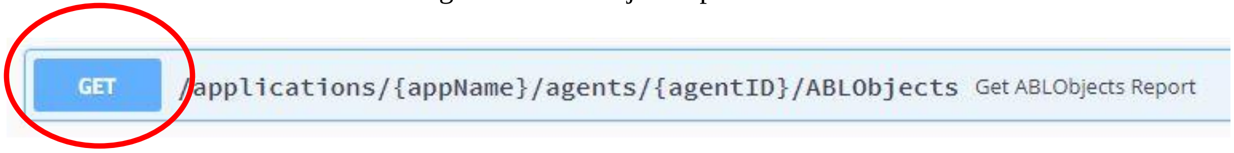


```
{
  "result": true,
  "errmsg": "",
  "versionNo": 1,
  "versionStr": "v11.7.4 ( 2018-09-25 )",
  "outcome": "SUCCESS",
  "operation": "TRACKING ABL OBJECTS"
}
```

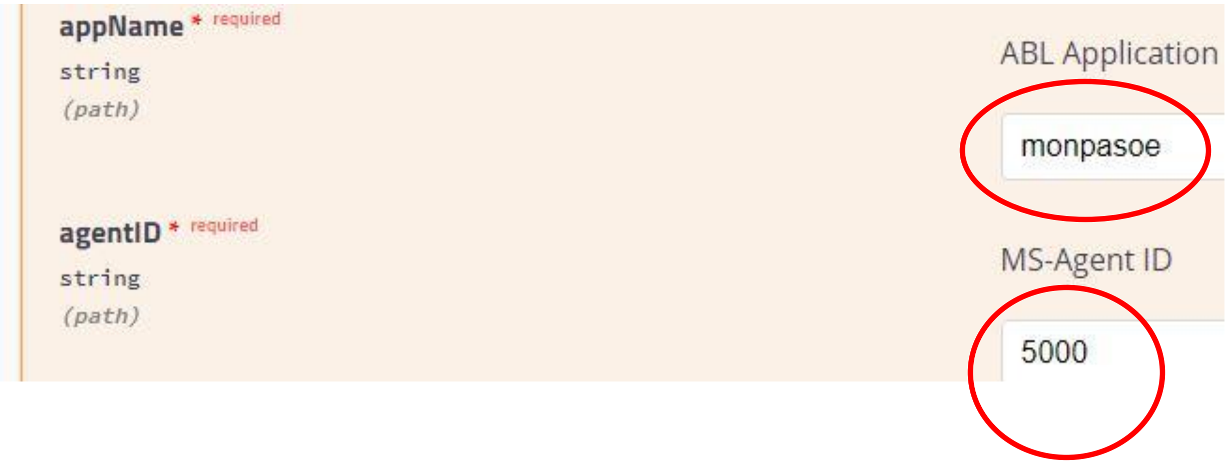
5. Run code that leaks mem-pointers and handles.

```
proenv>cd %WRKDIR%
proenv>prowin -b -p call-leak-code.p
```

6. Get the ABL Object Report. Under the Agent Manager section, find the URL to get the ABL Object report and click GET



Next click “Try it out.”
As above, add the application name “monpasoe” and fill in the agent ID.
Click “Execute.”

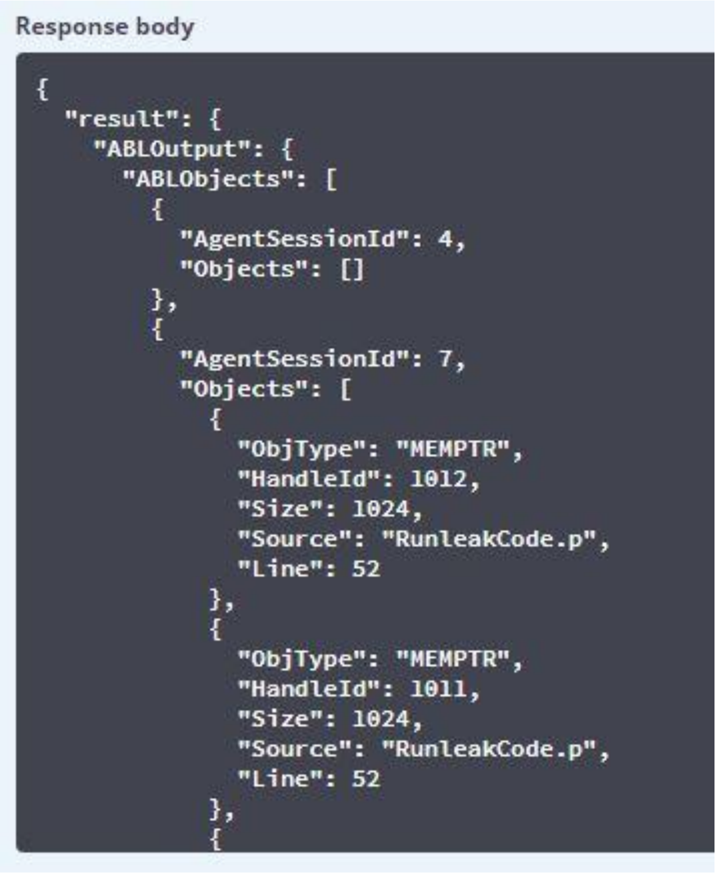


appName * required
string
(path)
ABL Application
monpasoe

agentID * required
string
(path)
MS-Agent ID
5000

The response should be a report like the one below:

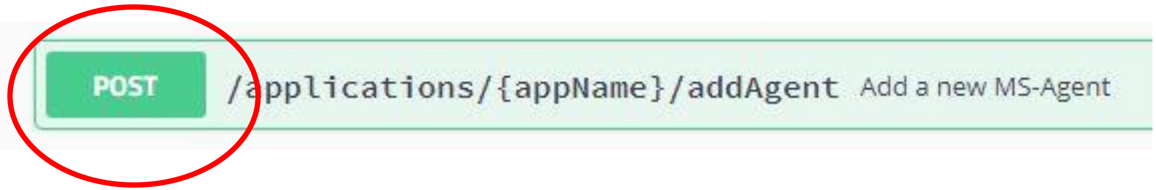
- Should show MEMPTR leaks
- Should show DATASET leaks



Response body

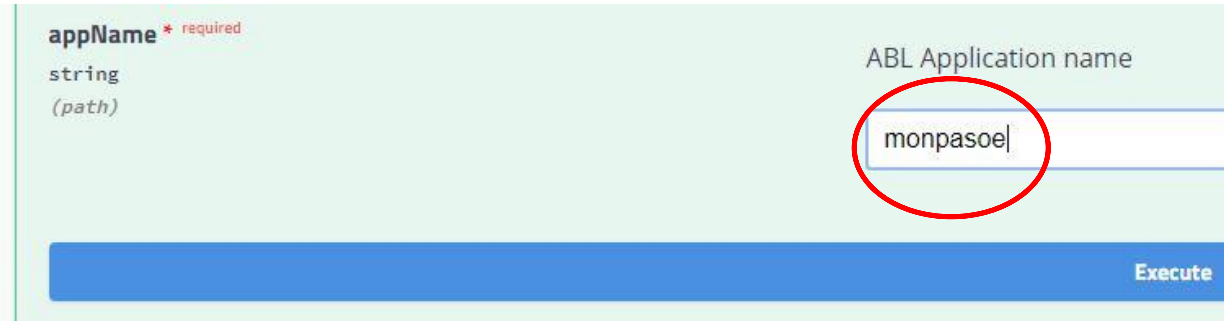
```
{
  "result": {
    "ABLOutput": {
      "ABLObjects": [
        {
          "AgentSessionId": 4,
          "Objects": []
        },
        {
          "AgentSessionId": 7,
          "Objects": [
            {
              "ObjType": "MEMPTR",
              "HandleId": 1012,
              "Size": 1024,
              "Source": "RunleakCode.p",
              "Line": 52
            },
            {
              "ObjType": "MEMPTR",
              "HandleId": 1011,
              "Size": 1024,
              "Source": "RunleakCode.p",
              "Line": 52
            },
            {
```

7. Manually add a new agent before stopping the current one. Under the Session Manager Administration section, find the URL to add an agent and click POST.



/applications/{appName}/addAgent Add a new MS-Agent

Next click “Try it out.”
Add the application name “monpasoe”.
Click “Execute.”

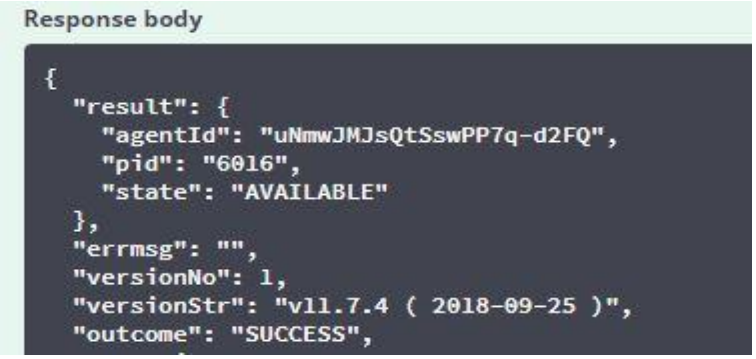


appName * required
string
(path)

ABL Application name

monpasoe

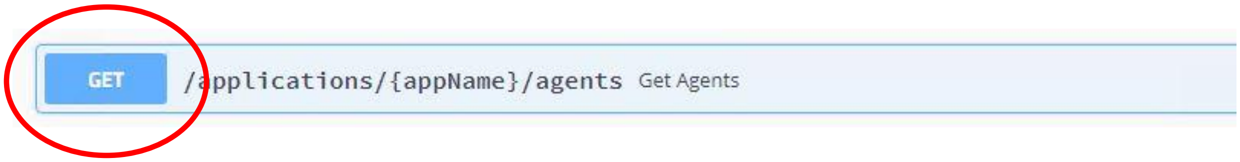
You should then see a response body that looks like:


Response body

```
{
  "result": {
    "agentId": "uNmwJMJsQtSswPP7q-d2FQ",
    "pid": "6016",
    "state": "AVAILABLE"
  },
  "errmsg": "",
  "versionNo": 1,
  "versionStr": "v11.7.4 ( 2018-09-25 )",
  "outcome": "SUCCESS",
```

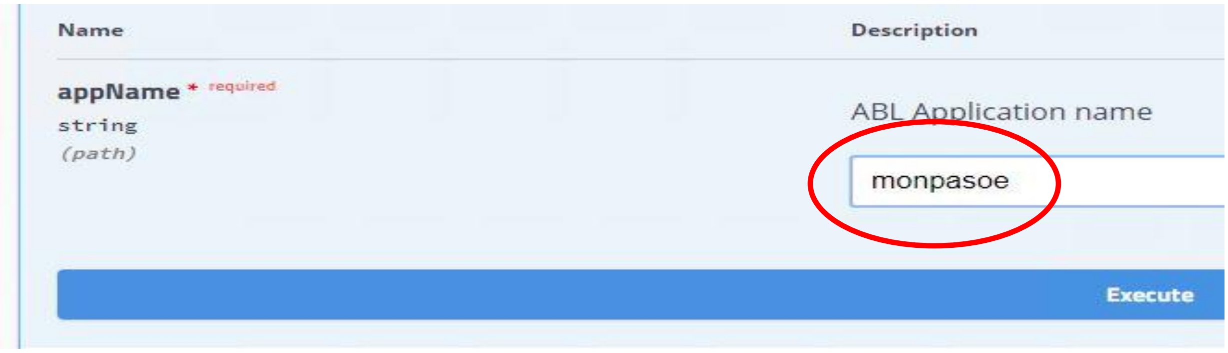
The result will include agentId and PID of the new agent.

8. To verify that you now have two running agents, find the agents URL under the Agent Manager section and click GET.



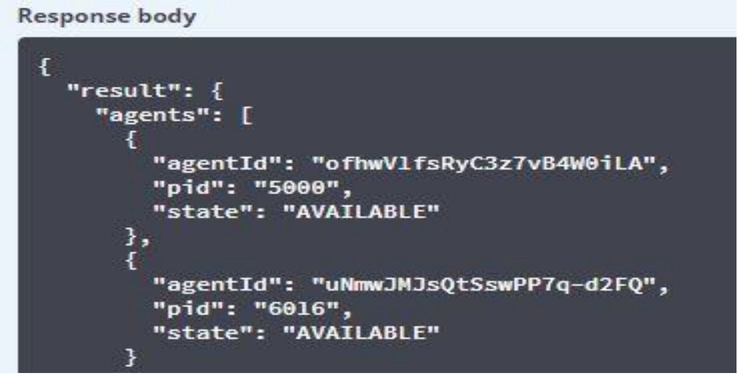
```
/applications/{appName}/agents  Get Agents
```

Then click “Try it out.”
Then add the application name “monpasoe” and click “Execute.”



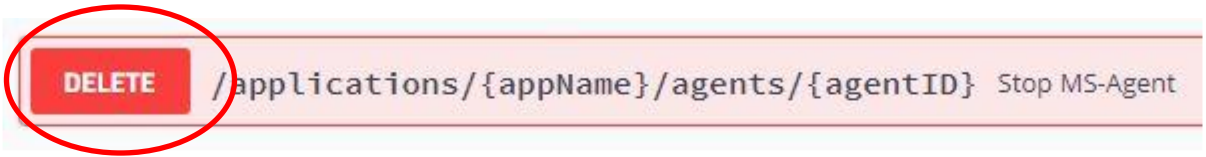
Name	Description
appName * required string (path)	ABL Application name monpasoe

You should see your new agent PID in the “Response body” section

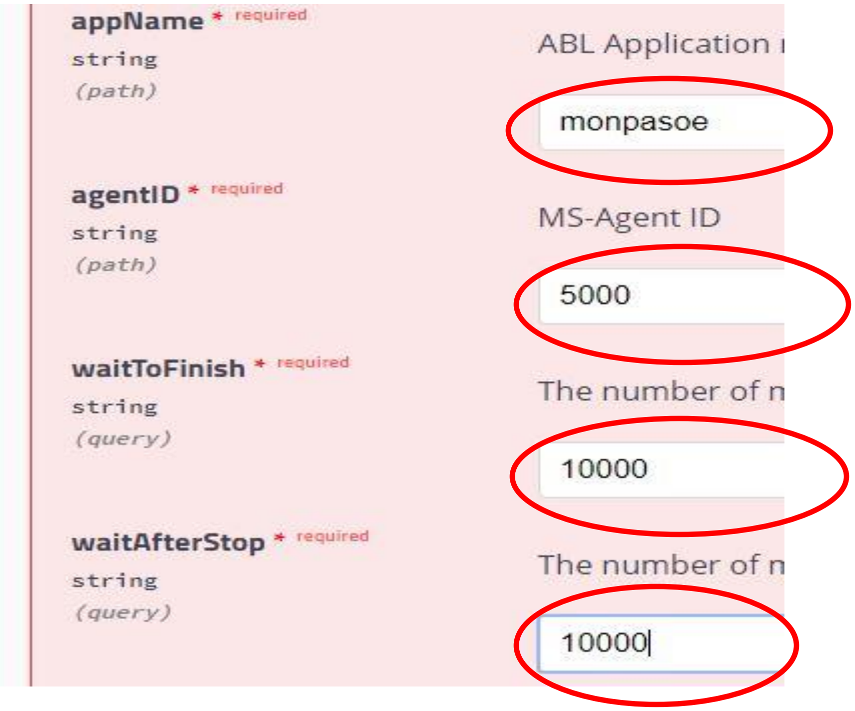

Response body

```
{
  "result": {
    "agents": [
      {
        "agentId": "ofhwVlfsRyC3z7vB4W0iLA",
        "pid": "5000",
        "state": "AVAILABLE"
      },
      {
        "agentId": "uNmwJMJsQtSswPP7q-d2FQ",
        "pid": "6016",
        "state": "AVAILABLE"
      }
```

9. To stop an agent, under the Agent Manager section find the URL for stopping agents and click DELETE.



Then click “Try it out.”
Then add the application name “monpasoe” and the PID of your *old* agent. Use 10 seconds for waitToFinish and 10 seconds for waitAfterStop so that the agent will terminate in about 20 seconds.
Click “Execute.”



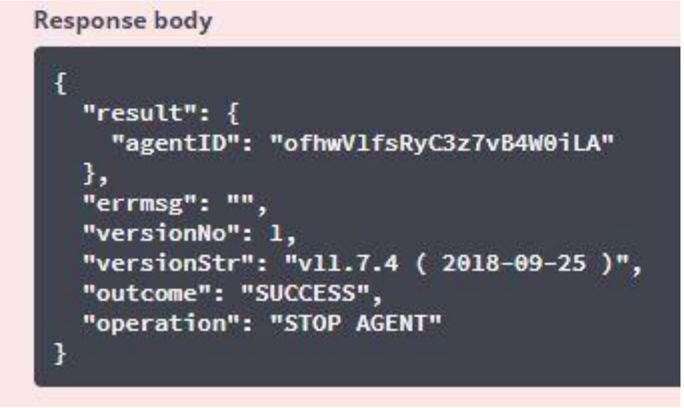
appName * required
string
(path)
ABL Application
monpasoe

agentID * required
string
(path)
MS-Agent ID
5000

waitToFinish * required
string
(query)
The number of n
10000

waitAfterStop * required
string
(query)
The number of n
10000

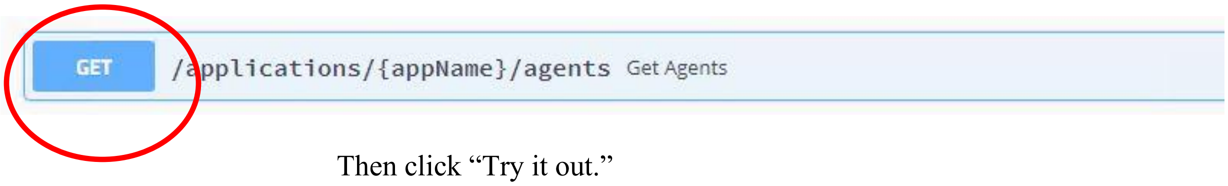
You should see a SUCCESS result in the response body:



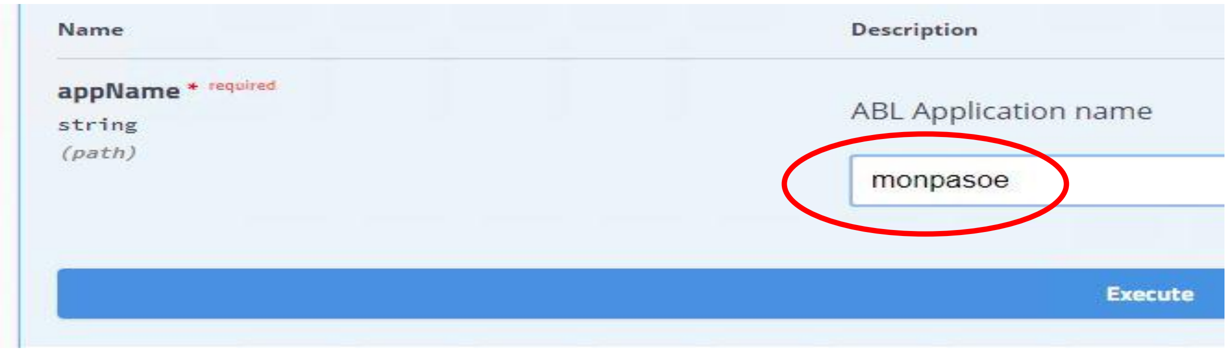
Response body

```
{
  "result": {
    "agentID": "ofhwVlfsRyC3z7vB4W0iLA"
  },
  "errmsg": "",
  "versionNo": 1,
  "versionStr": "v11.7.4 ( 2018-09-25 )",
  "outcome": "SUCCESS",
  "operation": "STOP AGENT"
}
```

10. Verify that the agent is terminated by finding the agents URL under the Agent Manager section and clicking GET.

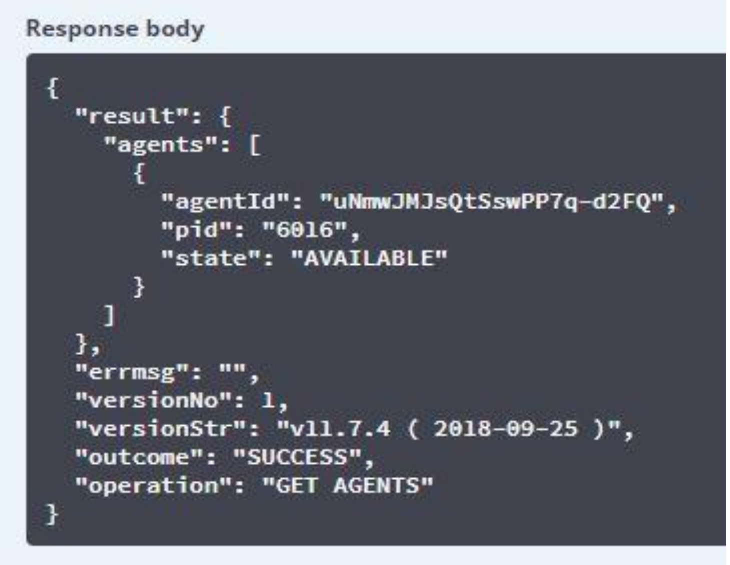


Then click “Try it out.”
Then add the application name “monpasoe” and click “Execute.”



appName: monpasoe

In the response body you will notice that the agent is gone.



```
{
  "result": {
    "agents": [
      {
        "agentId": "uNmwJMJsQtSswPP7q-d2FQ",
        "pid": "6016",
        "state": "AVAILABLE"
      }
    ]
  },
  "errmsg": "",
  "versionNo": 1,
  "versionStr": "v11.7.4 ( 2018-09-25 )",
  "outcome": "SUCCESS",
  "operation": "GET AGENTS"
}
```

Section summary

With the ability to track ABL Objects you can find memory leaks in production and with the ability to manually add and remove agents you can keep your production system up and running while cycling out misbehaving agents.

Tips and Tricks for Implementing the OpenEdge Application Server

LAB 5

Deferred Logging

Overview

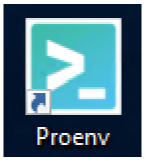
This section covers enabling and demonstrating the new Deferred Logging feature available in 12.1.

This feature allows for the dumping of detailed logging in the case of an msagent crash or by a REST API/oejmx request.

Enabling Deferred Logging

By default, Deferred Logging is not enabled

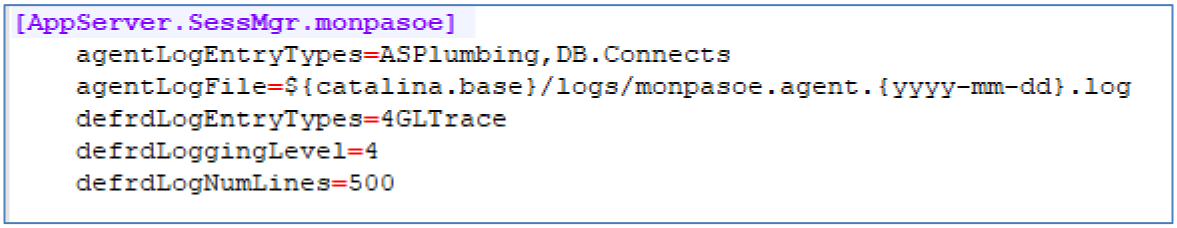
1. We could modify the openedge.properties manually but for this lab we will use the oeprop command.



2. Use oeprop to change the default setting

```
   proenv>cd %WRKDIR%
   proenv>cd monpasoe
   proenv>bin\oeprop.bat
+AppServer.SessMgr.monpasoe.defrdLogNumLines=500
   proenv>bin\oeprop.bat
+AppServer.SessMgr.monpasoe.defrdLoggingLevel=4
   proenv>bin\oeprop.bat
+AppServer.SessMgr.monpasoe.defrdLogEntryTypes=4GLTrace
```

This will add these values in the openedge.properties.



```
[AppServer.SessMgr.monpasoe]
    agentLogEntryTypes=ASPlumbing,DB.Connects
    agentLogFile=${catalina.base}/logs/monpasoe.agent.{yyyy-mm-dd}.log
    defrdLogEntryTypes=4GLTrace
    defrdLoggingLevel=4
    defrdLogNumLines=500
```

3. The monpasoe instance must be restarted for the changes to take effect.

NOTE: if “allowRuntimeUpdates=1” was set you could change this logging, and any other logging, values without having to restart.

```
proenv>cd %WRKDIR%
proenv>cd monpasoe
proenv>bin\tcman.bat pasoestart -restart
```

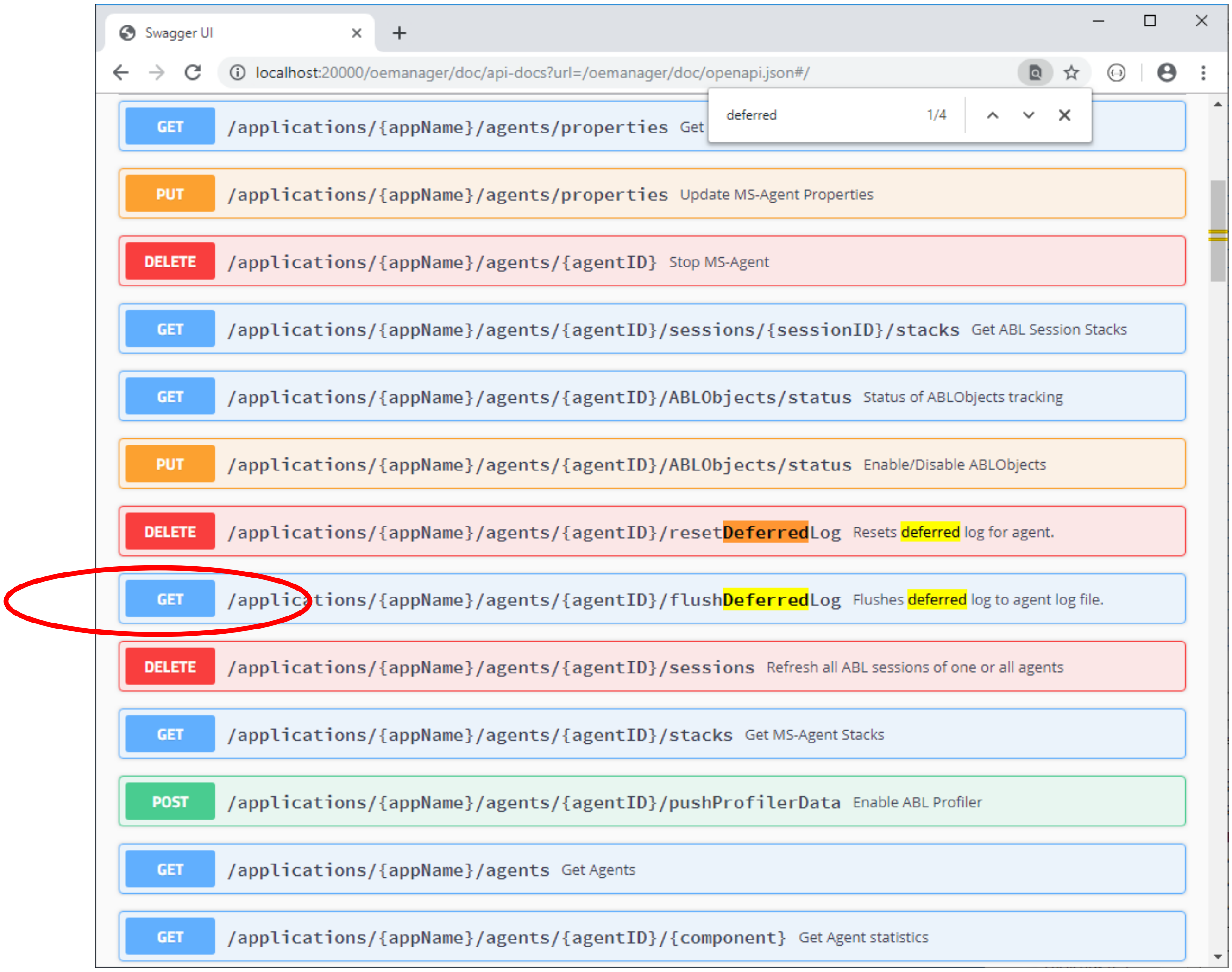
4. Run client code again

```
proenv> cd C:\OpenEdge\WRK
proenv> prowin -b -p clientLoop.p
```

5. Use Notepad++ to open the C:\OpenEdge\WRK\monpasoe\logs\monpasoe.agent.2019-##-##.log file.
6. Notice that there is very little logging, just connect and disconnect, even though we ran 100 requests.
7. Leave the agent file open in notepad++.
8. Now we will force the deferred logging to dump into the agent log file. Go to the swagger page for PASOE.

REMINDER: Open http://localhost:2000 and select the MANAGE PAS FOR OPENEGE link.

9. Search for “flushDeferredLog” and click on GET


Swagger UI
localhost:20000/oemanager/doc/api-docs?url=/oemanager/doc/openapi.json#/
deferred
1/4

GET /applications/{appName}/agents/properties Get

PUT /applications/{appName}/agents/properties Update MS-Agent Properties

DELETE /applications/{appName}/agents/{agentID} Stop MS-Agent

GET /applications/{appName}/agents/{agentID}/sessions/{sessionID}/stacks Get ABL Session Stacks

GET /applications/{appName}/agents/{agentID}/ABLObjects/status Status of ABLObjects tracking

PUT /applications/{appName}/agents/{agentID}/ABLObjects/status Enable/Disable ABLObjects

DELETE /applications/{appName}/agents/{agentID}/resetDeferredLog Resets deferred log for agent.

GET /applications/{appName}/agents/{agentID}/flushDeferredLog Flushes deferred log to agent log file.

DELETE /applications/{appName}/agents/{agentID}/sessions Refresh all ABL sessions of one or all agents

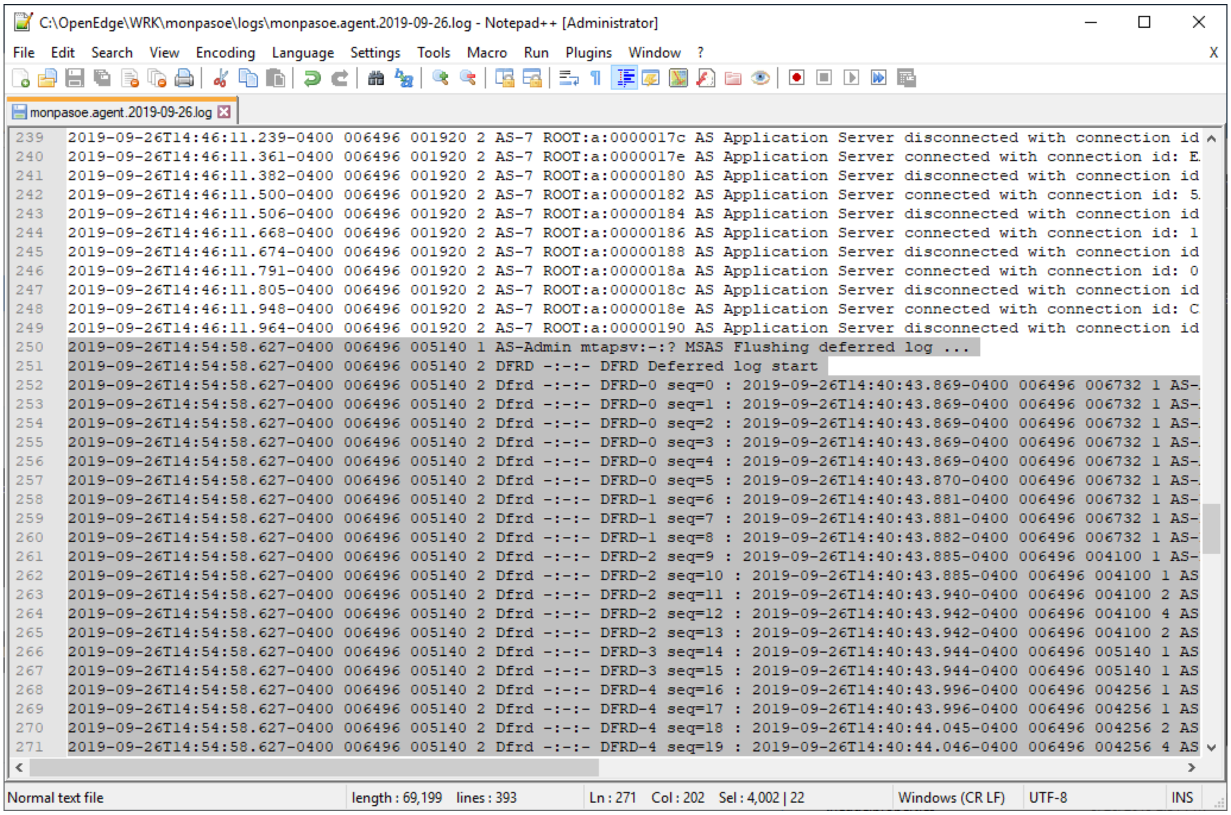
GET /applications/{appName}/agents/{agentID}/stacks Get MS-Agent Stacks

POST /applications/{appName}/agents/{agentID}/pushProfilerData Enable ABL Profiler

GET /applications/{appName}/agents Get Agents

GET /applications/{appName}/agents/{agentID}/{component} Get Agent statistics

10. Select [Try it out]
11. Enter appName: monpasoe
12. Enter agentID: all
13. Select Execute
14. Notice in the Notepad++ that the agent log was updated. Click yes to reload
15. See all the DFRD log messages with 4GLTrace



C:\OpenEdge\WRK\monpasoe\logs\monpasoe.agent.2019-09-26.log - Notepad++ [Administrator]
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
monpasoe.agent.2019-09-26.log
Normal text file
length : 69,199 lines : 393
Ln : 271 Col : 202 Sel : 4,002 | 22
Windows (CR LF)
UTF-8
INS

```
2019-09-26T14:46:11.239-0400 006496 001920 2 AS-7 ROOT:a:0000017c AS Application Server disconnected with connection id
2019-09-26T14:46:11.361-0400 006496 001920 2 AS-7 ROOT:a:0000017e AS Application Server connected with connection id: E
2019-09-26T14:46:11.382-0400 006496 001920 2 AS-7 ROOT:a:00000180 AS Application Server disconnected with connection id
2019-09-26T14:46:11.500-0400 006496 001920 2 AS-7 ROOT:a:00000182 AS Application Server connected with connection id: 5
2019-09-26T14:46:11.506-0400 006496 001920 2 AS-7 ROOT:a:00000184 AS Application Server disconnected with connection id
2019-09-26T14:46:11.668-0400 006496 001920 2 AS-7 ROOT:a:00000186 AS Application Server connected with connection id: 1
2019-09-26T14:46:11.674-0400 006496 001920 2 AS-7 ROOT:a:00000188 AS Application Server disconnected with connection id
2019-09-26T14:46:11.791-0400 006496 001920 2 AS-7 ROOT:a:0000018a AS Application Server connected with connection id: 0
2019-09-26T14:46:11.805-0400 006496 001920 2 AS-7 ROOT:a:0000018c AS Application Server disconnected with connection id
2019-09-26T14:46:11.948-0400 006496 001920 2 AS-7 ROOT:a:0000018e AS Application Server connected with connection id: C
2019-09-26T14:46:11.964-0400 006496 001920 2 AS-7 ROOT:a:00000190 AS Application Server disconnected with connection id
2019-09-26T14:54:58.627-0400 006496 005140 1 AS-Admin mtapsv:-:? MSAS Flushing deferred log ...
2019-09-26T14:54:58.627-0400 006496 005140 2 DFRD -:-:- DFRD Deferred log start
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-0 seq=0 : 2019-09-26T14:40:43.869-0400 006496 006732 1 AS-
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-0 seq=1 : 2019-09-26T14:40:43.869-0400 006496 006732 1 AS-
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-0 seq=2 : 2019-09-26T14:40:43.869-0400 006496 006732 1 AS-
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-0 seq=3 : 2019-09-26T14:40:43.869-0400 006496 006732 1 AS-
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-0 seq=4 : 2019-09-26T14:40:43.869-0400 006496 006732 1 AS-
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-0 seq=5 : 2019-09-26T14:40:43.870-0400 006496 006732 1 AS-
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-1 seq=6 : 2019-09-26T14:40:43.881-0400 006496 006732 1 AS-
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-1 seq=7 : 2019-09-26T14:40:43.881-0400 006496 006732 1 AS-
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-1 seq=8 : 2019-09-26T14:40:43.882-0400 006496 006732 1 AS-
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-2 seq=9 : 2019-09-26T14:40:43.885-0400 006496 004100 1 AS-
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-2 seq=10 : 2019-09-26T14:40:43.885-0400 006496 004100 1 AS
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-2 seq=11 : 2019-09-26T14:40:43.940-0400 006496 004100 2 AS
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-2 seq=12 : 2019-09-26T14:40:43.942-0400 006496 004100 4 AS
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-2 seq=13 : 2019-09-26T14:40:43.942-0400 006496 004100 2 AS
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-3 seq=14 : 2019-09-26T14:40:43.944-0400 006496 005140 1 AS
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-3 seq=15 : 2019-09-26T14:40:43.944-0400 006496 005140 1 AS
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-4 seq=16 : 2019-09-26T14:40:43.996-0400 006496 004256 1 AS
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-4 seq=17 : 2019-09-26T14:40:43.996-0400 006496 004256 1 AS
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-4 seq=18 : 2019-09-26T14:40:44.045-0400 006496 004256 2 AS
2019-09-26T14:54:58.627-0400 006496 005140 2 Dfrd -:-:- DFRD-4 seq=19 : 2019-09-26T14:40:44.046-0400 006496 004256 4 AS
```

Section summary

Deferred logging can give the extra logging only when you need it. Like when the agent crashes or you are debugging an issue and flush the logging manually.

Tips and Tricks for Implementing the OpenEdge Application Server

LAB 6

Enable OE HealthScanner

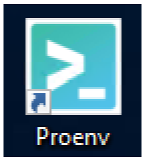
Overview

This section covers enabling and using the OE HealthScanner for PASOE available in 12.0.

Enabling OE HealthScanner Data Collection

By default, the OE HealthScanner data collection is turned off.

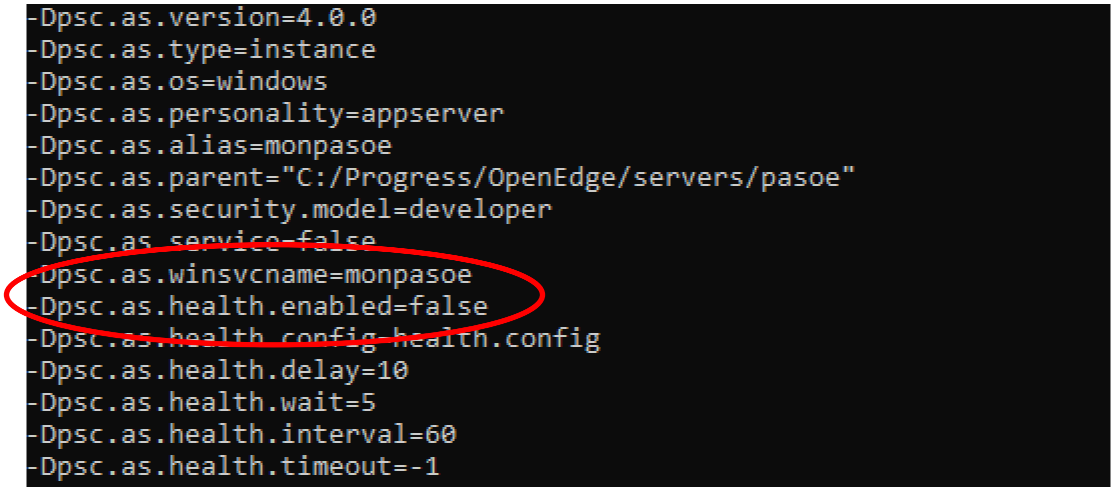
1. To turn on OE HealthScanner data collection, open the proenv window.



2. Use the tcman config command to enable data collection.

```
proenv>cd %WRKDIR%
proenv>cd monpasoe
proenv>bin\tcman.bat config
```

This will list all the configurable settings. You can find what you want to change and use it in this command

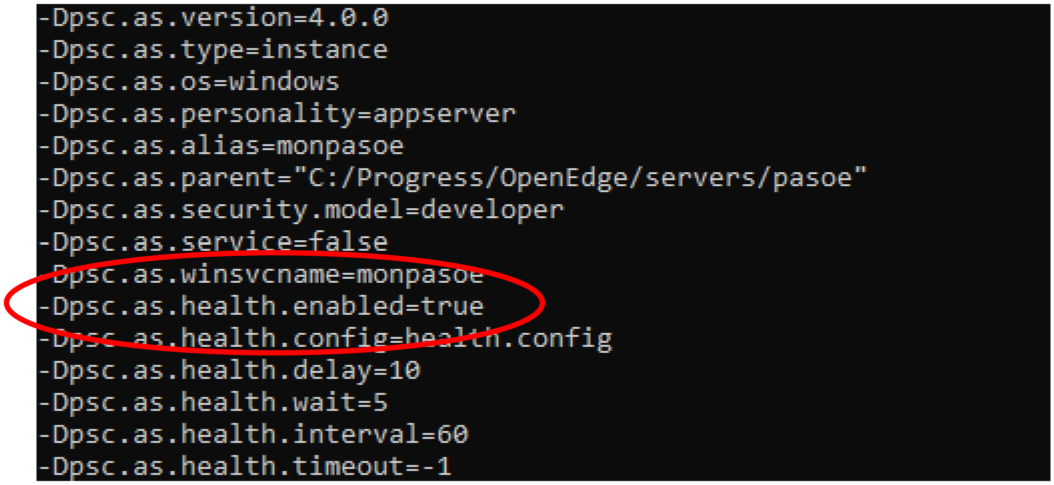


```
-Dpsc.as.version=4.0.0
-Dpsc.as.type=instance
-Dpsc.as.os=windows
-Dpsc.as.personality=appserver
-Dpsc.as.alias=monpasoe
-Dpsc.as.parent="C:/Progress/OpenEdge/servers/pasoe"
-Dpsc.as.security.model=developer
-Dpsc.as.service=false
-Dpsc.as.winsvcname=monpasoe
-Dpsc.as.health.enabled=false
-Dpsc.as.health.config=health.config
-Dpsc.as.health.delay=10
-Dpsc.as.health.wait=5
-Dpsc.as.health.interval=60
-Dpsc.as.health.timeout=-1
```

```
proenv>bin\tcman.bat config psc.as.health.enabled=true
```

Running the config command again should show the health enabled now true

```
proenv>bin\tcman.bat config
```



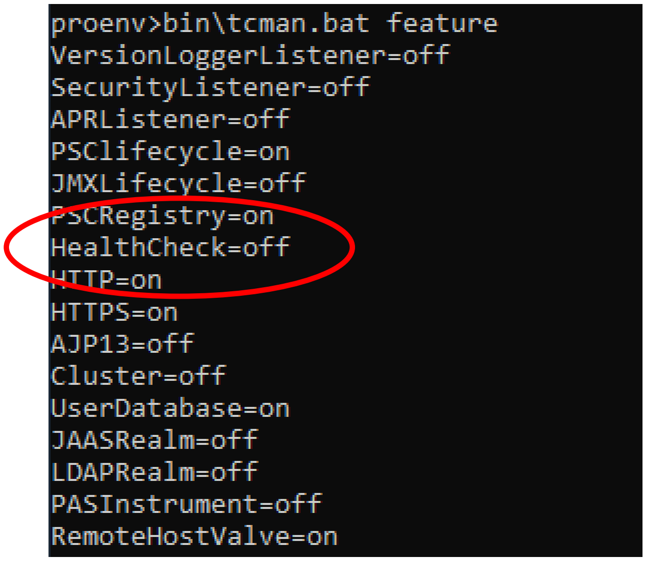
```
-Dpsc.as.version=4.0.0
-Dpsc.as.type=instance
-Dpsc.as.os=windows
-Dpsc.as.personality=appserver
-Dpsc.as.alias=monpasoe
-Dpsc.as.parent="C:/Progress/OpenEdge/servers/pasoe"
-Dpsc.as.security.model=developer
-Dpsc.as.service=false
-Dpsc.as.winsvcname=monpasoe
-Dpsc.as.health.enabled=true
-Dpsc.as.health.config=health.config
-Dpsc.as.health.delay=10
-Dpsc.as.health.wait=5
-Dpsc.as.health.interval=60
-Dpsc.as.health.timeout=-1
```

Turn on the web application that serves HTTP/S requests to the OE HealthScanner

3. To turn on web application use the tcman feature command

```
proenv>bin\tcman.bat feature
```

This will list the features which can be turned off or on and their current status.



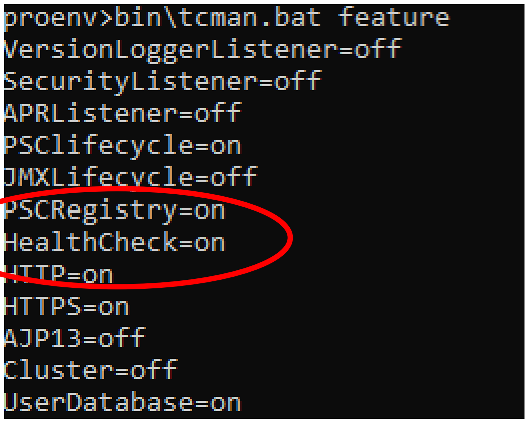
```
proenv>bin\tcman.bat feature
VersionLoggerListener=off
SecurityListener=off
APRListener=off
PSClifecycle=on
JMXLifecycle=off
PSCRegistry=on
HealthCheck=off
HTTP=on
HTTPS=on
AJP13=off
Cluster=off
UserDatabase=on
JAASRealm=off
LDAPRealm=off
PASInstrument=off
RemoteHostValve=on
```

4. Turn on the HealthCheck feature

```
proenv>bin\tcman.bat feature HealthCheck=on
```

5. Verify the HealthCheck in now “on”

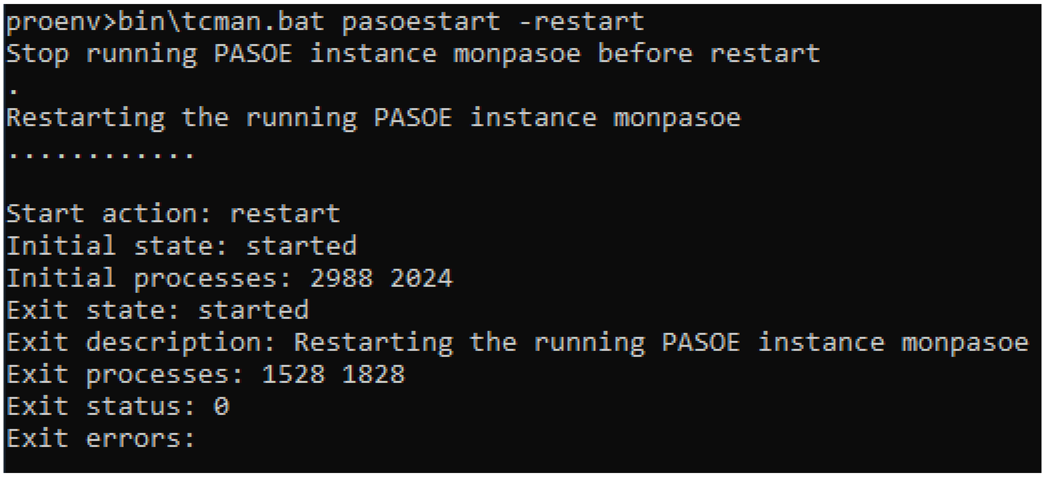
```
proenv>bin\tcman.bat feature
```



```
proenv>bin\tcman.bat feature
VersionLoggerListener=off
SecurityListener=off
APRListener=off
PSClifecycle=on
JMXLifecycle=off
PSCRegistry=on
HealthCheck=on
HTTP=on
HTTPS=on
AJP13=off
Cluster=off
UserDatabase=on
```

6. Restart monpasoe

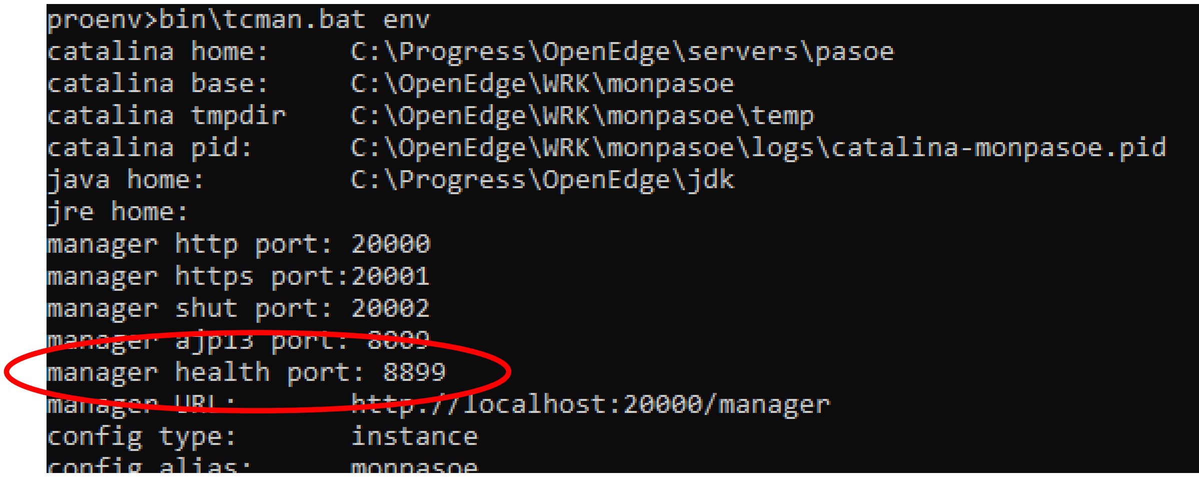
```
proenv>bin\tcman.bat pasoestart -restart
```



```
proenv>bin\tcman.bat pasoestart -restart
Stop running PASOE instance monpasoe before restart
.
Restarting the running PASOE instance monpasoe
............

Start action: restart
Initial state: started
Initial processes: 2988 2024
Exit state: started
Exit description: Restarting the running PASOE instance monpasoe
Exit processes: 1528 1828
Exit status: 0
Exit errors:
```

7. Confirm the OE HealthScanner is running with tcman env

```
proenv>bin\tcman.bat env
```



```
proenv>bin\tcman.bat env
catalina home:     C:\Progress\OpenEdge\servers\pasoe
catalina base:     C:\OpenEdge\WRK\monpasoe
catalina tmpdir    C:\OpenEdge\WRK\monpasoe\temp
catalina pid:      C:\OpenEdge\WRK\monpasoe\logs\catalina-monpasoe.pid
java home:         C:\Progress\OpenEdge\jdk
jre home:
manager http port: 20000
manager https port:20001
manager shut port: 20002
manager ajp13 port: 8009
manager health port: 8899
manager URL:       http://localhost:20000/manager
config type:       instance
config alias:      monpasoe
```

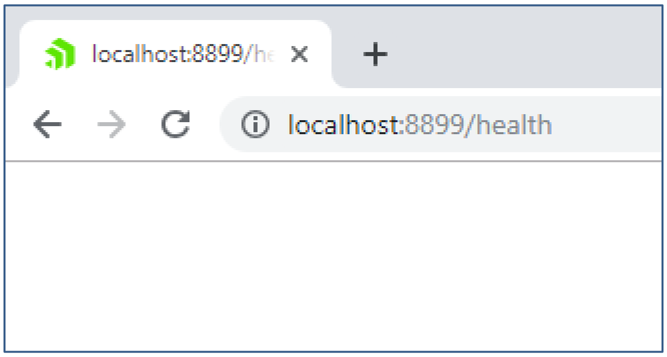
Access the OE HealthScanner data

The following table show the various ways to access the data collect by the PASOE HealthScanner probes.

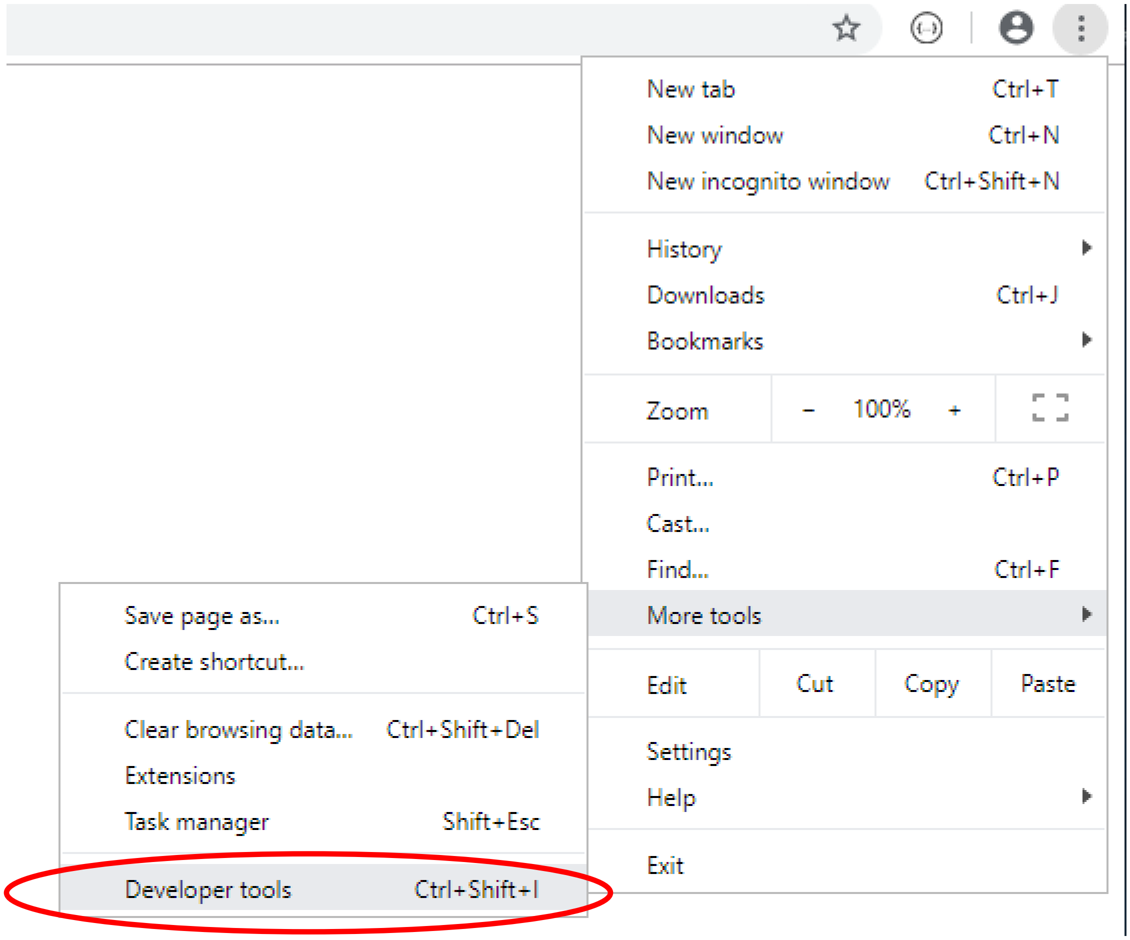
URL/Command	Use case
http://hostname:8899/health	Returns a status code. Typically, 200 when healthy and 500 if health falls below a threshold.
http://hostname:8899/health?view=summary	Returns JSON formatted summary of health score.
http://hostname:8899/health?view=details	Returns JSON formatted details of health probes.
<instance>/bin/oehealth (-D)	Uses OEJMX to gather the health values that can be pushed to a raw data collector for use with administrative scripts. (-D option returns details)

8. Get the status code returned, enter http://localhost:8899/health in a browser

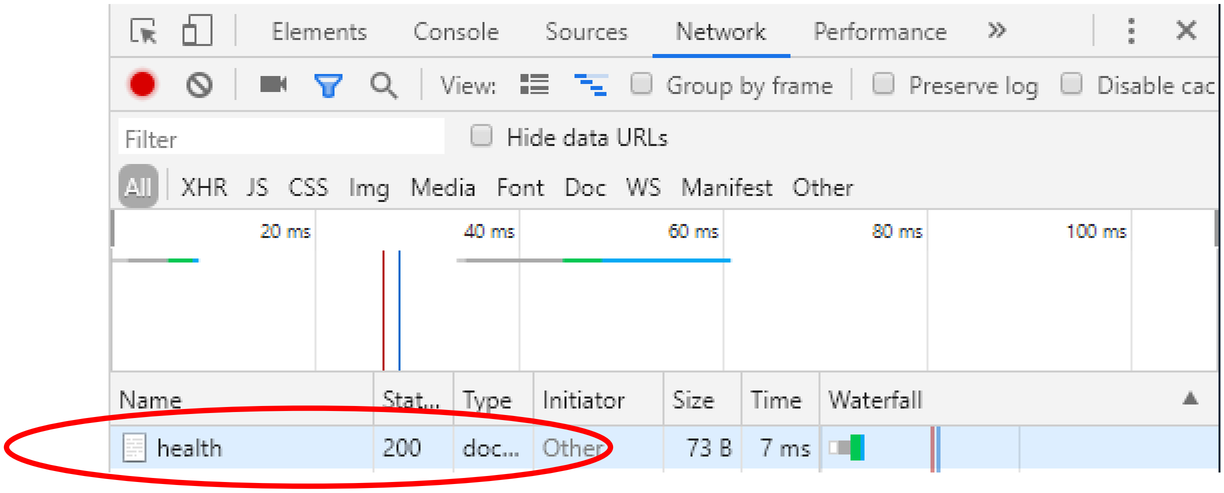
The page not return a body! The page will appear empty, but error either.



9. To see the HTTP 200 being returned enable the chrome development tools
   Select the ellipsis in the far-right corner of Chrome →
   More tools → Developer tools

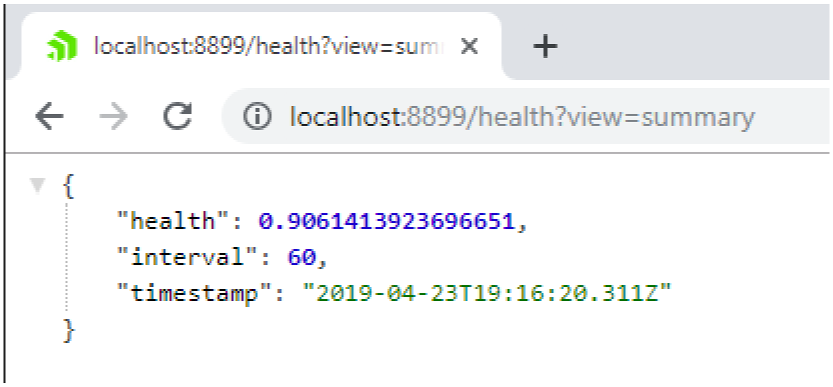


10. Select the Network tab and re-run http://localhost:8899/health



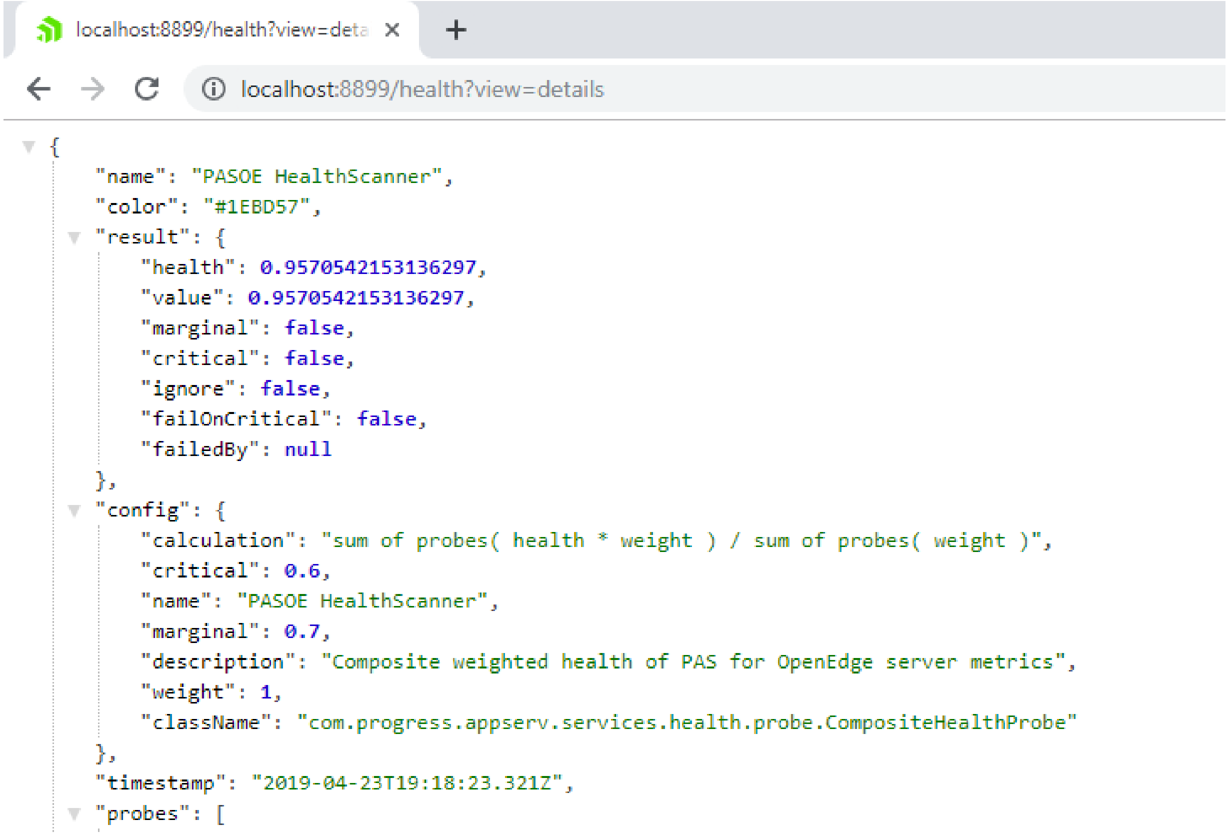
Name	Stat...	Type	Initiator	Size	Time	Waterfall
health	200	doc...	Other	73 B	7 ms	

11. Get the OE HealthScanner summary
    http://localhost:8899/health?view=summary


localhost:8899/health?view=sum
localhost:8899/health?view=summary

```
{
    "health": 0.9061413923696651,
    "interval": 60,
    "timestamp": "2019-04-23T19:16:20.311Z"
}
```

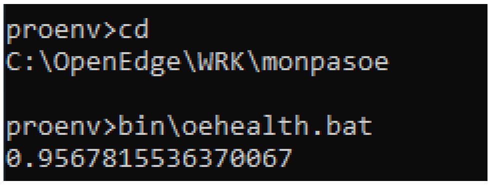
12. Get the OE HealthScanner detail
    http://localhost:8899/health?view=details


localhost:8899/health?view=deta
localhost:8899/health?view=details

```
{
    "name": "PASOE HealthScanner",
    "color": "#1EBD57",
    "result": {
        "health": 0.9570542153136297,
        "value": 0.9570542153136297,
        "marginal": false,
        "critical": false,
        "ignore": false,
        "failOnCritical": false,
        "failedBy": null
    },
    "config": {
        "calculation": "sum of probes( health * weight ) / sum of probes( weight )",
        "critical": 0.6,
        "name": "PASOE HealthScanner",
        "marginal": 0.7,
        "description": "Composite weighted health of PAS for OpenEdge server metrics",
        "weight": 1,
        "className": "com.progress.appserv.services.health.probe.CompositeHealthProbe"
    },
    "timestamp": "2019-04-23T19:18:23.321Z",
    "probes": [
```

13. Get OE HealthScanner summary with oehealth

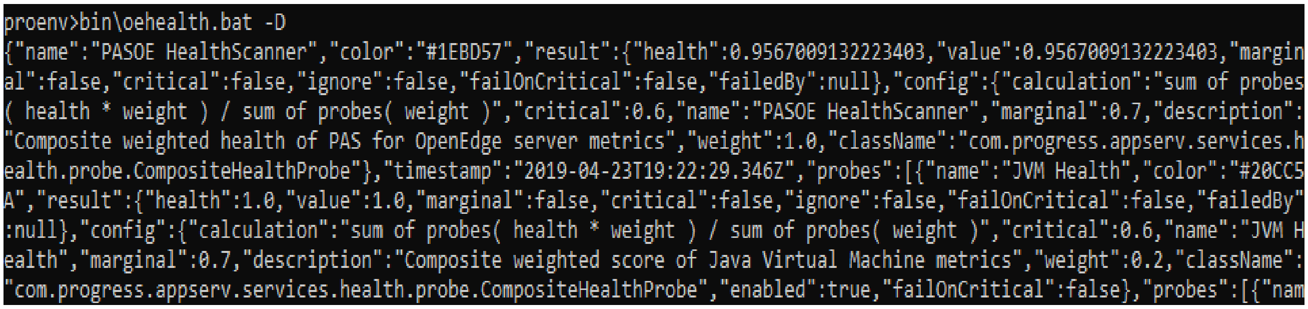
```
proenv>bin\oehealth.bat
```



```
proenv>cd
C:\OpenEdge\WRK\monpasoe

proenv>bin\oehealth.bat
0.9567815536370067
```

14. Get the OE HealthScanner detail with oehealth

```
proenv>bin\oehealth.bat -D
```



```
proenv>bin\oehealth.bat -D
{"name":"PASOE HealthScanner","color":"#1EBD57","result":{"health":0.9567009132223403,"value":0.9567009132223403,"margin
al":false,"critical":false,"ignore":false,"failOnCritical":false,"failedBy":null},"config":{"calculation":"sum of probes
( health * weight ) / sum of probes( weight )","critical":0.6,"name":"PASOE HealthScanner","marginal":0.7,"description":
"Composite weighted health of PAS for OpenEdge server metrics","weight":1.0,"className":"com.progress.appserv.services.h
ealth.probe.CompositeHealthProbe"},"timestamp":"2019-04-23T19:22:29.346Z","probes":[{"name":"JVM Health","color":"#20CC5
A","result":{"health":1.0,"value":1.0,"marginal":false,"critical":false,"ignore":false,"failOnCritical":false,"failedBy"
:null},"config":{"calculation":"sum of probes( health * weight ) / sum of probes( weight )","critical":0.6,"name":"JVM H
ealth","marginal":0.7,"description":"Composite weighted score of Java Virtual Machine metrics","weight":0.2,"className":
"com.progress.appserv.services.health.probe.CompositeHealthProbe","enabled":true,"failOnCritical":false},"probes":[{"nam
```

Section summary

Enabling OE HealthScanner requires 2 different settings and PASOE must be restarted for the changes to take effect. There are 2 different ways to get HealthScanner data via HTTP and oehealth.bat(.sh).