Progress®



OpenEdge® Management:
Progress® Application Server for OpenEdge Configuration

OpenEdge®

Copyright

© 2019 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

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.

Corticon, DataDirect (and design), DataDirect Cloud, DataDirect Connect, DataDirect Connect64, DataDirect XML Converters, DataDirect XQuery, DataRPM, Deliver More Than Expected, Icenium, Kendo UI, Kinvey, NativeScript, OpenEdge, Powered by Progress, Progress, Progress Software Developers Network, Rollbase, SequeLink, Sitefinity (and Design), Sitefinity, 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. Analytics360, AppServer, BusinessEdge, DataDirect Autonomous REST Connector, DataDirect Spy, SupportLink, DevCraft, Fiddler, JustAssembly, JustDecompile, JustMock, Kinvey Chat, NativeChat, NativeScript Sidekick, OpenAccess, ProDataSet, Progress Results, Progress Software, ProVision, PSE Pro, SmartBrowser, SmartComponent, SmartDataBrowser, SmartDataObjects, SmartDataView, SmartDialog, SmartFolder, SmartFrame, SmartObjects, SmartPanel, SmartQuery, SmartViewer, SmartWindow, and WebClient 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.

February 2019

Last updated with new content: Release 12.0.0

Updated: 2019/03/01

Table of Contents

Preface....7
Purpose....7
Audience....7
Organization....8
Using ABL documentation....8
References to ABL compiler and run-time features....8
References to ABL data types....8
Typographical conventions....9
Examples of syntax descriptions....10
Long syntax descriptions split across lines....11
Complex syntax descriptions with both required and optional elements....12
OpenEdge messages....12
Obtaining more information about OpenEdge messages....13

Chapter 1: Introducing Progress Application Server for OpenEdge....15
Create a PAS for OpenEdge instance....15
Starting or stopping a PAS for OpenEdge instance....19
Deleting a PAS for OpenEdge instance....19
Managing PAS for OpenEdge data....20
Perform Operations on Progress Application Server....21
Changing PAS for OpenEdge instance control settings....24
Installing other Web applications shipped with PAS for OpenEdge....32
Configuring a PAS for OpenEdge instance....33
Register a PAS for OpenEdge instance as a service....40

Chapter 2: Managing OpenEdge ABL applications deployed to a PAS for OpenEdge instance....43
Managing OpenEdge ABL application settings....44
Viewing OpenEdge ABL application Client Sessions....44
Viewing OpenEdge ABL application Agent Sessions....46
Viewing OpenEdge ABL application requests....47
Viewing OpenEdge ABL application connections....49
Viewing OpenEdge ABL application metrics....50
Configuring an OpenEdge ABL application....51
Managing PAS for OpenEdge transport services....59
Managing REST transports....59
Managing SOAP transports....66

Managing APSV transports....73
Managing WEB transports....77

Chapter 3: TCMAN Reference....83

Using TCMAN....83
The tcman command....84
Extending TCMAN....86
Manager actions....88
List deployed applications (list)....89
Display OS and server information (info)....90
Deploy a Web application (deploy)....91
Undeploy a Web application (undeploy)....93
Reload a Web application (reload)....94
Display detailed server status (status)....95
Display memory leaks (leaks)....96
Start a Web application (enable)....97
Stop a Web application (disable)....98
Display global server resources (resources)....99
Display Web application HTTP sessions (sessions)....100
Server actions....100
Create an instance (create)....101
Delete an instance (delete)....103
Display and manage an instance's configuration (config)....104
Display or modify the server features of an instance (feature)....106
Clean up or archive server log files (clean)....108
Display server instances (instances)....108
List process ids (plist)....110
Register an instance for tracking (register)....111
Register and manage an instance as a Windows service (service)....112
Show Windows process information (showproc)....114
Start an instance (start)....115
Stop an instance (stop)....117
Display server, OS, and runtime version information (version)....118
Test a server configuration (test)....119
Stop tracking an instance (unregister)....120
Create a Tomcat worker configuration file (workers)....121
General actions....123
Display help (help)....124
Display runtime environment information (env)....124

Index....127

Preface

For details, see the following topics:

- Purpose
- Audience
- Organization
- Using ABL documentation
- Typographical conventions
- Examples of syntax descriptions
- OpenEdge messages

Purpose

This guide describes how OpenEdge® Management and OpenEdge Explorer support the monitoring and managing of Progress Application Server for OpenEdge.

Audience

This manual is designed for system administrators, database administrators, and any other personnel responsible for the administrative and daily activities associated with monitoring and managing Progress Application Server for OpenEdge using OpenEdge Management and OpenEdge Explorer.

Organization

Introducing Progress Application Server for OpenEdge on page 15

Provides an introduction about Progress Application Server for OpenEdge (PAS for OE) and describes how to manage a PAS for OE instance.

Managing OpenEdge ABL applications deployed to a PAS for OpenEdge instance on page 43

Describes how to manage installed ABL applications and the associated transport services.

Using ABL documentation

OpenEdge provides a special purpose programming language for building business applications. In the documentation, the formal name for this language is *ABL (Advanced Business Language)*. With few exceptions, all keywords of the language appear in all `UPPERCASE`, using a font that is appropriate to the context. All other alphabetic language content appears in mixed case.

For the latest documentation updates go to https://docs.progress.com.

References to ABL compiler and run-time features

ABL is both a compiled and an interpreted language that executes in a run-time engine. The documentation refers to this run-time engine as the *ABL Virtual Machine (AVM)*. When the documentation refers to ABL source code compilation, it specifies *ABL* or *the compiler* as the actor that manages compile-time features of the language. When the documentation refers to run-time behavior in an executing ABL program, it specifies *the AVM* as the actor that manages the specified run-time behavior in the program.

For example, these sentences refer to the ABL compiler's allowance for parameter passing and the AVM's possible response to that parameter passing at run time: "ABL allows you to pass a dynamic temp-table handle as a static temp-table parameter of a method. However, if at run time the passed dynamic temp-table schema does not match the schema of the static temp-table parameter, the AVM raises an error." The following sentence refers to run-time actions that the AVM can perform using a particular ABL feature: "The ABL socket object handle allows the AVM to connect with other ABL and non-ABL sessions using TCP/IP sockets."

References to ABL data types

ABL provides built-in data types, built-in class data types, and user-defined class data types. References to built-in data types follow these rules:

- Like most other keywords, references to specific built-in data types appear in all `UPPERCASE`, using a font that is appropriate to the context. No uppercase reference ever includes or implies any data type other than itself.
- Wherever *integer* appears, this is a reference to the `INTEGER` or `INT64` data type.
- Wherever *character* appears, this is a reference to the `CHARACTER`, `LONGCHAR`, or `CLOB` data type.
- Wherever *decimal* appears, this is a reference to the `DECIMAL` data type.
- Wherever *numeric* appears, this is a reference to the `INTEGER`, `INT64`, or `DECIMAL` data type.

References to built-in class data types appear in mixed case with initial caps, for example, `Progress.Lang.Object`. References to user-defined class data types appear in mixed case, as specified for a given application example.

Typographical conventions

This documentation uses the following typographical and syntax conventions:

Convention	Description
Bold	Bold typeface indicates commands or characters the user types, provides emphasis, or the names of user interface elements.
Italic	Italic typeface indicates the title of a document, or signifies new terms.
SMALL, BOLD CAPITAL LETTERS	Small, bold capital letters indicate OpenEdge key functions and generic keyboard keys; for example, **GET** and **CTRL**.
KEY1+KEY2	A plus sign between key names indicates a **simultaneous** key sequence: you press and hold down the first key while pressing the second key. For example, **CTRL+X**.
KEY1 KEY2	A space between key names indicates a **sequential** key sequence: you press and release the first key, then press another key. For example, **ESCAPE H**.
Syntax:	
`Fixed width`	A fixed-width font is used in syntax, code examples, system output, and file names.
`Fixed-width italics`	Fixed-width italics indicate variables in syntax.
`Fixed-width bold`	Fixed-width bold italic indicates variables in syntax with special emphasis.
`UPPERCASE fixed width`	ABL keywords in syntax and code examples are almost always shown in upper case. Although shown in uppercase, you can type ABL keywords in either uppercase or lowercase in a procedure or class.
Period (.) or colon (:)	All statements except `DO`, `FOR`, `FUNCTION`, `PROCEDURE`, and `REPEAT` end with a period. `DO`, `FOR`, `FUNCTION`, `PROCEDURE`, and `REPEAT` statements can end with either a period or a colon.
[ ]	Large brackets indicate the items within them are optional.
[ ]	Small brackets are part of ABL.
{ }	Large braces indicate the items within them are required. They are used to simplify complex syntax diagrams.
{ }	Small braces are part of ABL. For example, a called external procedure must use braces when referencing arguments passed by a calling procedure.

Convention	Description
|	A vertical bar indicates a choice.
. . .	Ellipses indicate repetition: you can choose one or more of the preceding items.

Examples of syntax descriptions

In this example, ACCUM is a keyword, and *aggregate* and *expression* are variables:

Syntax

```
ACCUM aggregate expression
```

FOR is one of the statements that can end with either a period or a colon, as in this example:

```
FOR EACH Customer NO-LOCK:
  DISPLAY Customer.Name.
END.
```

In this example, STREAM *stream*, UNLESS-HIDDEN, and NO-ERROR are optional:

Syntax

```
DISPLAY [ STREAM stream ] [ UNLESS-HIDDEN ] [ NO-ERROR ]
```

In this example, the outer (small) brackets are part of the language, and the inner (large) brackets denote an optional item:

Syntax

```
INITIAL [ constant [ , constant ] ]
```

A called external procedure must use braces when referencing compile-time arguments passed by a calling procedure, as shown in this example:

Syntax

```
{ &argument-name }
```

In this example, EACH, FIRST, and LAST are optional, but you can choose only one of them:

Syntax

```
PRESELECT [ EACH | FIRST | LAST ] record-phrase
```

In this example, you must include two expressions, and optionally you can include more. Multiple expressions are separated by commas:

Syntax

```
MAXIMUM ( expression , expression [ , expression ] ... )
```

In this example, you must specify MESSAGE and at least one *expression* or SKIP [(*n*)], and any number of additional *expression* or SKIP [(*n*)] is allowed:

Syntax

```
MESSAGE { expression | SKIP [ ( n ) ] } ...
```

In this example, you must specify {*include-file*, then optionally any number of *argument* or &*argument-name* = "*argument-value*", and then terminate with }:

Syntax

```
{ include-file
    [ argument | &argument-name = "argument-value" ] ... }
```

Long syntax descriptions split across lines

Some syntax descriptions are too long to fit on one line. When syntax descriptions are split across multiple lines, groups of optional and groups of required items are kept together in the required order.

In this example, WITH is followed by six optional items:

Syntax

```
WITH [ ACCUM max-length ] [ expression DOWN ]
  [ CENTERED ] [ n COLUMNS ] [ SIDE-LABELS ]
  [ STREAM-IO ]
```

Complex syntax descriptions with both required and optional elements

Some syntax descriptions are too complex to distinguish required and optional elements by bracketing only the optional elements. For such syntax, the descriptions include both braces (for required elements) and brackets (for optional elements).

In this example, `ASSIGN` requires either one or more *`field`* entries or one *`record`*. Options available with *`field`* or *`record`* are grouped with braces and brackets:

Syntax

```
ASSIGN   { [ FRAME frame ] { field [ = expression ] }
           [ WHEN expression ] } . . .
       | { record [ EXCEPT field . . . ] }
```

OpenEdge messages

OpenEdge displays several types of messages to inform you of routine and unusual occurrences:

- **Execution messages** inform you of errors encountered while OpenEdge is running a procedure; for example, if OpenEdge cannot find a record with a specified index field value.
- **Compile messages** inform you of errors found while OpenEdge is reading and analyzing a procedure before running it; for example, if a procedure references a table name that is not defined in the database.
- **Startup messages** inform you of unusual conditions detected while OpenEdge is getting ready to execute; for example, if you entered an invalid startup parameter.

After displaying a message, OpenEdge proceeds in one of several ways:

- Continues execution, subject to the error-processing actions that you specify or that are assumed as part of the procedure. This is the most common action taken after execution messages.
- Returns to the Procedure Editor, so you can correct an error in a procedure. This is the usual action taken after compiler messages.
- Halts processing of a procedure and returns immediately to the Procedure Editor. This does not happen often.
- Terminates the current session.

OpenEdge messages end with a message number in parentheses. In this example, the message number is `200`:

```
** Unknown table name table. (200)
```

If you encounter an error that terminates OpenEdge, note the message number before restarting.

Obtaining more information about OpenEdge messages

In Windows platforms, use OpenEdge online help to obtain more information about OpenEdge messages. Many OpenEdge tools include the following Help menu options to provide information about messages:

- Choose **Help** > **Recent Messages** to display detailed descriptions of the most recent OpenEdge message and all other messages returned in the current session.
- Choose **Help** > **Messages** and then type the message number to display a description of a specific OpenEdge message.
- In the Procedure Editor, press the **HELP** key or **F1**.

On UNIX platforms, use the OpenEdge `pro` command to start a single-user mode character OpenEdge client session and view a brief description of a message by providing its number.

To use the pro command to obtain a message description by message number:

1. Start the Procedure Editor:

```
OpenEdge-install-dir/bin/pro
```

2. Press **F3** to access the menu bar, then choose **Help** > **Messages**.
3. Type the message number and press **ENTER**. Details about that message number appear.
4. Press **F4** to close the message, press **F3** to access the Procedure Editor menu, and choose **File** > **Exit**.

1

Introducing Progress Application Server for OpenEdge

The Progress Application Server for OpenEdge (PAS for OpenEdge) is a platform that provides Web server support for Progress applications. Progress applications are packaged as Web application archives (WAR files) and are deployed on the Java servlet container of a running PAS for OpenEdge instance. Apache Tomcat is used as the Web server that includes the Java servlet container for hosting Web applications. Client access to PAS for OpenEdge is through HTTP/HTTPS protocols.

OpenEdge Management and OpenEdge Explorer help you create, configure, and manage PAS for OpenEdge and PAS for OpenEdge instances. Additionally, you can install Web applications on PAS for OpenEdge and manage the applications.

For details, see the following topics:

- Create a PAS for OpenEdge instance
- Starting or stopping a PAS for OpenEdge instance
- Deleting a PAS for OpenEdge instance
- Managing PAS for OpenEdge data

Create a PAS for OpenEdge instance

OpenEdge Management and OpenEdge Explorer provide a sample PAS for OpenEdge instance (`oepas1`) with a default configuration. Each new instance of PAS for OpenEdge that you create uses the same default configuration, which you can edit.

OpenEdge Management monitors the PAS for OpenEdge installation and automatically creates resources if you use the `TCMAN` command to create a new PAS for OpenEdge instance. `TCMAN` is a command-line utility for managing and administering PAS for OpenEdge.

To create a PAS for OpenEdge instance:

1. Click **Resources** > **New** > **Progress Application Server** in the OpenEdge Management console menu.

   The **Create new Progress Application Server** page appears.

2. Configure your Application Server as described in the following table:

Field	Description
AdminServer	Select an AdminServer for the PAS for OpenEdge instance. If you have created multiple remote AdminServers in OpenEdge Management or OpenEdge Explorer (**Options** > **Remote AdminServers**), this field is displayed for you to select an AdminServer.
Instance name	Specify a name for the PAS for OpenEdge instance. This is a mandatory field. **Note:** The PAS for OpenEdge instance name is case-sensitive. It can include any character except periods (.) or square brackets ([ ]). The name must be unique among all configured PAS for OpenEdge names.
Location	Select the location of the PAS for OpenEdge instance: • **Local**: OpenEdge Management has access to the machine where PAS for OpenEdge runs through an AdminServer. PAS for OpenEdge can be running either through a remote AdminServer or through an AdminServer in the same location as OpenEdge Management. • **Remote**: The PAS for OpenEdge instance is in a different location than OpenEdge Management, that is, the instance exists in a location without an AdminServer. When you select the location as **Remote**, you cannot not use OpenEdge Management to start or stop a PAS for OpenEdge instance. **Note:** OpenEdge Management monitors only the local installation of new PAS for OpenEdge instances. If you have installed PAS for OpenEdge as a stand-alone product and have not installed an OpenEdge product that includes an AdminServer, select **Remote** to monitor the instance. The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to use the PAS for OpenEdge instance from a remote location.
Security model	Select a security model: • **Production**: The *oemanager* and *manager* Web applications need to be deployed separately. Make configuration changes to adjust security settings to ensure your application operates correctly in a production instance.

Field	Description
	• **Developer**: The *oemanager* and *manager* Web applications are deployed by default. Development mode minimizes or eliminates any accessibility restrictions, so a developer can quickly use the product “out of the box” to develop, test, and debug applications.
URL	Specify the URL of the PAS for OpenEdge instance if PAS for OpenEdge is running remotely. The format of the URL must be `http(s)://server:port/oemanager`. **Note:** For a local instance, OpenEdge Management or OpenEdge Explorer constructs the URL using localhost and port for secure communication. If you want to use HTTP to monitor the instance, you must specify the URL manually. For using secure communication, you must install all the required certificates in `$DLC/certs`.
No host verification for SSL	Select the check box if you do not want to validate the hostname in the URL of the PAS for OpenEdge instance that is enabled for secure communication.
Instance directory	Specify the name of the directory where the PAS for OpenEdge instance resides. This is an optional field. **Note:** If you do not provide the name of the instance directory, the PAS for OpenEdge instance is created in the working directory set during OpenEdge installation. The name of the instance is used as the name of the directory where the PAS for OpenEdge installation is created. The directory must not already exist (either when provided, or when using the default location).
HTTP port	Specify an unused port number associated with the PAS for OpenEdge HTTP. By default, this is set to 8080. **Note:** Each new PAS for OpenEdge instance that you create uses the default configuration. However, the port number must be unique for each PAS for OpenEdge instance for the instance to operate properly. If you specify a port number that is used by another PAS for OpenEdge instance, you are prompted to confirm whether you want to use the port number.
HTTPS port	Specify an unused port number associated with the PAS for OpenEdge HTTPS. By default, this is set to 8443. **Note:** If you specify a port number that is used by another PAS for OpenEdge instance, you are prompted to confirm whether you want to use the port number.

Field	Description
Shutdown port	Specify an unused port number for shutdown. If you are creating a PAS for OpenEdge instance on a Windows machine, this is a mandatory field. **Note:** If you specify a port number that is used by another PAS for OpenEdge instance, you are prompted to confirm whether you want to use the port number.
Autostart	Set the **Autostart** option to automatically start PAS for OpenEdge every time you start the AdminServer.
Tomcat manager login	Specify the login ID of the Tomcat Web application that hosts the PAS for OpenEdge instance. If you are using the Tomcat Web server shipped with OpenEdge, then the default login ID is `tomcat`. You can also specify your own login ID other than the default ID, and it is stored in the `tomcat-users.xml` file.
Tomcat manager password	Specify the password of the Tomcat Web application that hosts the PAS for OpenEdge instance. If you are using the Tomcat Web server shipped with OpenEdge, then the default password is `tomcat`. You can also specify your own login password other than the default password, and it is stored in the `tomcat-users.xml` file.

3. Click **Save** to complete creating the PAS for OpenEdge instance.

   The PAS for OpenEdge instance is listed along with other resource instances on the **Resources** page.

Note:

The oemanager and manager Web applications are installed depending on the following combinations:

License	Security Model	Result
Progress Dev AS for OE	Developer	oemanager and manager are installed.
Progress Dev AS for OE	Production	oemanager and manager are not installed.
Progress Prod AS for OE	Developer	oemanager and manager are not installed.
Progress Prod AS for OE	Production	oemanager and manager are not installed.
Progress Dev AS for OE and Progress Prod AS for OE	Developer	oemanager and manager are installed.
Progress Dev AS for OE and Progress Prod AS for OE	Production	oemanager and manager are not installed.

Starting or stopping a PAS for OpenEdge instance

From the management console, you can start or stop a local PAS for OpenEdge instance. You must first connect to the AdminServer that manages the PAS for OpenEdge instance that you want to start. You can also choose to start the PAS for OpenEdge instance automatically when the AdminServer starts.

If you start a PAS for OpenEdge instance with incorrect credentials, such as login, password, URL, hostname, or port, the OpenEdge manager login fails and displays the following in the instance's home page:

- An error message —

Error message	Reason
Access to oemanager 'Unauthorized' for https://hostname:port/oemanager	Incorrect login and/or password
Access to oemanager 'Forbidden' for https://hostname:port/oemanager	Incorrect URL
Access to oemanager 'Connect to hostname:port failed: connect timed out' for https://hostname:port/oemanager	Invalid hostname or port

- An alert — **OEManagerNotAccessible**

  This alert is triggered after the next poll and is also displayed on the **Alerts** page.

- The current status of the PAS for OpenEdge instance as **Internal Error**.

After a sucessful login, the error message is cleared and the status changes to **Passed** immediately. However, the alert is cleared only after the next poll. For editing the configuration properties of a PAS for OpenEdge instance, see Configuring a PAS for OpenEdge instance on page 33.

You must start or stop the application server in the same way you start or stop any resource in OpenEdge Management. To start or stop a PAS for OpenEdge instance, refer to the *Starting or Stopping OpenEdge resources* section in *OpenEdge Management: Servers, DataServers, Messengers, and Adapters*.

As for a remote PAS for OpenEdge instances, you must start or stop them from the remote location.Therefore, OpenEdge Management or OpenEdge Explorer only provides the ability to add, delete, and monitor the remote PAS for OpenEdge instance.

Deleting a PAS for OpenEdge instance

You can delete a local or a remote PAS for OpenEdge instance listed in the management console when the instance is not running. The **Delete** button is not available when the instance is running, therefore, you must first stop the PAS for OpenEdge instance to be able to delete it.

To delete a PAS for OpenEdge instance:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS for OpenEdge instance.

The **Progress Application Server: <instance name>** page appears.

3. Click **Delete**, and then click **Yes** in the confirmation dialog box that appears to confirm the deletion.

Note: You can delete a local PAS for OpenEdge instance only if it in the stopped state, and deleting a local PAS for OpenEdge deletes the entire instance directory along with the resource. As for a remote PAS for OpenEdge instance, delete operation simply removes the resource from the OpenEdge Management or OpenEdge Explorer configuration without actually stopping and deleting the remote PAS for OpenEdge instance from the remote location.

Managing PAS for OpenEdge data

You can install and manage ABL apps, Web apps and Web app services in PAS for OpenEdge using the **ABL APPLICATIONS**, **WEB APPLICATIONS** and **WEB APP SERVICES** tabs. All the related command and controls for ABL apps, Web apps and Web app services are internally performed using the *tcman* utility, because of which these operations are available only for the instances that are hosted locally and not for the remotely hosted ABL apps, Web apps and Web app services in which the *tcman* utility is inaccessible to OpenEdge Management and OpenEdge Explorer.

Note: You must install only the ABL apps, Web apps and Web app services that you want to manage. The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL apps, Web apps and Web app services while the PAS for OpenEdge instance is running. The Tomcat manager Web application, `manager`, must be installed to deploy and undeploy ABL apps, Web apps and Web app services while the PAS for OpenEdge instance is running. You may choose not to install these Web applications to improve security, but in such a scenario, OpenEdge Management or OpenEdge Explorer cannot manage the ABL apps, Web apps and Web app services.

Therefore, to manage remote ABL apps, Web apps and Web app services instances, you must use the *tcman* utility in the remote location. For more information on managing ABL apps, Web apps and Web app service instances using the *tcman* utility, see the TCMAN reference sections in your OpenEdge documentation.

The following sections discuss the different commands and controls used to configure PAS for OpenEdge. These are all the different options under a PAS for OpenEdge instance page.

You can perform the following functions from the **Progress Application Server: <instance name>** page:

- Changing PAS for OpenEdge instance control settings on page 24
- Installing other Web applications shipped with PAS for OpenEdge on page 32
- Manage Web applications on page 22
- Configuring a PAS for OpenEdge instance on page 33
- Register a PAS for OpenEdge instance as a service on page 40

The **Progress Application Server: <instance name>** page displays five of the most recent alerts. You can clear an individual alert or click **Clear all alerts** to clear all the alerts from the page. When there are more than five alerts, the page displays a **See complete list** button to open the **Alerts** page, which displays all alerts. For more information about alert features, see *OpenEdge Management: Alerts Guide and Reference*.

To view the graphs and charts pertaining to the CPU usage, physical, and virtual memory for PAS from Resources for a selected PAS instance, navigate to **Resources > Go to Resources > Progress Application Server > <PAS instance>** and click **Graphs/Charts**. Using the options on the top-right corner of the graph, you can select a time period, ranging from 2 hours to 2 weeks, to display the data collected for that period of time, reload the graph, and select different forms of graphical interpretation.

Perform Operations on Progress Application Server

The ABL applications page view appears as the default landing page for a specific Progress Application Server (PAS) instance.

This page displays the following tabs to select and perform required operations on:

- **ABL Applications** — Manage ABL applications
- **Web Applications** — Manage Web applications
- **Web App Services** — Manage Web app services

You can also navigate to the following pages:

Page Name	Description
Instance Control	Start or stop a Progress Application Server (PAS) instance.
Monitoring Plans	Define monitoring plans to alert on Progress Application Server conditions.
Configuration	Display the Progress file configuration associated with this Progress Application Server.
Log File	Perform view and examine operations on log files. • **Catalina Log File Monitor** — Log file on which to alert • **Catalina Log File Viewer** — To examine the log file • **Host Manager Log File Monitor** — Log file on which to alert • **Host Manager Log File Viewer** — To examine the log file
Register/Unregister as Service	Register or unregister the Progress Application Server as a Windows service (Windows only).
Charts/Graphs	Displays charts and graphs on CPU, Memory and Virtual Memory usage for the Progress Application Server on Tomcat.

Manage ABL applications

To manage ABL applications, ensure that the PAS for OpenEdge instance is running. From the OpenEdge Management menu, select **Resources-> Progress Application Server**. Filter or search for and select the required PAS for OpenEdge instance.

The **ABL APPLICATIONS** page appears by default, listing all the existing ABL applications managed by PAS for OpenEdge.

Perform any of the following operations:

Operation	Action Description
Refresh	Click to refresh the ABL application grid.
Deploy	To deploy an ABL application. Click the **Deploy** tab. The **Deploy ABL Application** wizard appears. **1.** In the **Select application .WAR** field, perform one of the following actions to upload an ABL application. **Note:** Ensure that the selected file is an ABL application. • **Browse** to locate and select an ABL application .war file. **2.** Type a name in the **ABL application name** field. Select the application name from the auto suggest list that appears. The name must contain only alphanumeric characters. It must not contain blank spaces or special characters, except underscore (“_”) and hyphen (“-”). **3.** Type a name for the associated web application in the **Web application name** field **4.** **Deploy** the application on PAS for OpenEdge. The OpenEdge ABL Web application is deployed.
Reload	This operation is disabled by default and is enabled when an ABL application is selected to reload the ABL application on PAS for OpenEdge. Choose multiple ABL applications to perform this operation in bulk.
Undeploy	This operation is disabled by default and is enabled when an ABL application is selected to undeploy an ABL application on PAS for OpenEdge. Choose multiple ABL applications to perform this operation in bulk.

Manage Web applications

To manage your Web applications on the PAS for OpenEdge:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

   The **ABL APPLICATIONS** page appears by default, listing all the existing ABL applications managed by PAS for OpenEdge.

   Select the **WEB APPLICATIONS** tab. The Web applications grid appears, listing all the existing Web applications managed by PAS for OpenEdge.

Operation	Perform
Deploy	To deploy a Web application. Click **Deploy**. The **Deploy web application** dialog appears. **1.** In the **Select application .war file** field, perform one of the following actions to upload a Web application. **Note:** Ensure that the selected file is an Web application. • Type a file name to select the Web application .war file from the extras folder. The auto suggest list appears as you type the name. You can also give a new name for the Web application. • **Browse** to locate and select an application .war file. • Type a name for the selected application in the **Web application name** field. The name must contain only alphanumeric characters. It must not contain blank spaces or special characters, except underscore (“_”) and hyphen (“-”). • Click **Deploy** to deploy the application on PAS for OpenEdge. The OpenEdge Web application is deployed.
Start	This operation is disabled by default and is enabled when there is a Web application(s) available with **Status** *stopped*. Select a Web application(s) and click **Start** to start a Web application. Choose multiple Web applications to perform this operation in bulk.
Stop	This operation is disabled by default and is enabled when there is a Web application(s) available with **Status** *running*. Select a Web application(s) and click **Stop** to stop a Web application. Choose multiple Web applications to perform this operation in bulk. This option makes the Web application and all its services unavailable for requests. **Note:** The Web application is automatically restarted when the PAS for OpenEdge instance is started.
Reload	Select a Web application and click **Reload** to stop a Web application and restart it on PAS for OpenEdge. Choose multiple Web applications to perform this operation in bulk. This option reloads the configurations settings and the Web application is unavailable till the reload is successful. For more information on the reload option, see http://tomcat.apache.org/tomcat-7.0-doc/manager-howto.html.
Undeploy	Select a Web application and click **Undeploy** to undeploy a Web application from PAS for OpenEdge. Choose multiple Web applications to perform this operation in bulk. This option stops the Web application and physically removes it from PAS for OpenEdge.

Manage Web app services

To manage Web App Services, ensure that the PAS for OpenEdge instance is running. From the OpenEdge Management menu, select **Resources-> Progress Application Server**. Filter or search for and select the required PAS for OpenEdge instance.

The **ABL APPLICATIONS** page appears by default, listing all the existing ABL applications managed by PAS for OpenEdge.

Select the **WEB APP SERVICES** tab. The Web application services grid displays a list of deployed Web application services, if available. The **_oepingService** is the default REST **Service name** for any Web application. Perform any of the following operations:

Operation	Action Description
Refresh	Click to refresh the Web application services grid.
Deploy	To deploy a Web application service of type REST or SOAP .Click the **Deploy** tab. The **Deploy web service** wizard appears. **1.** In the **Select transport service file** field, perform one of the following actions to upload a Web application service. • **Browse** to locate and select a `.paar`, `.zip` or `.wsm` file. • In the **Web application name** field, search and select the associated Web application name for the Web application service. **Note:** This field appears only when a `.paar` or `.zip` is selected. **2.** Type a name for the Web service in the **Web service name**. **3.** **Deploy** the Web service application on PAS for OpenEdge. The OpenEdge Web service application is deployed.
Undeploy	This operation is disabled by default and is enabled when a Web application service is selected. To undeploy a Web application service on PAS for OpenEdge. Choose multiple Web application services to perform this operation in bulk.
Disable	This operation is disabled by default and is enabled when there is a Web application service(s) available with **State** *ENABLED*. To disable a Web application service. Choose multiple Web application services to perform this operation in bulk.
Enable	This operation is disabled by default and is enabled when there is a Web application service(s) available with **State** *DISABLED*. To enable a Web application service. Choose multiple Web application services to perform this operation in bulk.

Changing PAS for OpenEdge instance control settings

From the **Progress Application Server: <instance name>** details page, you can access the **Instance Control** page. The **Instance Control** page provides information about the current PAS for OpenEdge instance, PAS for OpenEdge status, and PAS for OpenEdge agent settings, and allows you to start and stop PAS for OpenEdge.

You can perform the following from the **Instance Control** page of a PAS for OpenEdge instance:

- Stop PAS for OpenEdge by clicking **Stop Progress Application Server**. Click **Yes** in the confirmation dialog that appears. The status is updated to **Not Running**. To restart PAS for OpenEdge, click **Start Progress Application Server**.

  The PAS for OpenEdge agent works only if PAS for OpenEdge is running. Therefore, when you stop PAS for OpenEdge, the monitoring agent status automatically changes to **Not Running**.

- Change the status of a PAS for OpenEdge monitor from **Enabled** to **Disabled** by clicking **Edit** and then clearing the **Enabled** option.

  Even after disabling PAS for OpenEdge monitors, PAS for OpenEdge still appears under **Progress Application Server** in the **Resources** list. However, PAS for OpenEdge is no longer being monitored by OpenEdge Management.

For more information about accessing and reviewing PAS for OpenEdge log file data, see Accessing and reviewing PAS for OpenEdge log file data on page 25.

Accessing and reviewing PAS for OpenEdge log file data

OpenEdge Management and OpenEdge Explorer supports monitoring log files and their associated viewers for the following PAS for OpenEdge resources:

- The PAS for OpenEdge instance (Catalina)
- The Apache Tomcat host manager

Log files can store a tremendous amount of data. Therefore, monitoring and analyzing data collected within these files might help you to better determine performance expectations and examine trends related to the PAS for OpenEdge instance.

This section presents information related to both types of PAS for OpenEdge log file monitors and viewers. However, only the procedures specific to the Catalina log file monitor and its associated viewer are presented. These procedures will work with the host manager log file monitor and viewer.

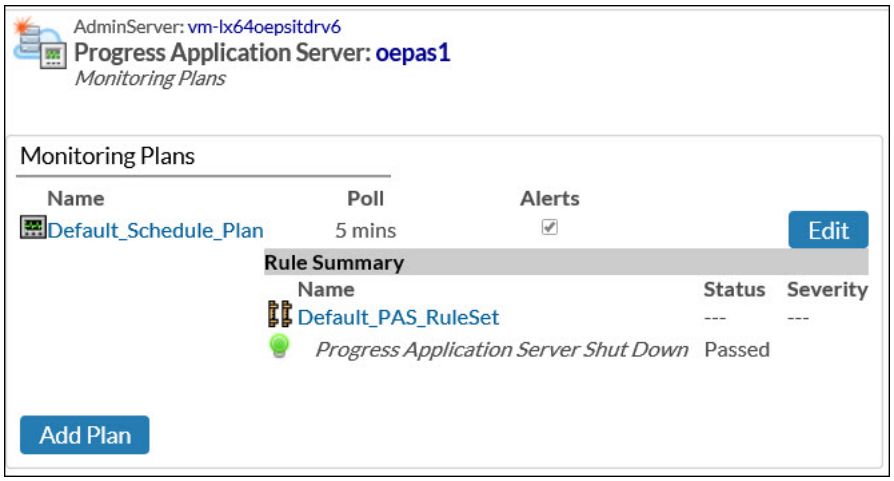
To access and review the PAS for OpenEdge log file data from the **Progress Application Server: <instance name>** details page, see:

- Default PAS for OpenEdge monitoring plans on page 26
- Modifying PAS for OpenEdge monitoring plans on page 28
- Characteristics of PAS for OpenEdge log file monitors on page 30
- Customizing PAS for OpenEdge log file monitors on page 30
- Using PAS for OpenEdge log file viewers on page 31

Default PAS for OpenEdge monitoring plans

OpenEdge Management and OpenEdge Explorer automatically creates a monitoring plan for each PAS for OpenEdge instance. The default monitoring plan contains a default rule set and schedule. You can change the default values at any time, or you can create a new monitoring plan.

Figure 1: Sample default PAS for OpenEdge instance monitoring plan



The default monitoring plan consists of the following values:

Table 1: Default monitoring plan

Field	Default value	Description
Name	Default_Schedule_Plan	The name of the system-defined 24/7 default schedule that is used when the plan is active
Poll	5 mins	The polling cycle set up for this PAS for OpenEdge instance resource monitor. The polling cycle is the frequency at which the resource's rules are checked.
Alerts	Enabled	Indicates whether alerts are active and whether they are generated when the plan is active
Rule summary	Default_PAS_RuleSet	This rule set consists of the **Progress Application Server Shut Down** rule set

Note: OpenEdge Management and OpenEdge Explorer prevents the assignment of schedules that share overlapping time periods. For example, if you have a default schedule set up for a resource monitor, you cannot set up an additional plan because **Default_Schedule_Plan** is defined for 7 days a week, 24 hours a day. In order to add other plans you must modify or remove **Default_Schedule_Plan**.

Default rules for PAS for OpenEdge monitoring plans

OpenEdge Management provides default rules to create rule sets for PAS for OpenEdge monitoring plans. A rule defines a performance standard for a resource and triggers alerts when the standard is not met. Each rule has a unique alert and the action associated with it.

The types of rules provided for PAS for OpenEdge are:

- Rules with asynchronous alerts
- Rules with polled alerts

Rules with asynchronous alerts – These rules trigger alerts when a specific event or condition is detected such as shutdown of a service, and do not have threshold values.

The rules with asynchronous alerts available for PAS for OpenEdge are:

Rule	Triggers alerts when...
Progress Application Server Start Up	The PAS for OpenEdge instance is started.
Progress Application Server Shut Down	The PAS for OpenEdge instance is stopped.

Rules with polled alerts – These rules check (poll) a PAS for OpenEdge instance according to the time interval specified and trigger alerts when the instance performs outside the defined threshold value. The threshold values range from 1 to 100 percent.

The rules with polled alerts available for PAS for OpenEdge are:

Rule	Triggers alerts when...	Default threshold value
Progress Application Server Agents CPU Percentage	The percentage of CPU utilized by all the PAS agents exceeds the threshold value.	80%
Progress Application Server Agents Memory	The amount of physical memory consumed by all the PAS agents exceeds the threshold value.	80%
Progress Application Server Agents Utilization High	The percentage of the number of agents utilized by each selected ABL application exceeds the threshold value. For more information, see Progress Application Server Agents Utilization High rule on page 28.	80%
Progress Application Server Sessions Utilization High	The percentage of the number of sessions utilized by the selected ABL applications exceeds the threshold value. For more information, see Progress Application Server Sessions Utilization High rule on page 28.	80%
Progress Application Server Tomcat CPU Percentage	The percentage of CPU utilized by the PAS Tomcat server exceeds the threshold value.	80%
Progress Application Server Tomcat Memory	The amount of physical memory consumed by the PAS Tomcat server exceeds the threshold value.	80%
Progress Application Server Tomcat Virtual Memory	The amount of virtual memory consumed by the PAS Tomcat server exceeds the threshold value.	80%

You cannot create rules for PAS for OpenEdge monitoring plans. However, you can edit the rules to reflect your resource monitoring objectives at any time. You can also group multiple rules of multiple resources logically to create a rule set and associate it with a monitoring plan.

For more information about rules, rule sets, and alerts, see *OpenEdge Management: Resource Monitoring* and *OpenEdge Management: Alerts Guide and Reference*.

Progress Application Server Agents Utilization High rule

The **Progress Application Server Agents Utilization High** rule triggers an alert when the percentage of the number of agents utilized by each selected ABL application exceeds a threshold value. The default threshold value is 80% of the maximum number of agents.

The **Applications List** section displays all the ABL applications that are currently deployed, and you can select a desired ABL application from the list. For more information about modifying a monitoring plan and the rules associated with it, see Modifying PAS for OpenEdge monitoring plans on page 28.

Note: You must select at least one ABL application from the **Applications List** section before saving the rule. If not, a validation message is displayed.

The percentage of the number of agents utilized by the selected ABL application is calculated using the following formula:

```
Number of agents utilized = Current agents count/Maximum number of agents
```

For configuring the maximum number of agents, see Configuring tuning parameters and process limits on page 54.

Progress Application Server Sessions Utilization High rule

The **Progress Application Server Sessions Utilization High** rule triggers an alert when the percentage of the number of sessions utilized by the selected ABL applications exceeds a threshold value. The default threshold value is 80% of the maximum sessions of the PAS instance.

The **Applications List** section displays all the ABL applications that are currently deployed, and you can select the desired ABL applications from the list. For more information about modifying a monitoring plan and the rules associated with it, see Modifying PAS for OpenEdge monitoring plans on page 28.

Note: You must select at least one ABL application from the **Applications List** section before saving the rule. If not, a validation message is displayed.

The percentage of the number of sessions utilized by the ABL applications is calculated using the following formula:

```
Number of sessions utilized = Current session count/(Maximum number of agents * Maximum
 ABL sessions per agent)
```

For configuring the maximum number of agents and maximum ABL sessions per agent, see Configuring tuning parameters and process limits on page 54.

Modifying PAS for OpenEdge monitoring plans

You can modify the default PAS for OpenEdge monitoring plan, **Default_Schedule_Plan**, from the **Progress Application Server: <instance name>** page.

To modify the PAS for OpenEdge monitoring plan:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

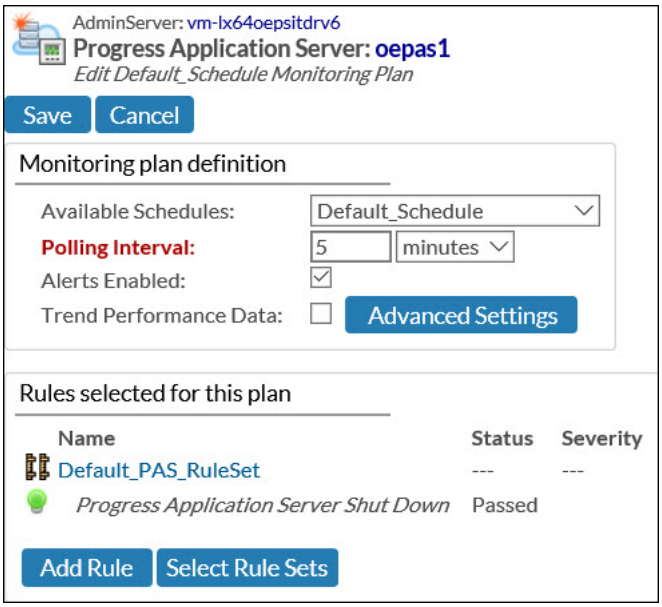
   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click **Monitoring Plans** in the **Command and control** section.

   The **Monitoring Plans** page appears.

4. Click **Edit** for the schedule associated with the plan that you want to update. When **Default_Schedule_Plan** is selected, the **Edit Default_Schedule Monitoring Plan** page appears.



5. Edit the values in the **Monitoring plan definition** section as required. Click **Advanced Settings** to view and modify the current trend value settings.

6. Click **Save**.

   The **Default_Schedule Monitoring Plan** page appears.

7. Click **Add Rule** in the **Rules selected for this plan** section.

   The **Available Rules** page appears.

8. Select the rule you want to add, update the values you want to edit, and click **Save**.

   Note: Any rules you define and add are associated only with this plan. If you create another plan and add the same rules, you can select values that are appropriate for that plan.

   The **Available Rules** page appears.

9. Repeat step 8 for each additional rule you want to apply to this plan. Once you add and define the criteria for each rule you want to add, click **Done Adding Rules** on the **Available Rules** page.

The **Default_Schedule Monitoring Plan** page reappears. You can also add more rule sets to your plan by clicking **Select Rule Sets** in the **Rules selected for this plan** section of the page.

10. Click **Save**. The updated monitoring plan appears in the **Monitoring plan definition** section at the top of the **Monitoring Plan** summary page.

Characteristics of PAS for OpenEdge log file monitors

The data that you can capture and view using the PAS for OpenEdge resources log file monitors and viewers helps you to:

- Ensure the integrity of these log files by monitoring files for errors and allowing you to define actions that trigger when errors occur.
- Use pre-defined PAS for OpenEdge-related search criteria, or create your own, to run against the data in these log files. OpenEdge Management and OpenEdge Explorer predefines search criteria to support the log file monitors.

Customizing PAS for OpenEdge log file monitors

You can customize the PAS for OpenEdge log file monitor from the **Progress Application Server: <instance name>** page. The steps for customizing the Catalina log file monitor and the host manager log file monitor are the same.

To customize the PAS for OpenEdge Catalina log file monitor:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click **Catalina Log File Monitor** in the **Command and control** section.

   The **Log File Monitor for: Collection pas catalina** page appears.

   The status of the log file is displayed, which includes information such as log file directory and file name pattern.

4. Customize or view the contents of the Catalina log file monitor as follows:

   - Click **Edit** at the top of the page to change the description, directory, and file name pattern of the log file monitor. You can set the file name pattern of the log file to follow a date-based file name or to use a fixed file name. Additionally, if you are using a date-based file name pattern, you can control whether you want receive alerts every time the log file name has changed.
   - Click **Copy** at the top of the page to create an editable copy of the log file. The log file copy is visible under the **Log File** resource section of the **Resources** tab.
   - Click **Log File Viewer** at the top of the page to view the contents of the log file monitor.
   - Five of the most recent alerts are also displayed on the page and you can clear an individual alert or click **Clear all alerts** to clear all the alerts from the page. When there are more than five alerts, the page displays a **See complete list** button to open the **Alerts** page, which displays all alerts.

Note: OpenEdge Management and OpenEdge Explorer prevents the assignment of schedules that share overlapping time periods. For example, if you have a default schedule set up for a resource monitor, you cannot set up an additional plan because **Default_Schedule_Plan** is defined for 7 days a week, 24 hours a day. In order to add other plans you must modify or remove **Default_Schedule_Plan**.

5. To add individual rules, click **Edit** under the **Monitoring plans** section.

   The **Edit Default_Schedule Monitoring Plan** page appears.

6. Click **Add Rule** under the **Rules selected for this plan** section. The **Create new Log File Rule** page appears. You can add a rule that is already defined or create a new rule.

   To use a PAS for OpenEdge instance rule already defined in the library:

   1. Select **Progress Application AppServer** from the **Choose Criteria Category** list.
   2. Select a criteria from the **Choose Search Criteria** list.

   To create a new PAS for OpenEdge instance rule:

   1. Click **Create Criterion**. The **Create new Search Criterion** page appears.
   2. Enter the required values of **Name** (the name of the search criteria) and **Search Text** (the string to be searched for).
   3. Select the search type as **Literal Search** or **Regular Expression**.
   4. Select whether you want to use an existing category or a new category for the rule. For an existing category, select **Progress Application AppServer** from the **Category** list.
   5. Click **Save**. The **Create new Log File Rule** page appears.

      The values you define and select to create a rule on the **Create new Search Criterion** page are now available on the **Create new Log File Rule** page. The **Choose Search Category** list displays the name you entered in the **Name** field on the **Create new Search Criterion** page. The **Choose Criteria Category** list displays the category in which you elect to store the new rule.

7. Select the required values from the **Severity** and **On Alert Action Perform** lists to complete the alert severity and action definition that you want to associate with this rule.
8. Click **Save**.
9. To add another individual rule, repeat steps 5 through 9.
10. Click **Select Rule Sets** to create a new log file rule. You can also select from the existing rule sets to add to the monitoring plan.
11. Click **Save**. The **Log File Monitor for: Collection pas catalina** page reappears. You can view the rules updated under the **Rules Summary** section.
12. Click **Add Plan** to add an existing monitoring plan to this resource monitor.

Using PAS for OpenEdge log file viewers

To view the contents of each PAS for OpenEdge instance log file, you can access the viewer associated with each log file. The log file viewer allows you to examine the contents of a PAS for OpenEdge instance log file using a Web interface. You can access these log file viewers from two locations:

- From the **Progress Application Server: <instance name>** page, click **Catalina Log File Viewer** in the **Command and Control** section.
- Click **Log File Viewer** at the top of the **Log File Monitor for: Collection pas catalina** page.

The following information helps you to use the PAS for OpenEdge instance log file viewer:

- At the bottom of the log file viewer page, you can view information about the log file, such as its size, the number of lines present in the log file, and status of the log file.
- Use the **Show** field to control how many PAS for OpenEdge instance log file entries appear at one time. The number entered in the **Show** field must not be less than 10.
- Use the **Overlap** field to control how many entries are repeated from screen to screen.

Note: The value in the **Overlap** field must be less than the number in the **Show** field minus one. For example, if you show 30 entries, you can overlap only 29 or fewer of them.

- After changing the values in the **Show** or **Overlap** fields, click **Reload** or press **ENTER**.
- Click **First** to display the first *x* entries, where *x* is the value in the **Show** field.
- Click **Prior** to display the previous *x* entries, where *x* is the value in the **Show** field.
- Click **Next** to display the next *x* entries, where *x* is the value in the **Show** field.
- Click **Last** to display the final *x* entries, where *x* is the value in the **Show** field.
- Use the **Go to** field to control the display of the log file in the viewer based on the line number in the log file. For example, if you enter the value `10` in the **Go to** field and press **ENTER**, the log file is displayed from the 10th line number entry.
- By default, the log file viewer entries are in the ascending order of the line number. You can select **Descending** in the **Display** field to change the order of the display.

Installing other Web applications shipped with PAS for OpenEdge

You can install the Web applications that are shipped with PAS for OpenEdge from the **Progress Application Server: <instance name>** page. This option is beneficial if you have not installed these Web applications while creating the PAS for OpenEdge instance or if you have previously undeployed these Web applications.

To install other Web applications shipped with PAS for OpenEdge:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. In the **Command and control** section of the page, click **Extras**. The **Extras** page appears.

   For the Web applications that are already installed on the PAS for OpenEdge, a notification is displayed.

4. If available, select the following options to install the Web applications:

   - Select the **Install Progress Application Server OpenEdge manager web application** option to install the OpenEdge manager web application, `oemanager`.
   - Select the **Install OpenEdge ABL web application** option to install the OpenEdge ABL Web application, `oeabl`.

- Select the **Install Tomcat manager web application** option to install the Apache Tomcat manager Web application, `manager`.
- Select the **Install Tomcat host manager web application** option to install the Apache Tomcat host manager Web application, `host-manager`.

5. Click:
   - **Save** to install the selected Web applications on the PAS for OpenEdge instance.
   - **Cancel** to cancel any changes.

Configuring a PAS for OpenEdge instance

You can set the configuration properties of a PAS for OpenEdge instance using the **Configure Progress Application Server Properties** page.

To set PAS for OpenEdge configuration properties:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.
2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.
3. In the **Command and Control** section, click **Configuration**.

   The **Configure Progress Application Server Properties** page appears with the following tabs:
   - **Connection Configuration** — Use this tab to configure the properties required for the OpenEdge Management to communicate with the Pacific Application Server.
   - **Server Configuration** — Use this tab and the other tabs under it to configure the server properties that are stored in the `catalina.properties` and `appserver.properties` files.

Connection configuration properties

Use the **Connection Configuration** tab to configure the properties that allow the OpenEdge Management to connect to the Progress Application Server. For example, you can edit the PAS for OpenEdge URL, port information, Tomcat and OpenEdge manager credentials, and so on.

Property	Default value	Description
URL	–	Specify the URL to connect to OpenEdge Manager.
No host verification for SSL	–	Select the check box to skip verification for SSL connections.
Description	–	Specify a description for the connection.
Tomcat manager login	`tomcat`	Specify the login ID for Tomcat manager.
Tomcat manager password	–	Specify the password for the Tomcat manager login ID you provided.

Property	Default value	Description
OpenEdge manager login	`tomcat`	This field is pre-filled with the default username from the `webUserName` property of the `pasmgr.properties` file located in the `...\Progress\OpenEdge\properties` in the machine where Progress OpenEdge is installed. You may choose to change the **OpenEdge manager login** either from the **Configuration** page or from the `pasmgr.properties` file.
OpenEdge manager password	–	This field is pre-filled with the default password from the `webPassword` property of the `pasmgr.properties` file located in the `...\Progress\OpenEdge\properties` in the machine where Progress OpenEdge is installed. You may choose to change the **OpenEdge manager password** either from the **Configuration** page or from the `pasmgr.properties` file.
Autostart		Select the check box to automatically start the PAS for OpenEdge instance
HTTPS enabled protocols	`TLSv1.2`	Select one or more cryptographic protocols for secure communications. By default, PAS for OpenEdge supports `TLSv1`, `TLSv1.1`, and `TLSv1.2`.
HTTPS enabled cipher suites	–	Specify comma-separated list of cipher suite values to manually set cipher suites. By default, OpenEdge Management and OpenEdge Explorer support all the cipher suites that are provided by the SSL implementation of the Java Secure Socket Extension (JSSE).

Optionally, you can click **TEST CONNECTION** to validate the new or updated properties and ensure that the connection is successful. If the configuration properties are valid, the **Test Status** dialog box appears with the **Test Passed successfully** message. And if the properties are invalid, the **Error** dialog box appears with an error message.

Server configuration properties

Use the following tabs under **Server Configuration** to configure the server properties that are stored in the `catalina.properties` and `appserver.properties` files:

- HTTPS
- HTTP
- JMX
- AJP13 Connector
- Server Options
- Request Options
- Security Options

- Health Check

HTTPS properties

Use the **HTTPS** tab to enable the HTTPS connector for Progress Application Server.

Property	Default value	Description
HTTPS connector	`ON`	Use the **ON**/**OFF** toggle switch to enable or disable the HTTPS connector respectively.
Port	`8443`	Specify the HTTPS protocol connector port number.
Connection timeout (milliseconds)	`20000`	Specify the time, in milliseconds, to wait between the establishment of a TCP (SSL/TLS) connection by a client and the arrival of the first HTTPS request.
SSL session timeout (seconds)	`86400`	Specify the time, in seconds, after which the SSL session ends.
Server key alias	`test`	Specify the alias name of the keystore entry holding the server's private key and public key certificate.
Server key password	`password`	Specify the password to use when accessing the SSL/TLS keystore.
Store type	`PKCS12`	Specify the type of Java keystore format used by tomcat-keystore.
Max queue size	`100`	Specify the maximum size of HTTPS connector message queue.
Max connections	`-1`	Specify the maximum number of TCP socket connections per HTTPS connector.
Client authentication	`false`	Select whether you want to enable or disable SSL client authentication by the HTTPS connector. The options are true, false, and want. The default is false.
Certificate trust type	`JKS`	Specify the type of Java certificate store format used by tomcat-certstore.jks that holds the root or intermediate CA certificates. These certificates are used to validate the clients using SSL client authentication.
Certificate store password	`password`	Specify the Java certificate store password used to access tomcat-certstore.jks that holds the root or intermediate CA certificates. These certificates are used to validate the clients using SSL client authentication.
Encryption protocol	`TLSv1.2`	The HTTPS protocol selected for secure communication. This value is not editable in OpenEdge Management for security purposes. However, if required, you can edit this value in the catalina.properties file.

Property	Default value	Description
Enabled ciphers	ALL	The list of cipher suites enabled for secure communication. This can either be all the cipher suites or a comma-separted list of cipher suites supported by JSSE.
Bind on init	false	Select the check box to control when to bind the socket used by the connector.
Compression	on	Select the check box to enable GZIP compression for HTTPS transports.

HTTP properties

Use the **HTTP** tab to enable the HTTP connector for Progress Application Server.

Property	Default value	Description
HTTP connector	ON	Use the **ON**/**OFF** toggle switch to enable or disable the HTTP connector respectively.
Port	8090	Specify the HTTP protocol connector port number.
Connection timeout (milliseconds)	20000	Specify the time, in milliseconds, to wait between the establishment of a TCP connection by a client and the arrival of the first HTTP request.
Max connections	-1	Specify the maximum number of TCP socket connections per HTTP connector.
Max queue size	100	Specify the maximum size of HTTP connector message queue.
URI encoding	ISO-8859-1	Specify the character encoding used to decode the URI bytes after decoding the URL. Note that this value is set to Tomcat's default value, and it impacts both the HTTP and HTTPS connectors.
Bind on init	false	Select the check box to control when to bind the socket used by the connector.
Compression	on	Select the check box to enable GZIP compression for HTTP transports.

JMX properties

Use the **JMX** tab to configure the network settings for administering Progress Application Server using a remote JMX console.

Property	Default value	Description
JMX	OFF	Use the **ON**/**OFF** toggle switch to enable or disable administration using remote JMX console.

Property	Default value	Description
Registry port	`10001`	Specify the TCP port number used for the RMI registry for the server's JMX MBeans.
Server port	`10002`	Specify the TCP port number used to access the JMX/RMI server and access the server's JMX MBeans.
JMX remote access file	`${CATALINA_BASE}/conf/jmxremote.access`	Specify the location of the JMX remote access control file for the remote JMX port.
JMX password file	`${CATALINA_BASE}/conf/jmxremote.password`	Specify the location of the file where the password for remote JMX port is stored.

AJP13 Connector properties

Use the **AJP13 Connector** tab to enable the AJP13 network connector for Progress Application Server, and thereby achieve load balancing by an Apache httpd or IIS Web server.

Property	Default value	Description
AJP13	`OFF`	Use the **ON**/**OFF** toggle switch to enable or disable the AJP13 network connector.
Port	`8009`	Specify the port number for the AJP13 Apache server connector.
Max connections	`-1`	Specify the maximum number of TCP socket connections per AJP13 connector.
Max queue size	`100`	Specify the maximum size of AJP13 connector message queue.
Tomcat authentication	`true`	Select the check box to pass the tomcat authenticated user id or parent authenticated user id to the web application.

Miscellaneous settings

Use the **Server Options** tab to configure miscellaneous properties for Progress Application Server.

Property	Default value	Description
Shutdown port	–	Specify the port number to shutdown the server that is running. The values can range from `1024` to `65535`. To disable the shutdown port, set the value to `-1`. You must specify a value in Windows, but this is optional in UNIX.

Property	Default value	Description
Shutdown password	`SHUTDOWN`	Specify a private shutdown port access code to prevent one server instance from being shutdown by anyone. **Note:** If the shutdown port is specified, you must change this value to avoid insecure configuration.
Stuck session valve	`ON`	–
Stuck thread threshold (seconds)	`600`	Specify the maximum amount of time, in seconds, an active HTTP request can be running before it is considered as HUNG and is reported in the server log file.
Max threads	`300`	Specify the maximum number of OS process threads the PAS server can use.
Minimum spare threads	`10`	Specify the number of spare threads the server will reserve for future client operations.
Web application directory	`webapps`	Specify the location of the directory where the web applications are to be deployed. If not specified, the Tomcat's default directory `webapps` is used. If a relative path is specified, it will be relative to the instance's root directory. If an absolute path is used, it must conform to a single platform type.
Common library path	–	Specify a comma-delimited list of library paths. The delimiter is platform dependent, and the format of the path must conform to the Tomcat's common library path format (https://tomcat.apache.org/tomcat-9.0-doc/class-loader-howto.html).
Auto deploy WAR files	`false`	Select the checkbox to enable the security settings used by the tomcat server for deploying web applications. The default value depends on whether the PAS instance is created as a development or a production security configuration. In a production configuration, the default is `false`. In a development configuration, the default is `true`.
Unpack WAR files	`true`	Select the checkbox to allow unpacking of web archive (.war) files when the PAS instance is started.
Version logger	`ON`	Use the **ON**/**OFF** toggle switch to enable or disable logging the command line arguments passed to Java when Tomcat is started.
Log command line arguments	`true`	Select the checkbox to log the command line arguments passed to Java when Tomcat is started.
Log environment variables	`false`	Select the checkbox to log the current environment variables when Tomcat is started.
Log java system properties	`false`	Select the checkbox to log the current Java system properties when Tomcat is started.

Client request settings

Use the **Request Options** tab to configure client request settings for Progress Application Server.

Property	Default value	Description
Max POST request size (bytes)	`2097152`	Specify the maximum size, in bytes, of a connector's POST message body.
Tomcat session timeout (seconds)	`1800`	Specify the maximum amount of time, in milliseconds, that can elapse between the client HTTP requests before the client's HTTP session expires and is no longer usable
Max pipeline requests	`100`	Specify the maximum number of pipelined HTTP keepalive requests before the TCP socket to the client is closed.
Message timeout (milliseconds)	`10000`	Specify the maximum time, in milliseconds, to wait for async messages to complete.
Compression minimum size (bytes)	`2048`	Specify the minimum size, in bytes, of message that will be compressed, in bytes. This is applicable to both HTTPS and HTTP transports.
Compression MIME types	`text/html, text/xml, text/javascript, text/css, application/json`	Specify a comma-separated list of MIME types that must be compressed. This is applicable to both HTTPS and HTTP transports.

Security settings

Use the **Security Options** tab to configure security settings for Progress Application Server.

Property	Default value	Description
Tomcat user file path	`conf/tomcat-users.xml`	Specify the name and location of the local user account file. You can change this to be external to the server instance's directory tree so that it is not overwritten by incremental updates.
Lockout removal warning time (seconds)	`3600`	Specify the time limit, in seconds, before which the Lockout Realm should log when it removes a lockout cache entry.
User lockout cache size	`1000`	Specify the cache size of locked out users the Lockout Realm maintains.
Lockout failure count	`5`	Specify the number of failed login attempts a user can perform before the Lockout Realm locks out that user account.
Lockout failure timeout (seconds)	`300`	Specify the time, in seconds, the Lockout Realm keeps a user account locked out.

Health check properties

Use the **Health Check** tab to configure health check properties for Progress Application Server.

Property	Default value	Description
Health Check	`OFF`	Use the **ON**/**OFF** toggle switch to enable or disable the Health Check feature.
Port	`8899`	Specify the HTTP port number on which the health check web application should run.
Application directory	`hcapps`	Specify the directory where the health check web application is deployed. This directory is an alternative for the `webapps` folder. If you change this directory, you must deploy the web application to the updated directory.
Auto deploy	`false`	Select the checkbox to enable the security settings used by the tomcat server for deploying web applications. This is turned off by default to avoid starting a running application by simply copying the web archive (.war) into the deployment directory.
Unpack WAR files	`true`	Select the checkbox to allow unpacking of web archive (.war) files when the PAS instance is started.

Register a PAS for OpenEdge instance as a service

Just like you can manage the AdminServer (**AdminService for OpenEdge <*version number*>**) as a service in the Services view of Microsoft Management Console, you can register and manage a PAS for OpenEdge instance as a service as well.

Note: Registering a PAS for OpenEdge instance as a service is applicable only for Windows platforms.

You cannot register a *development* instance of PAS for OpenEdge as a service. However, if a development instance was registered as a service in previous release of OpenEdge, you can unregister that instance as a service.

To register a PAS for OpenEdge instance as a service:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type a PAS for OpenEdge instance name in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. In the **Command and control** section of the page, click **Register as service**.

   The **Register PAS as Service** page appears.

4. Click **Register as Service** to register the PAS for OpenEdge as a service.

You can now start or stop the PAS for OpenEdge instance in the Services view of Microsoft Management Console. Note that the **Start** and **Stop** buttons in the **Progress Application Server: <instance name>** page changes to **Start as windows service** and **Stop as windows service**, respectively.

5. Optionally, you can select **Unregister as service** on the **Progress Application Server: <instance name>** page to unregister the PAS for OpenEdge instance as a service.

2

Managing OpenEdge ABL applications deployed to a PAS for OpenEdge instance

A PAS for OpenEdge instance can have several deployed OpenEdge ABL Web applications on it. An OpenEdge ABL Web application is hosted on an Application Server (OpenEdge AppServer or PAS for OpenEdge) and is accessible to a client application through a Web server. OpenEdge also supports REST, SOAP, APSV, and WEB transport services for accessing the ABL application logic.

For more information on creating and deploying OpenEdge ABL Web applications, see *Progress Developer Studio Online Help*.

Note: In the default PAS for OpenEdge configuration, the `ROOT` application is the default OpenEdge ABL application.

To manage deployed OpenEdge ABL applications, a PAS for OpenEdge instance provides **ABL Applications** section. You must install only the Web application you want to manage.

Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

You can also collect metrics to view count and time-related information about the ABL applications over a period. Examples of such information include the number of requests made and average response time of the Web application. To manage the collection of metrics, set the OpenEdge ABL application-specific `collectMetrics` property in the `openedge.properties` file that is available in the `<PAS-instance-dir>\conf` folder.

OpenEdge Management and OpenEdge Explorer provides collection viewlets using which you can monitor PAS for OpenEdge resources. You can select a collection viewlet from the **Collections** list in the **Dashboard** menu to view graphs for resources such as requests sent to the ABL application, requests to the ABL application transports, and CPU usage by ABL agents and Web server. If you rest your mouse pointer on plotted graph line for a resource, you can view a pop-up that displays the resource ID and x and y-axis values at that point on the graph. For more information on creating and managing collection viewlets, see the *Collection views* section in *OpenEdge Management: Resource Monitoring*.

This section discusses all the tasks you require to manage OpenEdge ABL applications deployed on your PAS for OpenEdge instance.

For details, see the following topics:

- Managing OpenEdge ABL application settings
- Managing PAS for OpenEdge transport services

Managing OpenEdge ABL application settings

The functions of the log file monitors and viewers for ABL applications and agents are similar to the functions of the Catalina log file monitor and viewer. For more information about the Catalina log file monitor and viewer, see the Customizing PAS for OpenEdge log file monitors on page 30 and Using PAS for OpenEdge log file viewers on page 31 sections.

The **ABL Application: <ABL application name>** page also shows the graphs that display the usage of CPU, physical, and virtual memory for each available PAS agent. Using the options on the top-right corner of the graph, you can select a time period, ranging from 2 hours to 2 weeks, to display the data collected for that period of time, reload the graph, and select different forms of graphical interpretation.

The following sections describe how you can view and manage OpenEdge ABL application settings from the **ABL Application: <ABL application name>** page:

Viewing OpenEdge ABL application Client Sessions

You can view the OpenEdge ABL agent sessions by using the **ABL Application: <ABL application name>** page.

To view the list of sessions:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

Note: To enable the OpenEdge ABL appplications and transport services while the PAS for OpenEdge instance is running, ensure that the OpenEdge manager web application, `oemanager.war` (available at `<$DLC>\servers\pasoe\extras\oemanager.war`) is installed on PAS for OpenEdge. If it is not deployed, then deploy it from the WEB APPLICATIONS interface.

4. Click **Client Sessions**.

   The **Client Sessions for ABL Application < ABL application name>** page appears. You can view the following ABL application session properties:

   Table 2: OpenEdge ABL application session properties

Property	Description
ABL ID	Unique identity of the ABL agent
Client	Identity (IP address) of the client that is performing the request
Request State	State of the OpenEdge ABL agent request (INIT, READY, READING, RUNNING, WRITING, DISCONNECT)
Bound	Set to true if the client request is bound to the session
Session State	State of the OpenEdge ABL agent session (INIT, STARTING, AVAILABLE, RESERVED, STOPPING, STOPPED)
Session Type	Type of the OpenEdge ABL agent session (SESSION_MANAGED, SESSION_FREE)
Adapter Type	The adapter (HTTP, SOAP, or REST) used to make the client request
Last Access	The time at which the OpenEdge ABL session was accessed
Last Request Time	The time at which an OpenEdge ABL session request was made
Session Pool ID	Manages instances of session objects.
Session ID	The ID of the session within the agent process
Agent ID	OpenEdge ABL service name.

   If an ABL ID has no client session(s) available, the result set displays an empty list with a message that there are no client sessions available.

   To filter from a list of sessions, begin typing a client name in the **Client** field to view the sessions for that client. You can also filter the list by selecting from the **Request state**, **Session state**, **Session type**, or **Adapter type** list and then clicking the **Refresh** icon to view the sessions for the selected options.

Note: You can sort the columns in ascending or descending order by clicking the arrow next to the selected column name and then clicking **Sort Ascending** or **Sort Descending**, respectively. You can also select or deselect a list of columns by clicking **Columns**.

5. Optionally, select any agent session from the list and click **Stop** to stop the session.

Note: To stop an ABL agent session, you must have execute permissions (start and stop) that are granted by the administrator.

Viewing OpenEdge ABL application Agent Sessions

You can view OpenEdge ABL agent sessions by using the **ABL Application: <ABL application name>** page. To view the list of threads:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

Note: To enable the OpenEdge ABL appplications and transport services while the PAS for OpenEdge instance is running, ensure that the OpenEdge manager web application, `oemanager.war` (available at `<$DLC\servers\pasoe\extras\oemanager.war`) is installed on PAS for OpenEdge. If it is not deployed, then deploy it from the WEB APPLICATIONS interface

4. Click **Agent Sessions**.

   The **OpenEdge ABL Agent Sessions for <ABL application name>** page appears. You can view the following ABL agent properties:

Table 3: OpenEdge ABL agent properties

Property	Description
PID	The process ID of the server.
Agent ID	OpenEdge ABL service name.
Agent State	State of the agent (AVAILABLE or STOPPED)
Session ID	The ID of the session within the agent process
Session State	State of the OpenEdge ABL agent session (INIT, STARTING, AVAILABLE, RESERVED STOPPING, STOPPED)

Property	Description
Session Start Time	The Date and Time when the session started
Session External State	Returns 1 when the session is executing outside of the AVM in a dll or shared library. Returns 0 otherwise.
Session Memory	The amount of memory allocated by the session
Stack trace link	Displays the agent stack trace information such as Session Information, Databases, Persistent procedures, ABL objects and Callstack for reference to debug any issues.

If a PID has no agent session(s) available, the result set displays an empty list with a message that there are no agent sessions available.

You can filter from a list of agents by selecting an agent state from the **Agent State** list and then clicking the **Refresh** icon to view the agents for that state.

Note: You can sort the columns in ascending or descending order by clicking the arrow next to the selected column name and then clicking **Sort Ascending** or **Sort Descending**, respectively. You can also select or deselect a list of columns by clicking **Columns**.

5. To stop one or more agents, select the agent(s) from the list and click **Stop**.

   Specify the following in the **Stop agent for <pas application name>** window that appears:

   - **Wait to finish** — The time, in milliseconds, the agent waits to complete the requests that are currently running before stopping gracefully.

     After waiting, if **Wait after stop** is set to 0, the agent is stopped forcibly.

   - **Wait after stop** — The time, in milliseconds, the agent waits after completing **Wait to finish** and before stopping forcibly.

   Click **STOP** to stop the selected agent(s).

Note: To stop an ABL agent, you must have execute permissions (start and stop) that are granted by the administrator.

Viewing OpenEdge ABL application requests

You can view the OpenEdge ABL application requests by using the **ABL Application: < ABL application name>** page. To collect the time and count metrics related to the requests, set the AppServer-specific `collectMetrics` property to `3` in the `openedge.properties` file.

Note: Until you reset statistics or until you set the `collectMetrics` property to `3` in the `openedge.properties` file, the agent continues to track both active and completed requests. This tracking can consume a lot of memory.

To view the list of requests:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

---

4. Click **Requests**.

   The **OpenEdge ABL Agent Requests** page appears. You can view the following ABL agent request properties:

   Table 4: OpenEdge ABL agent request properties

Property	Description
Request ID	Unique identity of the request
Request State	State of the OpenEdge ABL agent request (INIT, READY, READING, RUNNING, WRITING, DISCONNECT)
Request Elapsed Time	The time since the previous client request has been made
Request Start Time	The time at which the client request has been made
Session ID	Numeric ID that uniquely defines the session

   You can filter from a long list of requests by selecting an request state from the **Request State** list and then clicking the **Refresh** icon to view the requests for that state.

Note: You can also sort the columns in ascending or descending order by clicking the arrow next to the selected column name and then clicking **Sort Ascending** or **Sort Descending**, respectively. You can also select or deselect a list of columns by clicking **Columns**

---

5. Optionally, select any agent request from the list and click **Stop** to stop the request.

Note: To stop an ABL agent request, you must have execute permissions (start and stop) that are granted by the administrator. You can stop only one request at a time.

Viewing OpenEdge ABL application connections

You can view the OpenEdge ABL application connections by using the **ABL Application: <ABL application name>** page.

To view the list of connections:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

   Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

4. Click **Connections**.

   The **Client connections for ABL Application <ABL application name>** page appears. You can view the following ABL application client connection properties:

   Table 5: OpenEdge ABL application client connection properties

Property	Description
Client	Identity (IP address) of the client that is performing the request
Adapter Type	The adapter (HTTP, SOAP, or REST) used to make the client request
Request Start Time	The time at which the client request has been made
Elapsed time	The time since the client request has been made
Procedure	The ABL procedure invoked by the client
URL	The URL used by the client to perform the request

   To filter from a list of connections, begin typing a client name in the **Client** field to view the connections for that client. You can also filter from the list by selecting an adapter type from the **Adapter type** list and then clicking the **Refresh** icon to view the connections for that adapter type.

   Note: You can also sort the columns in ascending or descending order by clicking the arrow next to the selected column name and then clicking **Sort Ascending** or **Sort Descending**, respectively. You can also select or deselect a list of columns by clicking **Columns**.

Viewing OpenEdge ABL application metrics

You can view OpenEdge ABL application metrics by using the **ABL Application: <ABL application name>** page.

To view the list of metrics:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

   Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

4. Click **Application Metrics**.

   The **PAS Application metrics for <ABL application name>** page appears. You can view the following PAS application metric properties:

   Table 6: PAS Application Metric Properties

Property	Description
Access Time	The date and time at which the metrics have been retrieved
Last Reset Time	The date and time at which you have last reset the metrics for the service.
Requests	A group of various requests made to the PAS application for accessing the AppServer
Errors	A group of various errors encountered from the time the PAS for OpenEdge has started or last reset.
Time	A group of various time components

   You can reset the time and count values of PAS application metrics to zero by clicking **Reset statistics**. The **Last Reset Time** field is updated when the statistics are reset.

   The **PAS Application metrics for <ABL application name>** page also shows the graphs that display the connections, sessions, concurrent connected clients for PAS application. Using the options on the top-right corner of the graph, you can select a time period, ranging from 2 hours to 2 weeks, to display the data collected for that period of time, reload the graph, and select different forms of graphical interpretation.

Configuring an OpenEdge ABL application

You can set configuration options for customizing an OpenEdge ABL application. These options are discussed in the following sections.

- Configuring the startup environment on page 51
- Configuring OpenEdge ABL application logging on page 53
- Configuring tuning parameters and process limits on page 54
- Configuring a network environment on page 56
- Configuring advanced settings on page 57

Configuring the startup environment

You can configure your OpenEdge ABL application startup environment using the **ABL Application Configuration for <ABL application name>** page.

To configure your startup parameters and environment:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application in **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

   Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

4. Click **Configuration**.

   The **Application configuration for <ABL application name>** page appears.

5. Select the **Startup Parameters and Environment** tab to set the following properties:

   Table 7: Startup Parameters and Environment

Parameter	Description
Agent working directory	Specify the location of the working directory of the OpenEdge ABL agent.

Parameter	Description
Agent startup parameter	Optionally, specify the parameter value to be passed as part of the session startup procedure. You can configure the PAS for OpenEdge to pass a set of parameters to any agents that it starts. These are standard OpenEdge client startup parameters, and can include any parameters that you require for each PAS for OpenEdge session, including (but not limited to) all the standard database, code-page, and process-management parameters. For more information, see *OpenEdge Deployment: Startup Command and Parameter Reference*.
PROPATH	Optionally, specify the semicolon-separaterd list of PROPATH entries for the agent. These are directories in which the PAS for OpenEdge can locate ABL procedures to execute. This setting overrides any PROPATH environment variable settings on the Progress Application Server when it starts up. Make sure that all the ABL procedures (r-code or source) that you want the PAS for OpenEdge to execute are located in one of these PROPATH directories. Otherwise, the procedure must be executed using its fully qualified pathname.
Activate procedure	Optionally, specify the name of the procedure that activates a session. This procedure executes immediately before a remote procedure request when the connection is in the unbound state. A typical use of Activate procedures is to retrieve the connection context using an application-specific context database. For more information, see *OpenEdge Application Server: Developing AppServer Applications*.
Deactivate procedure	Optionally, specify the name of the procedure that deactivates a session. This procedure executes immediately after remote procedure and delete procedure requests when the connection is in the unbound state. A typical use of the Deactivate procedures is to store connection context using an application-specific context database. For more information, see *OpenEdge Application Server: Developing AppServer Applications*.
Connect procedure	Optionally, specify the connect procedure that you want to execute before a connection request is accepted. This procedure executes in the AppServer agent that handles the client connection request. If the Connect procedure completes with no error, the connection request from the client application is accepted. If the Connect procedure returns an error, the connection request is rejected.
Disconnect procedure	Optionally, specify the disconnect procedure that you want to execute before a connection is terminated. This procedure executes in the AppServer agent that handles a client disconnection request. You can use the AppServer Disconnect procedure to specify logic that you want executed at the time of client disconnection.
Session startup procedure	Optionally, specify the name of the procedure that executes as a PAS for OpenEdge session starts up.

Parameter	Description
Startup procedure parameter	Optionally, specify a value for the startup procedure. You can set this parameter to any arbitrary value. If you do not specify a value for this property, the parameter is set to the Unknown value (?) when the AppServer agent executes the startup procedure.
Shutdown procedure	Optionally, specify the name of the procedure that executes just before a PAS for OpenEdge session shuts down. Unlike the Startup procedure, the Shutdown procedure is run as a non-persistent procedure, and errors propagated by the Shutdown procedure are ignored. The PAS for OpenEdge agent terminates immediately after the Shutdown procedure executes. For more information, see *OpenEdge Application Server: Developing AppServer Applications*.
Startup procedure	Optionally, specify the name of the procedure that executes as a PAS for OpenEdge agent starts up.
Startup procedure parameter	Optionally, specify a value for the agent's startup procedure.
Shutdown procedure	Optionally, specify the name of the procedure that executes just before a PAS for OpenEdge agent shuts down.

6. Click:
   - **Save** to save the changes.
   - **Reset** to apply the last-saved settings.
   - **Cancel** to reject the recent changes.

Configuring OpenEdge ABL application logging

You can configure your ABL application logging by using the **Application configuration for <ABL application name>** page.

To configure your log settings:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application in **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

4. Click **Configuration**.

   The **Application configuration for <ABL application name>** page appears.

5. Select the **Log Settings** tab to set the following properties:

   Table 8: Log settings

Parameter	Description
Log file name	A file where OpenEdge ABL application logging information must be written.
Logging level	The level of detail that is required when logging information in the log file.
Log entry types	Comma-separated values of the log entry types.
Number of log files	The number of log files to use for rollover for an OpenEdge ABL application.
Sever log threshold	A size in bytes for log file rollover for the OpenEdge ABL application.

6. Click:

   - **Save** to save the changes.
   - **Reset** to apply the last-saved settings.
   - **Cancel** to reject the recent changes.

Configuring tuning parameters and process limits

You can configure tuning parameters and process limits of your OpenEdge ABL application using the **Application configuration for <ABL application name>** page.

To set tuning parameters and process limits:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application in **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

4. Click **Configuration**.

   The **Application configuration for <ABL application name>** page appears.

5. Select the **Performance Tuning and Limits** tab to set the following performance tuning and process limits:

Table 9: Tuning and process limits parameters

Parameter	Default value	Description
Minimum number of agents	0	Specifies the minimum number of agents expected at any time. If the number of agents drops below this number, a client request will cause the session manager to start enough agents to equal this number.
Maximum number of agents	10	Specifies the maximum number of concurrent multi-session agents that can run on the AppServer.
Initial number of agents	1	Specifies the number of multi-session agents to start when AppServer is started.
Initial number of sessions	0	Specifies the number of sessions associated with a PAS for OpenEdge instance when it starts.
Maximum connections per agent	16	Specifies the maximum number of TCP connections between session manager and multi-session agent.
Maximum ABL sessions per agent	200	Specifies the maximum number of concurrent sessions a multi-session agent can run concurrently.
Maximum number of agents to start	0	Determines the maximum number of agents that may be started at the same time. If this value is zero or not defined, then no limit is enforced.
Request wait timeout	15000	Specifies the maximum amount of time, in milliseconds, a request will wait in the session manager for a connection to be available before rejecting the request.
Connection wait timeout	3000	Specifies the maximum amount of time, in milliseconds, that the session manager will wait for a connection to be freed up before creating a new connection.
Idle agent timeout	300000	Timeout value in milliseconds for an idle agent. If an agent is idle for more than the specified timeout value, then the session will be deleted when idleResource cleanup is done.

Parameter	Default value	Description
Idle connection timeout	300000	Timeout value in milliseconds between an AppServer client and the Session Manager. If a connection is idle for more than the specified timeout value, then the SessionManager will terminate the connection by automatically disconnecting the connection from the AppServer.
Idle resource timeout	0	Timeout value in milliseconds to determine the frequency with which the PASOE server checks for idle resources. Any resource (e.g. connection, agent, or client session) that has not been accessed more recently than the specified timeout for that property will be terminated. If this property is set to zero, then idle resource checking is disabled.
Idle session timeout	300000	Specifies the timeout value in milliseconds for an idle SessionManager session. If a SessionManager session is idle for more than the specified timeout value, then the session will be deleted when idleResource cleanup is done.
Agent listener timeout	10000	Specifies the maximum amount of time, in milliseconds, the session manager will wait for a multi-session agent to start up on connect.

6. Click:
   - **Save** to save the changes.
   - **Reset** to apply the last-saved settings.
   - **Cancel** to reject the recent changes.

Configuring a network environment

You can configure your OpenEdge ABL service network environment using the **Application configuration for <ABL application name>** page.

To configure your network environment:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application in **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

   Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

4. Click **Configuration**.

   The **Application configuration for <ABL application name>** page appears.

5. Click the **Network Environment** tab to set the following properties:

   Table 10: Network environment

Parameter	Description
Server minimum port	Specify an agent minimum port number. The range of port numbers is 0 to 65535.
Server maximum port	Specify an agent maximum port number. The range of port numbers is 0 to 65535.
IP version	Specify the IP version to be used for network communication. • IPv4 to allow connections only with IPv4. • IPv6 to allow connections with IPv6 version and mapped IPv4. This is the default IP version.

6. Click:
   - **Save** to save the changes.
   - **Reset** to apply the last-saved settings.
   - **Cancel** to reject the recent changes.

Configuring advanced settings

You can configure your OpenEdge ABL application's advanced settings using the **Application configuration for <ABL application name>** page.

To configure advanced settings:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application in **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

   Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

4. Click **Configuration**.

   The **Application configuration for <ABL application name>** page appears.

5. Select the **Configure advanced settings** tab to set the following properties:

   Table 11: Advanced Settings

Parameter	Description
Server executable file	Optionally, specify the name of the agent executable.
Thread Safe External Libraries	Select **Yes** to mark all the ABL application external libraries (DLL) as thread-safe. Use this setting when all the external libraries have already been certified as thread-safe.
Lock All Non Thread Safe External Libraries	Select **Yes** to lock all the ABL application external libraries that are not thread-safe. This setting also ensures that only one external library is accessed at a time. Use this setting to stop an external library from loading another library.
Lock All External Libraries	Select **Yes** to lock all the ABL application external libraries. This setting also ensures that only one external library is accessed at a time. Use this setting to debug and ensure that a non thread-safe external library has not been defined as a thread-safe external library.
Collect statistics data	Select the check box to collect the PAS for OpenEdge-related statistics data. By default, this is not selected.
Flush statistics data	Select the check box to remove the collected statistics data. By default, this is not selected.

6. Click:

   - **Save** to save the changes.
   - **Reset** to apply the last-saved settings.
   - **Cancel** to reject the recent changes.

Managing PAS for OpenEdge transport services

An OpenEdge ABL Web application is hosted on an Application Server (OpenEdge AppServer or PAS for OpenEdge) and is accessible to a client application through a Web server. OpenEdge also supports REST, SOAP, APSV, and WEB transport services for accessing the ABL application logic.

You can view and manage PAS for OpenEdge transport services from the **ABL WebApp: <webapp name>** page.

Note: SOAP based Web services are not supported by OpenEdge Management in 12.0.

For more information, see

- Managing REST transports on page 59
- Managing SOAP transports on page 66
- Managing APSV transports on page 73
- Managing WEB transports on page 77

Managing REST transports

You can manage and configure your REST transport services using the **ABL WebApp: <webapp name>** page.

To manage the REST transport services deployed on PAS for OpenEdge:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

   Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

4. Click an ABL Web application listed in the **ABL WebApps** section.

   The **ABL WebApp: <webapp name>** page appears. You can view the following information about the REST transport:

Table 12: REST Transport Summary

Field	Description
State	State of the REST transport (ENABLED or DISABLED)
Access URL	When working with HTTP, use this URL to access the REST transport service in the `http://<host_name>:<port>/rest` format
Secure Access URL	When working with HTTPS, use this URL to access the REST transport service in the `https://<host_name>:<port>/rest` format
Total # services	The number of REST Web services deployed

5. Click **Deploy** under the **REST Transport** section.

   The **Deploy REST application paar file** page appears.

6. Click **Browse** to locate an **Application .paar file**. You can change the file name by providing an **Application name**. The name can contain only alphanumeric characters. It must not contain blank spaces or special characters, except underscore ("_") and hyphen ("-").

7. Click **Deploy** to deploy the Web service on the PAS for OpenEdge instance.

   A success message appears if the application is successfully deployed on the PAS for OpenEdge, else you must troubleshoot any errors in your REST `.paar` file or the OpenEdge Management or OpenEdge Explorer instance, and then try deploying again.

   Note: Click the refresh icon to see the latest status in the list of REST Web services deployed on the PAS for OpenEdge instance.

8. Click **Enable** in the **REST Transport** section to enable the REST transport. Click **Yes** in the confirmation dialog that appears.

   The enabled state is displayed in the **REST Transport Summary** section.

9. To disable the REST transport service, click **Disable** in the **REST Transport** section. Click **Yes** in the confirmation dialog that appears.

   The disabled state is displayed in the **REST Transport Summary** section.

10. To undeploy a REST Web service from the PAS for OpenEdge instance, select the deployed REST Web services from the list, and then click **Undeploy**.

    A success message appears if the application is successfully undeployed from PAS for OpenEdge.

    Note: You can only undeploy a disabled Web service.

Configuring REST transport run time properties

Each PAS for OpenEdge instance can have a different transport mechanism for all its deployed services based on sessions, timeouts, and other advanced properties. You can configure your REST transport run time properties using the **REST transport configuration for ABL Application <ABL application name> WebApp <webapp name>** page.

To set REST transport run time properties:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

   Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

4. Click an ABL Web application listed in the **ABL WebApps** section.

   The **ABL WebApp: <webapp name>** page appears.

5. Click **Run-time Properties** under the **REST Transport** section.

   The **REST transport configuration for ABL Application <ABL application name> WebApp <webapp name>** page appears.

6. Set the following properties in the **REST Transport Properties** section:

   Table 13: REST transport run time properties

Parameter name	Default value	Description
Initial sessions	1	The number of network sessions to be created (and shared by all clients) when the connection pool for the service is initialized by the adapter.
Minimum sessions	1	The minimum number of connected sessions that the service attempts to maintain in the service connection pool.
Maximum sessions	0	The maximum number of connected sessions allowed in the service connection pool.
Minimum idle connections	0	The minimum number of idle, or inactive, PAS for OpenEdge connections to maintain.
Idle session timeout	0	The duration (in seconds) between attempts by the adapter to shut down extra network connections to the AppServer, based on client demand.
Request wait timeout	-1	Determines how the Web service handles requests when the service connection pool becomes full. -1 signifies that the request is queued indefinitely until a PAS for OpenEdge session becomes available.
Stale OOABL object timeout	0	The maximum duration (in seconds) that a service object can be idle before it is released.

Parameter name	Default value	Description
Connection lifetime	0	The maximum lifetime (in seconds) of PAS for OpenEdge connections in the connection pool for this service.
Service fault level	2	Specifies the degree of detail returned to the client for a service fault.
Wait if busy	Not selected	Determines how to handle client requests to a service that is busy processing a prior request.

7. Click:
   - **Save** to save the changes.
   - **Cancel** to reject the recent changes.
   - **Reset** to apply the last-saved settings.

Configuring REST transport properties

You can configure few properties of REST transport service of a local PAS for OpenEdge instance even when the OpenEdge manager web application, `oemanager`, is not installed and the instance is not running. You can access these properties using the **REST transport configuration for ABL Application <ABL application name> WebApp <webapp name>** page.

Note: To configure the properties of a transport service of a remote PAS for OpenEdge instance, ensure that the `oemanager` is installed and the instance is running.

To configure the REST transport properties:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

4. Click an ABL Web application listed in the **ABL WebApps** section.

   The **ABL WebApp: <webapp name>** page appears.

5. Click **Configuration** under the **REST Transport** section.

   The **REST transport configuration for ABL Application <ABL application name> WebApp <webapp name>** page appears.

6. Set the following properties in the **REST Transport Properties** section:

Table 14: REST transport properties

Parameter name	Default value	Description
Web Application	ROOT	The name of the Web service application.
Adapter Enabled	1	Indicates whether WSA (Web Service Adapter) is enabled.
Status Enabled	1	Indicates whether the REST Web service is enabled for client access.
Allow Runtime Updates	0	Indicates whether run time property changes are allowed for this application server.
Collect metrics	1	Collects count and time related information of REST Web service over a period of time. You can type the following values: • 0 — Do not collect metrics. • 1 — Collect count metrics only. • 2 — Collect time metrics only. • 3 — Collect both count and time metrics.

7. Click:
   - **Save** to save the changes.
   - **Cancel** to reject the recent changes.
   - **Reset** to apply the last-saved settings.

Viewing REST transport metrics

You can view REST transport metrics by using the **REST transport metrics for ABL Application <ABL application name> WebApp <webapp name>** page.

To view REST transport metrics:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

4. Click an ABL Web application listed in the **ABL WebApps** section.

   The **ABL WebApp: <webapp name>** page appears.

5. Click **Metrics** in the **REST Transport** section.

   The **REST transport metrics for ABL Application <ABL application name> WebApp <webapp name>** page appears. You can view the following metrics:

Table 15: REST Transport Metrics

Metric	Metric type	Description
Access Time	-	The date and time at which the metrics have been retrieved
Average AppServer time	Time	The average time spent in returning a response from the AppServer to the REST transport
Average connect time	Time	The average time spent in using the REST transport to connect to the AppServer
Average transport time	Time	The average of the times spent in making a request to the AppServer via the REST transport and in returning a response from the AppServer by the REST transport
Connect requests	Count	The number of requests for which new connections have been made between the REST transport and AppServer
Expression errors	Count	The number of requests in which a parameter value has incorrectly formatted JSON data
Failed requests	Count	The number of requests that have resulted in a failure error message
Last reset time	-	If you click **Reset**, this field displays the date and time at which you have reset the metrics for the service. When the metrics are reset, the **Last Reset Time** field is updated.
Maximum AppServer time	Time	The maximum time spent in returning a response from the AppServer to the REST transport
Maximum connect time	Time	The maximum time spent in using the REST transport to connect to the AppServer
Maximum transport time	Time	The maximum value of the times spent in making a request to the AppServer via the REST transport and in returning a response from the AppServer by the REST transport
Minimum AppServer time	Time	The minimum time spent in returning a response from the AppServer to the REST transport

Metric	Metric type	Description
Minimum connect time	Time	The minimum time spent by the REST transport to connect to the AppServer
Minimum transport time	Time	The minimum value of the times spent in making a request to the AppServer via the REST transport and in returning a response from the AppServer by the REST transport
Requests	Count	The total number of requests (successful or failed) made to the REST transport for access to the AppServer
Run requests	Count	The number of requests for which an ABL class, function, or procedure has been invoked on the AppServer
Service not found	Count	The number of requests for which a REST service has not been found
Service unavailable requests	Count	The number of requests made during which the REST transport is disabled
Standard deviation AppServer time	Time	The standard deviation of the time spent in returning a response from the AppServer to the REST transport
Standard deviation connect time	Time	The standard deviation of the time spent by the REST transport to connect to the AppServer
Standard deviation transport time	Time	The standard deviation of the times spent in making a request to the AppServer via the REST transport and in returning a response from the AppServer by the REST transport
Status requests	Count	The number of requests made for the status of the REST transport
Successful connect requests	Count	The number of requests for which connection objects have been successfully created. These objects are used to run a service on the AppServer.
Successful requests	Count	The number of requests for which the AppServer has sent a successful response
Successful run requests	Count	The number of requests for which an ABL class, function, or procedure has been executed successfully on the AppServer and a response is sent to the REST transport
Type	-	The transport for which the metrics are collected

Note: If the `collectMetrics` property is not set in the `openedge.properties` file, only the **Access Time**, **Last Reset Time**, and **Type** metrics are collected.

You can reset the time and count values of REST transport metrics to zero by clicking **Reset statistics**. The **Last Reset Time** field is updated when the statistics are reset.

Additionally, you can see **PAS REST WebApp Requests Rate** graph that displays the request data over time. Using the options on the top-right corner of the graph, you can select a time period, ranging from 2 hours to 2 weeks, to display the data collected for that period of time, reload the graph, and select different forms of graphical interpretation.

Managing SOAP transports

You can manage your SOAP Web service transport and configuration using the **ABL WebApp: <webapp name>** page.

To manage SOAP transports:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

   Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

4. Click an ABL Web application listed in the **ABL WebApps** section.

   The **ABL WebApp: <webapp name>** page appears. You can view the following information about the SOAP transport:

   Table 16: SOAP Transport Summary

Field	Description
State	State of the SOAP transport (ENABLED or DISABLED)
Access URL	When working with HTTP, use this URL to access the SOAP transport service in the `http://<host_name>:<port>/soap` format
Secure Access URL	When working with HTTPS, use this URL to access the SOAP transport service in the `https://<host_name>:<port>/soap` format
Enabled services	The number of enabled SOAP Web services
Disabled services	The number of disabled SOAP Web services
Total # services	The total number of SOAP Web services deployed

5. Click **Deploy** under the **SOAP Transport** section.

   The **Deploy SOAP application wsm file** page appears.

6. Browse an **Application .wsm file** from your file system, and then click **Deploy** to deploy the Web service on the PAS for OpenEdge instance.

   Note: Click the refresh icon to see the latest status in the list of SOAP Web services deployed on the PAS for OpenEdge instance.

   Click the name of the deployed Web service to view a summary of its metrics, time and count metrics, and graphs that display the request data over time and active requests in the **SOAP <Web service name> metrics** page. For more information about graphs and resetting the metrics, see Viewing SOAP transport metrics on page 70.

   To configure the properties of the deployed Web service, in the **SOAP <Web service name> metrics** page, click **Configuration** under the **Soap Service <Web service name>** section. Set the properties as described in Configuring SOAP transport run time properties on page 67.

7. Select any of the deployed SOAP Web services from the list, and then click **Enable** in the **SOAP Transport** section to enable the Web service artifact on the PAS for OpenEdge instance. Click **Yes** in the confirmation dialog that appears.

8. To disable a SOAP Web service, select the SOAP Web service from the list, and then click **Disable** in the **SOAP Transport** section. Click **Yes** in the confirmation dialog that appears.

9. To undeploy a SOAP Web service from the PAS for OpenEdge instance, select the deployed SOAP Web service from the list and then click **Undeploy**.

   Note: You can only undeploy a disabled Web service. SOAP based Web services are not supported by OpenEdge Management in 12.0.

Configuring SOAP transport run time properties

Every PAS for OpenEdge instance can have a different transport mechanism for all its deployed services based on sessions, timeouts, and other advanced properties. You can configure your SOAP transport run time properties using the **SOAP transport configuration for ABL Application <ABL application name> WebApp <webapp name>** page.

To set SOAP transport run time properties:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

4. Click an ABL Web application listed in the **ABL WebApps** section.

   The **ABL WebApp: <webapp name>** page appears.

5. Click **Run-time Properties** under the **SOAP Transport** section.

   The **SOAP transport configuration for ABL Application <ABL application name> WebApp <webapp name>** page appears.

6. Set the following properties:

   Table 17: SOAP transport run time properties

Parameter name	Default value	Description
Initial sessions	1	The number of network sessions to be created (and shared by all clients) when the connection pool for the service is initialized by the adapter.
Minimum sessions	1	The minimum number of connected sessions that the service attempts to maintain in the service connection pool.
Maximum sessions	0	The maximum number of connected sessions allowed in the service connection pool.
Minimum idle connections	0	The minimum number of idle, or inactive, PAS for OpenEdge connections to maintain.
Idle session timeout	0	The duration, in seconds, between attempts by the adapter to shut down extra network connections to the AppServer, based on client demand.
Request wait timeout	-1	Determines how the Web service handles requests when the service connection pool becomes full. -1 signifies that the request is queued indefinitely until a PAS for OpenEdge session becomes available.
Stale OOABL object timeout	0	The maximum duration, in seconds, that a service object can be idle before it is released.
Connection lifetime	0	The maximum lifetime, in seconds, of PAS for OpenEdge connections in the connection pool for this service.
Service fault level	2	Specifies the degree of detail returned to the client for a service fault.
Wait if busy	Not selected	Determines how to handle client requests to a service that is busy processing a prior request.

7. Click:

- **Save** to save the changes.
- **Cancel** to reject the recent changes.
- **Reset** to apply the last-saved settings.

Configuring SOAP transport properties

You can configure few properties of SOAP transport service of a local PAS for OpenEdge instance even when the OpenEdge manager web application, `oemanager`, is not installed and the instance is not running. You can access these properties using the **SOAP transport configuration for ABL Application <ABL application name> WebApp <webapp name>** page.

Note: To configure the properties of a transport service of a remote PAS for OpenEdge instance, ensure that the `oemanager` is installed and the instance is running.

To configure SOAP transport properties:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

4. Click an ABL Web application listed in the **ABL WebApps** section.

   The **ABL WebApp: <webapp name>** page appears.

5. Click **Configuration** under the **SOAP Transport** section.

   The **SOAP transport configuration for ABL Application <ABL application name> WebApp <webapp name>** page appears.

6. Set the following properties

Table 18: SOAP transport properties

Parameter	Default	Description
Admin Enabled	1	Indicates whether administrator privileges are in effect.
Adapter Enabled	1	Indicates whether WSA (Web Service Adapter) is enabled.
Debug Clients	None	A list of dot-formatted IP addresses for SOAP Web service client machines whose HTTP and SOAP messages you want dumped to the WSA log file.

Parameter	Default	Description
Status Enabled	1	Indicates whether the SOAP Web service is enabled for client access.
Allow Runtime Updates	0	Indicates whether run time property changes are allowed for this application server.
Collect Metrics	1	Collects count and time related information of SOAP Web service over a period of time. You can type the following values: • 0 — Do not collect metrics. • 1 — Collect count metrics only. • 2 — Collect time metrics only. • 3 — Collect both count and time metrics.
WSDL Enabled	1	Controls the WSA's ability to respond to inquiries about WSDL documents for any of the SOAP Web service applications it hosts.
Admin SOAP Action	`urn:services-progress-com:wsa-admin:01`	A string placed in the SOAPAction HTTP header when the AdminServer forwards administration operations to the WSA.
WSA url	`http://host:port/webapp-context/servlet-instance`	The URL for the WSA instance where the Web service is deployed, and forms the root for all URLs which access the Web services it manages.

7. Click:
   - **Save** to save the changes.
   - **Cancel** to reject the recent changes.
   - **Reset** to apply the last-saved settings.

Viewing SOAP transport metrics

You can view SOAP transport metrics by using the **SOAP transport metrics for ABL Application <ABL application name> WebApp <webapp name>** page.

To view SOAP transport metrics:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

   Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

4. Click an ABL Web application listed in the **ABL WebApps** section.

   The **ABL WebApp: <webapp name>** page appears.

5. Click **Metrics** in the **SOAP Transport** section.

   The **SOAP transport metrics for ABL Application <ABL application name> WebApp <webapp name>** page appears. You can view the following metrics:

   Table 19: SOAP Transport Metrics

Metric	Metric type	Description
Access Time	-	The date and time at which the metrics have been retrieved
Active requests	Count	The number of active requests made to the SOAP transport
Average total time	Time	The average of the times spent in making a request to the AppServer via the SOAP transport and in returning a response from the AppServer by the SOAP transport
Http request errors	Count	The number of HTTP requests made that are incorrectly formed
Http requests	Count	The number of requests made that are not related to Web Services Description Language (WSDL) or SOAP
Last Reset Time	-	If you click **Reset**, this field displays the date and time at which you have reset the metrics for the service. When the metrics are reset, the **Last Reset Time** field is updated.
Maximum total time	Time	The maximum value of the times spent in making a request to the AppServer via the SOAP transport and in returning a response from the AppServer by the SOAP transport

Metric	Metric type	Description
Method not allowed errors	Count	The number of errors raised due to HTTP method requests that are not supported by the OpenEdge Web Services Adapter (WSA). **Note:** WSA supports only the HTTP GET and POST methods.
Minimum total time	Time	The maximum value of the times spent in making a request to the AppServer via the SOAP transport and in returning a response from the AppServer by the SOAP transport
Requests	Count	The total number of requests (successful or failed) made to the SOAP transport for access to the AppServer
Service disabled	Count	The number of WSDL or SOAP requests that have been denied because the WSDL or SOAP services are disabled by the administrator
Soap processor errors	Count	The number of SOAP requests that have failed due to an internal SOAP processor-related error
Soap requests	Count	The number of SOAP requests made to the AppServer
Standard deviation total time	Time	The standard deviation of the times spent in making a request to the AppServer via the SOAP transport and in returning a response from the AppServer by the SOAP transport
Start Time	Count	This field displays the number of failed administration and SOAP requests. These failed requests are caused due to an internal SOAP processor-related error.
Status requests	Count	The number of requests made for the status of the SOAP transport
Successful soap requests	Count	The number of requests for which the AppServer has sent a successful response
Type	-	This field displays the service type for which the metrics are collected.
Url not found errors	Count	The number of HTTP requests to a URL path that is not supported by the SOAP transport
Wsdl requests	Count	The number of requests made for retrieving the WSDL document

Note: If the `collectMetrics` property is not set in the `openedge.properties` file, only the **Access time**, **Last reset time**, and **Type** metrics are collected.

You can reset the time and count values of SOAP transport metrics to zero by clicking **Reset statistics**. The **Last Reset Time** field is updated when the statistics are reset.

Additionally, you can see **PAS SOAP WebApp Requests Rate** and **PAS SOAP WebApp Active Requests** graphs that display the request data over time and active requests, respectively. Using the options on the top-right corner of the graph, you can select a time period, ranging from 2 hours to 2 weeks, to display the data collected for that period of time, reload the graph, and select different forms of graphical interpretation.

Managing APSV transports

You can manage and configure your APSV transport services using the **ABL WebApp: <webapp name>** page.

To manage the APSV services deployed on PAS for OpenEdge:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

   Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

4. Click an ABL Web application listed in the **ABL WebApps** section.

   The **ABL WebApp: <webapp name>** page appears. You can view the following information about the APSV transport:

Table 20: APSV Transport Summary

Field	Description
State	State of the APSV transport (ENABLED or DISABLED)
Access URL	When working with HTTP, use this URL to access the APSV transport service in the `http://<host_name>:<port>/apsv` format
Secure Access URL	When working with HTTPS, use this URL to access the APSV transport service in the`https://<host_name>:<port>/apsv` format

5. Click **Enable** in the **APSV Transport** section to enable the APSV transport. Click **Yes** in the confirmation dialog that appears.

   The enabled state is displayed in the **APSV Transport Summary** section.

6. To disable the APSV transport, click **Disable** in the **APSV Transport** section. Click **Yes** in the confirmation dialog that appears.

   The disabled state is displayed in the **APSV Transport Summary** section.

Configuring APSV transport properties

Each PAS for OpenEdge instance can have a different transport mechanism for all its deployed services based on sessions, timeouts, and other advanced properties. You can configure your APSV transport configurations using the **APSV transport configuration for ABL Application <ABL application name> WebApp <webapp name>** page.

To set APSV transport properties:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

   You can configure the properties of a transport service of a local PAS for OpenEdge instance even when the OpenEdge manager web application, `oemanager`, is not installed and the instance is not running. However, to configure the properties of a transport service of a remote PAS for OpenEdge instance, ensure that the `oemanager` is installed and the instance is running.

4. Click an ABL Web application listed in the **ABL WebApps** section.

   The **ABL WebApp: <webapp name>** page appears.

5. Click **Configuration** under the **APSV Transport** section.

   The **APSV transport configuration for ABL Application <ABL application name> WebApp <webapp name>** page appears.

6. Set the following properties in the **APSV Transport Properties** section:

   Table 21: APSV Transport properties

Parameter	Default	Description
Status enabled	1	Returns the status of the server or APSV transport in JSON format.
Enable request chunking	1	Enables chunking for the ABL client. This results in a better performance when transferring large amounts of data.
Allow runtime updates	0	If the value is 0, it does not allow certain properties to be updated dynamically.

Parameter	Default	Description
Collect metrics	1	Collects the metrics used by oemanager.
Adapter enabled	1	Enables the APSV transport from servicing requests.
Use HTTP sessions	1	Enables HTTP sessions for the ABL client. An HTTP session is created when the ABL client connects.

7. Click:
   - **Save** to save the changes.
   - **Cancel** to reject the recent changes.
   - **Reset** to apply the last-saved settings.

Viewing APSV transport metrics

You can view APSV transport metrics by using the **APSV transport metrics for ABL Application <ABL application name> WebApp <webapp name>** page.

To view APSV transport metrics:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.
2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.
3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

   Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.
4. Click an ABL Web application listed in the **ABL WebApps** section.

   The **ABL WebApp: <webapp name>** page appears.
5. Click **Metrics** in the **APSV Transport** section.

   The **APSV transport metrics for ABL Application <ABL application name> WebApp <webapp name>** page appears. You can view the following metrics:

   Table 22: APSV Transport Metrics

Metric	Metric type	Description
Access Time	-	The date and time at which the metrics have been retrieved.

Metric	Metric type	Description
Average connect time	Time	The average time spent in using the APSV transport to connect to the AppServer
Average disconnect time	Time	The average time spent in using the APSV transport to disconnect from the AppServer
Average session time	Time	The average time spent in retrieving a session to invoke an ABL class, function, or procedure
Connect errors	Count	The number of connect requests that have led to an error
Connect requests	Count	The number of connect requests that have been made between the APSV transport and AppServer
Disconnect requests	Count	The number of disconnect requests that have been made between the APSV transport and AppServer
Forbidden errors	Count	The number of requests made during which the APSV transport is disabled
Last reset time	-	If you click **Reset**, this field displays the date and time at which you have reset the metrics for the service. When the metrics are reset, the **Last Reset Time** field is updated.
Maximum connect time	Time	The maximum time spent in using the APSV transport to connect to the AppServer
Maximum disconnect time	Time	The maximum time spent in using the APSV transport to disconnect from the AppServer
Maximum session time	Time	The maximum time spent in retrieving a session to invoke an ABL class, function, or procedure
Minimum connect time	Time	The minimum time spent in using the APSV transport to connect to the AppServer
Minimum disconnect time	Time	The minimum time spent in using the APSV transport to disconnect from the AppServer
Minimum session time	Time	The minimum time spent in retrieving a session to invoke an ABL class, function, or procedure
Session errors	Count	The number of errors raised in retrieving a session to invoke an ABL class, function, or procedure
Session requests	Count	The number of requests made in retrieving a session to invoke an ABL class, function, or procedure
Standard deviation connect time	Time	The standard deviation of the time spent in using the APSV transport to connect to the AppServer

Metric	Metric type	Description
Standard deviation disconnect time	Time	The standard deviation of the time spent in by the APSV transport to disconnect from the AppServer.
Standard deviation session time	Time	The standard deviation of the time spent in retrieving a session to invoke an ABL class, function, or procedure
Status requests	Count	The number of requests made for the status of the APSV transport
Type	-	The transport for which the metrics are collected

Note: If the `collectMetrics` property is not set in the `openedge.properties` file, only the `type`, `accessTime`, and `lastResetTime` metrics are collected.

You can reset the time and count values of APSV transport metrics to zero by clicking **Reset statistics**. The **Last Reset Time** field is updated when the statistics are reset.

Additionally, you can see **PAS APSV WebApp Requests Rate** graph that displays the request data over time. Using the options on the top-right corner of the graph, you can select a time period, ranging from 2 hours to 2 weeks, to display the data collected for that period of time, reload the graph, and select different forms of graphical interpretation.

Managing WEB transports

PAS for OpenEdge supports deployment of WebSpeed applications and you can manage your WEB transport services using the **ABL WebApp: <webapp name>** page.

To manage the WEB services deployed on PAS for OpenEdge:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

   Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

4. Click an ABL Web application listed in the **ABL WebApps** section.

   The **ABL WebApp: <webapp name>** page appears. You can view the following information about the WEB transport:

Table 23: WEB Transport Summary

Field	Description
State	State of the WEB transport (ENABLED or DISABLED)
Access URL	When working with HTTP, use this URL to access the WEB transport service in the `http://<host_name>:<port>/apsv` format
Secure Access URL	When working with HTTPS, use this URL to access the WEB transport service in the`https://<host_name>:<port>/apsv` format

5. Click **Enable** in the **WEB Transport** section to enable the WEB transport. Click **Yes** in the confirmation dialog that appears.

   The enabled state is displayed in the **WEB Transport Summary** section.

6. To disable the WEB transport service, click **Disable** in the **WEB Transport** section . Click **Yes** in the confirmation dialog that appears.

   The disabled state is displayed in the **WEB Transport Summary** section.

Configuring WEB transport properties

Each PAS for OpenEdge instance can have a different transport mechanism for all its deployed services based on sessions, timeouts, and other advanced properties. You can configure your WEB transport properties using the **WEB transport configuration for ABL application <ABL application name> WebApp <webapp name>** page.

To configure WEB transport properties:

1. Click **Resources** > **Go to Resources** in the OpenEdge management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server:<instance name>** page appears.

3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

   You can configure the properties of a transport service of a local PAS for OpenEdge instance even when the OpenEdge manager web application, `oemanager`, is not installed and the instance is not running. However, to configure the properties of a transport service of a remote PAS for OpenEdge instance, ensure that the `oemanager` is installed and the instance is running.

4. Click an ABL Web application listed in the **ABL WebApps** section.

   The **ABL WebApp: <webapp name>** page appears.

5. Click **Configuration** under the **WEB Transport** section.

   The **WEB transport configuration for ABL Application <ABL application name> WebApp <webapp name>** page appears.

6. Set the following properties in the **WEB Transport Properties** section:

Table 24: WEB Transport properties

Parameter	Default	Description
Status enabled	1	Returns the status of the server or WEB transport in JSON format.
Adapter enabled	1	Enables the WEB transport from servicing requests.
Server debug	1	Allows debugging via URL: `(?debug=on)`
Collect metrics	1	Collects the metrics used by oemanager.
Allow runtime updates	0	If the value is 0, it does not allow certain properties to be updated dynamically.
Default cookie domain		The domain to send cookies. By default, this is blank which restricts the web browser to set the cookies only to the current host. You can specify the domain name for which the cookies need to be available to multiple hosts.
Default cookie path		A server relative URL for all cookies. The
Default handler	`OpenEdge.Web.CompatibilityHandler`	The handler to which all requests are routed. These requests are not handled by a mapped handler. The Compatibility handler supports classic WebSpeed applications and can also be mapped to a URL.

7. Click:
   - **Save** to save the changes.
   - **Cancel** to reject the recent changes.
   - **Reset** to apply the last-saved settings.

Viewing WEB transport metrics

You can view WEB transport metrics by using the **WEB transport metrics for ABL Application <ABL application name> WebApp <webapp name>** page.

To view WEB transport metrics:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.
2. Filter or search for, and select the required PAS for OpenEdge instance.

For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

The **Progress Application Server: <instance name>** page appears.

3. Click an ABl application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

---

4. Click an ABL Web application listed in the **ABL WebApps** section.

   The **ABL WebApp: <webapp name>** page appears.

5. Click **Metrics** in the **WEB Transport** section.

   The **WEB transport metrics for ABL Application <ABL application name> WebApp <webapp name>** page appears. You can view the following properties:

Table 25: WEB Transport Metric properties

Property	Description
Access Time	The date and time at which the metrics have been retrieved
Last Reset Time	The date and time at which you have last reset the metrics for the service.
Requests	A group of various requests made to the service
Errors	A group of various errors encountered from the time the service has started or last reset.

Note: If the `collectMetrics` property is not set in the `openedge.properties` file, only the **Access Time**, **Last Reset Time**, and **Type** metrics are collected.

You can reset the time and count values of WEB transport metrics to zero by clicking **Reset statistics**. The **Last Reset Time** field is updated when the statistics are reset.

Additionally, you can see **PAS REST WebApp Requests Rate** and **PAS WEB WebApp Error Rate** graphs to display the request data and error requests over time, respectively. Using the options on the top-right corner of the graph, you can select a time period, ranging from 2 hours to 2 weeks, to display the data collected for that period of time, reload the graph, and select different forms of graphical interpretation.

Managing WEB handlers

You can add, edit or delete a WebHandler using the **ABL WebApp: <webapp name>** page.

To manage the WebHandlers:

1. Click **Resources** > **Go to Resources** in the OpenEdge Management console menu.

   All resources managed by your console appear in the grid frame.

2. Filter or search for, and select the required PAS for OpenEdge instance.

   For example, type the default PAS for OpenEdge instance name, `oepas1`, in the **Filter** field or select **Progress Application Server** as the selection in the **Type** drop-down menu and select the PAS instance.

   The **Progress Application Server: <instance name>** page appears.

3. Click an ABL application listed in the **ABL Applications** section.

   The **ABL Application: <ABL application name>** page appears.

   Note: The OpenEdge manager web application, `oemanager`, must be installed on PAS for OpenEdge for you to manage OpenEdge ABL applications and transport services while the PAS for OpenEdge instance is running.

   ---

4. Click an ABL Web application listed in the **ABL WebApps** section.

   The **ABL WebApp: <webapp name>** page appears.

5. Click **Handlers** in **WEB Transport** section.

   The **Web handlers: <webapp name>** page appears.

6. To add a handler:

   a) Click **Add handler** to add a WebHandler.

   b) Enter the properties of the handler you want to add in the **Handler class** and **Handler URL** sections.

   c) Click **Update** to update the grid list.

   d) Click **Save** to save the handler properties to `openedge.properties` file.

7. To edit a handler listed in the grid:

   a) Click **Edit** against the handler you want to make changes.

   b) Click **Update** and click **Save**.

      To cancel the changes you made, click **Cancel**.

8. To delete a handler listed in the grid:

   a) Click **Delete** against the handler you want to delete.

   b) Click **Yes** to confirm in the **Delete handler** confirmation dialog box that appears.

      To cancel deletion, click **No**

   c) Click **Save** to update the `openedge.properties` file.

9. Optionally, you can drag and drop the handlers to change the order of the handlers in the list. Click **Save** to update the `openedge.properties` file

3

TCMAN Reference

TCMAN is a command-line utility for managing and administering the Progress Application Server. TCMAN extends the basic Tomcat scripts for starting, stopping, and managing server instances.

This *TCMAN Reference* contains usage information for the `tcman` command as well as syntax information on all of the TCMAN actions.

For details, see the following topics:

- Using TCMAN
- Manager actions
- Server actions
- General actions

Using TCMAN

You use the TCMAN utility by running the `tcman` command, with specified actions and options, from either the core PAS (`$CATALINA_HOME/bin`) or from an instance (`$CATALINA_BASE/bin`). You can run `tcman` from a command line, or in a script. Both Windows (`tcman.bat`) and UNIX (`tcman.sh`) versions of TCMAN are implemented in the Progress Application Server.

See also

The tcman command on page 84
Extending TCMAN on page 86

Manager actions on page 88
Server actions on page 100
General actions on page 123

The tcman command

Purpose

TCMAN is a command-line utility for managing and administering PAS. On UNIX systems, you run the `tcman.sh` script followed by appropriate TCMAN actions and options. On Windows systems, you run the `tcman.bat` batch file, which is identical syntactically and functionally with `tcman.sh`.

Note: For the sake of brevity, all the syntax statements and examples in this reference show the `tcman.sh` script.

Syntax

```
{ $CATALINA_HOME | $CATALINA_BASE } /bin/tcman.sh action  [ general_options ]
[ action_options ]
```

Parameters

`$CATALINA_HOME|$CATALINA_BASE`

Specify whether to run TCMAN from the root directory of the installed PAS (`$CATALINA_HOME`) or from the root directory of an instance (`$CATALINA_BASE`). The context of where you run TCMAN (whether from the `/bin` directory of the parent, or the `/bin` directory of an instance) affects which server the utility acts on.

Note: TCMAN automatically determines the value of `CATALINA_BASE` from the directory where you start it. When you run it from the `/bin` directory of an instance, the value of `CATALINA_BASE` is the root directory of the instance. If you run it from the `/bin` directory of the installed Progress Application Server, the value of `CATALINA_BASE` is the root directory of the installed server (which is the same value as `CATALINA_HOME`).

`action`

Specify which TCMAN action to invoke.

`general_options`

Specify one or more of the TCMAN common options that can apply to most actions. Note that one or more of the general options may be required by a specific action. For example, the `list` action requires `-u` in order to pass a user name and password.

The output of `tcman.sh help` *`action`* includes a list of general options that are applicable to a particular action.

The following table is a list of the common options:

Table 26: TCMAN general options

Common options	Function
-u *user_name*:*password*	Pass a valid user name and a password for HTTP Basic access authentication.
-v	Display verbose output.
-M *URL*	Override the default manager that manages Web applications by specifying the URL of an alternative manager. *URL* is expressed in the following format: {http|https}://*host*:*port*/*manager_application*
-B	Override default CATALINA_BASE environment settings.
-n	Debug the TCMAN action but do not execute changes.
-I *instance_name*	Run TCMAN from the /bin directory of the specified instance.

action_options

Specify an option that applies to the selected *action*. These options are explained in the topics that describe each action.

Example

Run the `help` action from the core server (`/psc/pashome`) to display a list of available TCMAN actions:

```
/psc/pashome/bin/tcman.sh help
usage:  tcman action [options...]
  manager actions:
      list         list deployed applications
      info         list server info
      deploy       deploy application
      undeploy     undeploy application
      reload       reload application
      status       show server status
      leaks        show server memory leaks
      enable       start web application running
      disable      stop running web application
      resources    list server global resources
      sessions     list a web application's sessions
  server actions:
      create       create a new server instance
      delete       delete server instance
      config       dump CATALINA_BASE configuration
      clean        clean/archive log files
      instances    list tracked server instances
      register     manually register an instance
      unregister   manually unregister an instance
      start        start this server
      stop         stop this server
      version      show the server version information
      test         test the server's configuration
  general actions:
      env          show tcman execution environment
      help         show this information
```

See also

Manager actions on page 88
Server actions on page 100
Extending TCMAN on page 86
General actions on page 123

Extending TCMAN

You can extend TCMAN by adding your own actions to the built-in actions of the TCMAN utility. You implement actions as UNIX shell scripts, Windows batch files, or Windows PowerShell scripts. Customizing TCMAN with your own actions allows you to unify all of your administrative tasks under a single utility.

File name format

Name your TCMAN action scripts using the following convention:

```
action_nameCmd{.sh | .bat | .ps1}
```

where file name extensions `.sh`, `.bat`, and `.ps1` are used for UNIX scripts, Windows batch files, and PowerShell scripts respectively.

UNIX action scripts

UNIX action scripts must be executable. That is, the file must have execution permissions for the current user. Each script must exit with a zero (0) status when successful, or a positive non-zero value to indicate an error.

When the action script is in the core server's bin directory (`$CATALINA_HOME/bin`), it is available to administrators of any instance. When the action script is in an instance's bin directory (`$CATALINA_BASE/bin`), it is only available to administrators of that instance.

The search order is:

1. `$CATALINA_BASE/bin`
2. `$CATALINA_HOME/bin`

Windows batch and PowerShell action scripts

On Windows, action scripts can be either DOS batch scripts (`.bat`) or PowerShell scripts (`.ps1`). Batch scripts should return zero (0) for success, or a positive non-zero integer to indicate an error level. Powershell scripts should return `\$true` for success or `\$false` to indicate an error.

When the action script is in the core server's bin directory (`$CATALINA_HOME/bin`), it is available to administrators of any instance. When the action script is in an instance's bin directory (`$CATALINA_BASE/bin`), it is only available to administrators of that instance.

If you have both batch and PowerShell scripts, TCMAN gives precedence to batch files. The search order is:

1. `$CATALINA_BASE/bin/`*`action_name`*`Cmd.bat`
2. `$CATALINA_BASE/bin/`*`action_name`*`Cmd.ps1`
3. `$CATALINA_HOME/bin/`*`action_name`*`Cmd.bat`
4. `$CATALINA_HOME/bin/`*`action_name`*`Cmd.ps1`

Command-line Syntax

You invoke an action script by specifying the *`action-name`* segment of a script's file name in a TCMAN command line.

```
tcman.sh action-name [general_options] [ action-name_options]
```

Note: You do not include the `Cmd.`*`extension`* portion of the file name on the TCMAN command line.

TCMAN first looks for action scripts in the instance (`$CATALINA_BASE/bin`) and then in the core server (`$CATALINA_HOME/bin`). When it finds the action script, TCMAN passes all command line options and parameters during execution.

Example

The following command line runs a script file named `viewCmd.sh` and passes the `-v` option and the `logs/catalina.out` argument.

```
tcman.sh view -v  logs/catalina.out
```

Notes

- When TCMAN does not recognize *action-name* as one of the built-in actions, it looks for an action script . If the specified action is not a built-in action or external action script it will exit with an error.
- When the external action script exits, TCMAN uses its return status to either issue an error or to exit with success. Therefore, extension action scripts should always return a specific value according to the platform type.
- Before executing an external action script, TCMAN exports these environment variables for consumption by the action script file:
  - JAVA_HOME : the root directory of the JAVA JDK (optional)
  - JRE_HOME : the root directory of the JAVA JRE (optional)
  - CATALINA_HOME : the root directory of where PAS was installed
  - CATALINA_BASE : the root directory of the PAS instance being managed (can be the same as CATALINA_HOME)
  - CATALINA_TMPDIR : the temporary file directory for temp files
  - CATALINA_PID : the file where the PAS process-id is stored
  - tcman_osshell : The name of the OS and command line shell TCMAN is running in
  - tcman_instancebase : the name of the new instance base directory when executing a create action
  - tcman_httpport : the HTTP network port command line option (`-p`)
  - tcman_httpsport : the HTTPS network port command line option (`-P`)
  - tcman_cfghttpport : the current instance's configured HTTP network port
  - tcman_cfghttpsport : the current instance's configuredHTTPS network port
  - tcman_cfgshutport : the current instance's shutdown network port (can be `-1` if there is no network shutdown port)
  - tcman_alias : the instance's alias name
  - tcman_type : the type of instance (home, service (on Windows), instance)
  - tcman_secmodel : the security model to apply (`default`, `production`, or `development`)
  - tcman_umask : the UNIX umask to use when creating new directories and files
  - tcman_verbose : the state of the TCMAN `-v` command line option
  - tcman_debug : The state of the TCMAN `-g` command line option

Manager actions

This section details the actions available for deploying, running, and monitoring web applications on a server instance.

See also

The tcman command on page 84

List deployed applications (list)

Purpose

Display all the web applications that are deployed on an instance.

Note: This command may be used whether the instance is online or offline. However, the output differs. When used offline, TCMAN simply shows a list of deployed application directories in the instance's web applications directory. When used online, it provides additional run-time details about the deployed web applications.

To use this action, the Tomcat manager (`manager.war`) must be deployed on the instance if the instance is online. You can deploy `manager.war` from `$CATALINA_HOME/extras`.

Syntax

```
tcman.sh list [general_options] [-u user_id:password]
```

Parameters

general_options

Specify one or more of the options that can be used with any TCMAN action. Run `tcman.sh help list` to see which general options are appropriate.

`-u` *user_id*`:`*password*

Specify a valid user name and password for HTTP Basic access authentication. (The default is `-u tomcat:tomcat`.)

Note: This option is required if the server is online. It is not required if the server is offline.

Example

Show the Web applications deployed to `acme1` when the instance is online:

```
/psc/acme1/bin/tcman.sh list -u tomcat:tomcat
OK - Listed applications for virtual host localhost
/:running:0:ROOT
/manager:running:4:manager
/oemanager:running:0:oemanager
/oeadapters:running:0:oeabl
```

Show the Web applications deployed to `acme1` when the instance is offline:

```
 /psc/acme1/bin/tcman.sh list
OK - Listing directories for /psc/acme1/webapps
/manager:stopped:0:manager
/oeadapters:stopped:0:oeabl
/oemanager:stopped:0:oemanager
/:stopped:0:ROOT
```

See also

Display Web application HTTP sessions (sessions) on page 100
Deploy a Web application (deploy) on page 91
Undeploy a Web application (undeploy) on page 93
Start a Web application (enable) on page 97
Stop a Web application (disable) on page 98
The tcman command on page 84

Display OS and server information (info)

Purpose

Display server and OS information for a running instance.

To use this action, the Tomcat manager (`manager.war`) must be deployed on the instance and the instance must be running. You can deploy `manager.war` from `$CATALINA_HOME/extras`.

Use the `test` action to show configuration information about a server that is not running.

Syntax

```
tcman.sh info [general_options]  -u user_name:password
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run `tcman.sh help info` to see which general options are appropriate.

`-u` *user_name*`:`*password*

Pass a valid user name and a password for HTTP Basic access authentication. (The default is `-u tomcat:tomcat`.)

Example

Display the OS and server information for the running instance named `acme1`:

```
$: /psc/pashome/tcman.sh info -I acme1 -u tomcat:tomcat
OK - Server info
Tomcat Version: Apache Tomcat/7.0.42
OS Name: Linux
OS Version: 2.6.18-164.el5
OS Architecture: amd64
JVM Version: 1.7.0_02-b13
JVM Vendor: Oracle Corporation
```

See also

Display server, OS, and runtime version information (version) on page 118
Display detailed server status (status) on page 95
Test a server configuration (test) on page 119
The tcman command on page 84
Display or modify the server features of an instance (feature) on page 106
The tcman command on page 84

Deploy a Web application (deploy)

Purpose

Deploy an ABL web application (`.war` file) to a PAS for OpenEdge instance, online (when the server is running) or offline (when the server is not running). TCMAN copies the web application to the server's web application directory.

Syntax

```
tcman.sh deploy [general_options] -l [-u user_id:password] [-a app_name]
war_file_path [abl_app_name]
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run `tcman.sh help deploy` to see which general options are appropriate.

`-l`

Use this switch to deploy an ABL web application online without having to restart the instance. The switch loads the context necessary for the successful online deployment of a web application.

Note: The Tomcat manager (oemanager.war) must already be deployed for the `-l` switch to work. In the PAS for OpenEdge development server, the Tomcat manager is by default deployed in the default instance, `oepas1`. For all new instances in both the development and the production servers, you have to deploy the Tomcat manager first before you can use this switch. Also note that the `-l` switch can be used only in conjunction with the `-u` parameter.

`-u` *`user_id`*`:`*`password`*

Specify a valid user name and password for HTTP Basic access authentication.

Note: This parameter is required if the server is online. It is not required if the server is offline.

`-a` *`app _name`*

Specify a name for the web application. If you do not use this option, the application name will be the same as the `.war` file name.

`war_file_path`

Specify the location of the web application `.war` file that you want to deploy.

`abl_app_name`

It is the name of the ABL application under which you want to deploy your ABL web application.

An ABL application is a logical grouping of ABL web applications. It consists of a session manager, one or more multi-session agents, and one or more deployed ABL web applications.

An instance can contain more than one ABL application. You can deploy an ABL web application to either the default ABL application, which is named after the instance, or to a new ABL application. If you want to deploy an ABL web application under a new ABL application, then specify a name for the ABL application here.

Example

Deploy and rename `oeabl.war` (a web application that implements OpenEdge adapters) to the `acme1` instance of the core `pashome` server online:

```
/psc/acme1/bin/tcman.sh deploy -l -u tomcat:tomcat -a oeadapters
/psc/pashome/extras/oeabl.war
OK - deployed /psc/pashome/extras/oeabl.war to local directory /psc/acme1/webapps
```

Note: The `$CATALINA_HOME/extras` directory (`/psc/pashome/extras` in the example above) also contains number of instance management applications, including `host-manager.war`, `manager.war`, and `oemanager.war`.

See also

List deployed applications (list) on page 89
Undeploy a Web application (undeploy) on page 93
Reload a Web application (reload) on page 94
Start a Web application (enable) on page 97

Stop a Web application (disable) on page 98
The tcman command on page 84

Undeploy a Web application (undeploy)

Purpose

Remove a Web application from running (online) or stopped (offline) instances. If the instance's autodeploy option is off, you must stop and restart a running server to complete removal. Note that the autodeploy option is set in the `.../conf/appserver.properties` file and is off by default.

Syntax

```
tcman.sh undeploy [general_options] [-u user_id:password]  app _name
```

Parameters

general_options

Specify one or more of the options that can be used with any TCMAN action. Run `tcman.sh help undeploy` to see which general options are appropriate.

`-u` *user_id*`:`*password*

Specify a valid user name and password for HTTP Basic access authentication. (The default is `-u tomcat:tomcat`.) This option is required if you are accessing an online instance.

app_name

Specify the name of the web application to remove.

Example

Remove the oemanager application from the `acme1` instance:

```
/psc/acme1/bin/tcman.sh undeploy -u tomcat:tomcat oemanager
              OK - Undeployed application at context path /oemanager
```

See also

Deploy a Web application (deploy) on page 91
List deployed applications (list) on page 89
Reload a Web application (reload) on page 94
The tcman command on page 84

Reload a Web application (reload)

Purpose

Restart a deployed, running Web application so that the application can pick up changes to its classes or libraries.

To use this action, the Tomcat manager (`manager.war`) must be deployed on the instance and the instance must be running. You can deploy `manager.war` from `$CATALINA_HOME/extras`.

Note: The reload action does not reload the web application's `web.xml` file. To begin using changes to `web.xml`, you must stop and restart the web application.

Syntax

```
tcman.sh reload [general_options] -u user_id:password app_name
```

Parameters

general_options

Specify one or more of the options that can be used with any TCMAN action. Run `tcman.sh help reload` to see which general options are appropriate.

`-u` *user_id*`:`*password*

Specify a valid user name and password for HTTP Basic access authentication. (The default is `-u tomcat:tomcat`.)

Note: This option is required if the server in online. It is not required if the server is offline.

app_name

Specify the name of the web application to restart.

Example

Reload the `oemanager` web application running on the `acme1` instance:

```
 /psc/acme1/bin tcman.sh reload  -u tomcat:tomcat oemanager
OK - Reloaded application at context path /oemanager
```

See also

Start a Web application (enable) on page 97
Stop a Web application (disable) on page 98
List deployed applications (list) on page 89
The tcman command on page 84

Display detailed server status (status)

Purpose

List information from the core server's memory, including web application statistics. Information includes memory pool usage, connector thread status, and connector status. Output is in XML format. (Note that redirecting the output to an XML viewer makes it more readable.)

To use this action, the Tomcat manager (`manager.war`) must be deployed on the instance and the instance must be running. You can deploy `manager.war` from `$CATALINA_HOME/extras`.

Syntax

```
tcman.sh status [general_options]  -u user_name:password [-f]
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run `tcman.sh help status` to see which general options are appropriate.

`-u` *user_name*`:`*password*

Pass a valid user name and a password for HTTP Basic access authentication. (The default is `-u tomcat:tomcat`.)

`-f`

Return full status information.

Example

Display core server's memory and web application statistics and use `xmllint` to format for readability:

```
$: tcman.sh status -u tomcat:tomcat | xmllint --format -
<?xml version="1.0" encoding="utf-8"?>
<?xml-stylesheet type="text/xsl" href="/manager/xform.xsl" ?>
<status>
  <jvm>
    <memory free="453196832" total="520028160" max="1051394048"/>
    <memorypool name="PS Eden Space" type="Heap memory" usageInit="50331648"
usageCommitted="48758784" usageMax="55967744" usageUsed="1525560"/>
    <memorypool name="PS Old Gen" type="Heap memory" usageInit="469762048"
usageCommitted="469762048" usageMax="1006632960" usageUsed="63861584"/>
    <memorypool name="PS Survivor Space" type="Heap memory" usageInit="8388608"
usageCommitted="1507328" usageMax="1507328" usageUsed="1444184"/>
    <memorypool name="Code Cache" type="Non-heap memory" usageInit="2555904"
usageCommitted="3407872" usageMax="50331648" usageUsed="3303104"/>
    <memorypool name="PS Perm Gen" type="Non-heap memory" usageInit="67108864"
usageCommitted="67108864" usageMax="134217728" usageUsed="47406400"/>
  </jvm>
  <connector name="&amp;quot;http-bio-8601&amp;quot;">
    <threadInfo maxThreads="150" currentThreadCount="0" currentThreadsBusy="0"/>
    <requestInfo maxTime="0" processingTime="0" requestCount="0" errorCount="0"
bytesReceived="0" bytesSent="0"/>
    <workers/>
  </connector>
  <connector name="&amp;quot;http-bio-8501&amp;quot;">
    <threadInfo maxThreads="300" currentThreadCount="10" currentThreadsBusy="1"/>
    <requestInfo maxTime="2008" processingTime="2116" requestCount="10" errorCount="0"
 bytesReceived="0" bytesSent="5838"/>
    <workers>
      <worker stage="S" requestProcessingTime="2" requestBytesSent="0"
requestBytesReceived="0" remoteAddr="127.0.0.1" virtualHost="localhost" method="GET"
currentUri="/manager/status" currentQueryString="XML=true" protocol="HTTP/1.1"/>
    </workers>
  </connector>
</status>
```

See also

Display OS and server information (info) on page 90
Display memory leaks (leaks) on page 96
The tcman command on page 84

Display memory leaks (leaks)

Purpose

List Web applications with potential memory leaks.

To use this action, the Tomcat manager (`manager.war`) must be deployed on the instance and the instance must be running. You can deploy `manager.war` from `$CATALINA_HOME/extras`.

Syntax

```
tcman.sh leaks [general_options]  -u user_name:password
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run `tcman.sh help leaks` to see which general options are appropriate.

`-u` *user_name*`:`*password*

Pass a valid user name and a password for HTTP Basic access authentication. (The default is `-u tomcat:tomcat`.)

Example

Display memory leaks for web applications deployed on the `acme1` server instance:

```
/psc/acme1/bin/tcman.sh leaks -u tomcat:tomcat
OK - Found potential memory leaks in the following applications:
/warehouse
```

See also

List deployed applications (list) on page 89
The tcman command on page 84

Start a Web application (enable)

Purpose

Start a web application that is deployed but not running.

To use this action, the Tomcat manager (`manager.war`) must be deployed on the instance and the instance must be running. You can deploy `manager.war` from `$CATALINA_HOME/extras`.

Syntax

```
tcman.sh enable [general_options] -u user_id:password app_name
```

Parameters

general_options

Specify one or more of the options that can be used with any TCMAN action. Run `tcman.sh help start` to see which general options are appropriate.

`-u` *user_id*`:`*password*

Specify a valid user name and password for HTTP Basic access authentication. (The default is `-u tomcat:tomcat`.)

app_name

Specify the name of the web application to start.

Note: To start the ROOT web application, you can specify / or ROOT.

Example

Start the oeabl application deployed on the acme1 instance:

```
tcman.sh enable -u tomcat:tomcat oeabl
OK - Started application at context path /oeabl
```

See also

Stop a Web application (disable) on page 98
The tcman command on page 84
The tcman command on page 84

Stop a Web application (disable)

Purpose

Stop a running Web application.

To use this action, the Tomcat manager (manager.war) must be deployed on the instance and the instance must be running. You can deploy manager.war from $CATALINA_HOME/extras.

Syntax

```
tcman.sh disable [general_options] [-u user_id:password] app_name
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run tcman.sh help disable to see which general options are appropriate.

-u *user_id*:*password*

Specify a valid user name and password for HTTP Basic access authentication. (The default is -u tomcat:tomcat.)

app_name

Specify the name of the web application to disable.

Note: To disable the ROOT web application, you can specify / or ROOT.

Example title

Disable the oeabl application running on the acme1 instance:

```
/psc/acme1/bin/tcman.sh disable -u tomcat:tomcat oeabl
OK - Stopped application at context path /oeabl
```

See also

Start a Web application (enable) on page 97
The tcman command on page 84

Display global server resources (resources)

Purpose

List the global resources used by the core server.

To use this action, the Tomcat manager (manager.war) must be deployed on the instance and the instance must be running. You can deploy manager.war from $CATALINA_HOME/extras.

Syntax

```
tcman.sh resources [general_options]  -u user_name:password
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run tcman.sh help resources to see which general options are appropriate.

-u *user_name*:*password*

Pass a valid user name and a password for HTTP Basic access authentication.
(The default is -u tomcat:tomcat.)

Example

Display global resources for the running instance, acme1:

```
$: /psc/acme1/bin/tcman.sh resources -u tomcat:tomcat
OK - Listed global resources of all types
ServiceRegistry/ServiceRegistryFactory:com.progress.appserv.services.naming.ServiceRegistry
UserDatabase:org.apache.catalina.users.MemoryUserDatabase
```

Display Web application HTTP sessions (sessions)

Purpose

Display how many sessions are active for the specified Web application, categorized by their duration.

To use this action, the Tomcat manager (`manager.war`) must be deployed on the instance and the instance must be running. You can deploy `manager.war` from `$CATALINA_HOME/extras`.

Syntax

```
tcman.sh sessions [general_options] -u user_id:password app_name
```

Parameters

general_options

Specify one or more of the options that can be used with any TCMAN action.

`-u` *user_id*`:`*password*

Specify a valid user name and password for HTTP Basic access authentication. (The default is `-u tomcat:tomcat`.)

app_name

Specify the name of the web application to analyze for session information.

Example

Show the active sessions for the `manager` application deployed on the `acme1` instance:

```
/psc/acme1/bin/tcman.sh sessions -u tomcat:tomcat manager
OK - Session information for application at context path /manager
Default maximum session inactive interval 30 minutes
<1 minutes: 1 sessions
8 - <9 minutes: 2 sessions
9 - <10 minutes: 1 sessions
```

See also

List deployed applications (list) on page 89

Server actions

This section details the actions available for creating and monitoring server instances.

See also

The tcman command on page 84

Create an instance (create)

Purpose

Create a new instance of the core PAS server by running this action from `/bin` directory of the core server (`$CATALINA_HOME/bin/tcman.sh create`).

Syntax

```
tcman.sh create [general_options] [-f] [-p port_num] [-P port_num]
   [-s port_num] [-j port_num] [-m uid:pwd] [-W pathname] [-N instance_name]
   [-U user_id ] [-G group_id] [-Z {prod | dev}] base_path
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run `tcman.sh help create` to see which general options are appropriate.

`-f`

Copy all deployed web application archives (`.war` files) from `$CATALINA_HOME` to the new instance.

`-p` *port_num*

Specify the TCP port that listens for HTTP messages. The default is 8080.

`-P` *port_num*

Specify the TCP port that listens for HTTPS messages. The default is 8443.

`-s` *port_num*

Specify the TCP port to use to stop an instance. On Windows systems, you must specify a shutdown port. On UNIX, shutdown ports are optional.

`-j` *port_num*

Specify the TCP port that listens for AJP13 messages (an Apache protocol for handling requests from a web server to an application server). The default is 8009.

`-m` *`uid:pwd`*

Specify a user name and password that will be required to access Tomcat container-level security, which includes the `manager` and `oemanager` web applications. Replaces the defaults (`tomcat:tomcat`) in `/conf/tomcat-users.xml`.

`-W` *`pathname`*

Specify the directory where web applications will be deployed. The default is `$CATALINA_BASE/webapps`.

`-N` *`instance_alias`*

Specify an alias for the instance. If you do not specify an alias, the instance name will be the name of the directory where the instance is created.

Note:

All instances are automatically registered for tracking when they are created. However, for tracking to function, the instance name must not contain spaces or any of the following characters: `"[ . # | ] $ ? + = { / , }"`

`-U` *`user_id`*

Specify the user-id of the owner of all the files and directories of the instance. The default is the user-id of the current process. `-G` must be specified if you use this option.

`-G` *`group_id`*

Specify the group-id of the owner of all the files and directories of the instance. The default is the group-id of the current process. `-U` must be specified if you use this option.

`-Z` **{** `dev` **|** `prod` **}**

Specify the security model of the instance to development (`dev`) or secure (`prod`).

A typical use of this option is for testing web applications in a secure server environment before packaging and deploying.

Note: The `-Z prod` option does not create a production server. To actually create a production server, you must have a production server license.

`base_path`

Specify the pathname where you will create the instance.

Example

Create an instance of `/psc/pashome` in `/psc/acme1`:

```
$: /psc/pashome/bin/tcman.sh create -p 8501 -P 8601 -s 8701 /psc/acme1
Server instance acme1 created at /psc/acme1
```

See also

Delete an instance (delete) on page 103
Display and manage an instance's configuration (config) on page 104
Register an instance for tracking (register) on page 111
Start an instance (start) on page 115
Stop an instance (stop) on page 117

Delete an instance (delete)

Purpose

Remove the directory tree and all of the files in an instance. Alias tracking is disabled for servers that are removed.

To execute this action, the instance cannot be running.

Note: You cannot recover any files or directories removed by the `delete` action. Backup anything you want to save before launching this action.

Also note that you cannot use delete to remove the installed, root server (`$CATALINA_HOME`).

Syntax

```
tcman.sh delete [general_options] [-y] [base_path | alias_name]
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run `tcman.sh help delete` to see which general options are appropriate.

`-y`

Delete everything without prompting for confirmation.

base_path

Specify the pathname of the instance that you intend to delete.

alias_name

Refer to the instance that you intend to delete by its alias rather than its pathname.

Example

Delete the instance of `pashome` that was created in `/psc/acme3`:

```
$: /psc/pashome/bin/tcman.sh delete /psc/acme3
The following directory tree will be removed permanently:
 ( WARNING all deployed web applications will be DELETED!! )
/PAS/wrkdir/acme3
/PAS/wrkdir/acme3/conf
/PAS/wrkdir/acme3/temp
/PAS/wrkdir/acme3/common
/PAS/wrkdir/acme3/common/lib
/PAS/wrkdir/acme3/logs
/PAS/wrkdir/acme3/webapps
/PAS/wrkdir/acme3/webapps/ROOT
/PAS/wrkdir/acme3/webapps/ROOT/static
/PAS/wrkdir/acme3/webapps/ROOT/static/error
/PAS/wrkdir/acme3/webapps/ROOT/static/auth
/PAS/wrkdir/acme3/webapps/ROOT/META-INF
/PAS/wrkdir/acme3/webapps/ROOT/WEB-INF
/PAS/wrkdir/acme3/webapps/ROOT/WEB-INF/adapters
/PAS/wrkdir/acme3/webapps/ROOT/WEB-INF/adapters/rest/PingService
/PAS/wrkdir/acme3/webapps/ROOT/WEB-INF/adapters/soap
/PAS/wrkdir/acme3/webapps/ROOT/WEB-INF/classes
/PAS/wrkdir/acme3/webapps/ROOT/WEB-INF/classes/com
/PAS/wrkdir/acme3/webapps/ROOT/WEB-INF/classes/com/progress
/PAS/wrkdir/acme3/webapps/ROOT/WEB-INF/classes/com/progress/appserv
/PAS/wrkdir/acme3/work
/PAS/wrkdir/acme3/bin
Type 'yes' to continue
yes
Delete operation complete
server  removed at /PAS/wrkdir/acme3
```

See also

Create an instance (create) on page 101
Display server instances (instances) on page 108
The tcman command on page 84

Display and manage an instance's configuration (config)

Purpose

View, add, update, or delete the property values specified in `../conf/appserver.properties` and in `../conf/catalina.properties`.

When you run `tcman.sh config` with no parameters, it displays the core Tomcat server's configuration, and all the properties in both `.../conf/appserver.properties` and `.../conf/jvm.properties`. Note, however, that you can only view `jvm.properties`. You cannot modify its contents with the `config` action.

Syntax

```
tcman.sh config [general_options]
[prop_name | prop_name=value | +prop_name=value | ~prop_name]
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run `tcman.sh help config` to see which general options are appropriate.

prop_name

Display the specified property and its value.

prop_name=value

Set the value of a property that exists in `.../conf/appserver.properties`.

+prop_name=value

Add a new property to `.../conf/appserver.properties` and set its value.

~prop_name

Remove the specified property from `.../conf/appserver.properties`.

Examples

Show the configuration and properties of `acme1`, an instance of the core server, `pashome`:

```
$: /psc/acme1/bin/tcman.sh config
Using CATALINA_BASE:   /psc/acme1
Using CATALINA_HOME:   /psc/pashome
Using CATALINA_TMPDIR: /psc/acme1/temp
Using JRE_HOME:        /tools/linuxx86_64/java64/jdk1.7.0_02/
Using CLASSPATH:       /psc/pashome/bin/bootstrap.jar:/psc/pashome/bin/tomcat-juli.jar
Using CATALINA_PID:    /psc/acme1/temp/catalina.pid
Server version: Apache Tomcat/7.0.42
Server built:   Jul 2 2013 08:57:41
Server number:  7.0.42.0
OS Name:        Linux
OS Version:     2.6.18-164.el5
Architecture:   amd64
JVM Version:    1.7.0_02-b13
...
```

Display the value of a single property:

```
$: /psc/acme1/bin/tcman.sh config psc.as.http.port
psc.as.http.port=8501
```

Update the value of a property that exists in the `appserver.properties` file and then check the value:

```
$: /psc/acme1/bin/tcman.sh config psc.as.http.port=6543
$: tcman.sh config psc.as.http.port
psc.as.http.port=6543
```

Add a new property/value pair to the `appserver.properties` file and check the value:

```
$: /psc/acme1/bin/tcman.sh config +my.home.dir=/home/jarhead
$: tcman.sh config my.home.dir
my.home.dir=/home/jarhead
```

Update the server certificate in the `catalina.properties` file (see https://docs.oracle.com/cd/E19879-01/821-0185/ablqz/index.html for information about generating, exporting, and downloading a new server certificate):

```
$: /psc/acme1/bin/tcman.sh config psc.as.https.keyalias=myNewCert
```

Remove a property/value pair from the `appserver.properties` file and check if deletion was successful:

```
$: /psc/acme1/bin/tcman.sh config ~my.home.dir
$: tcman.sh config my.home.dir
Property does not exist - my.home.dir
```

Caution: There are no restrictions to property removal. The server will be unable to start if you remove a property required by `conf/server.xml`.

Notes

- All property names are case sensitive.
- You cannot enter multiple property names (*`prop_name`*) on the command line to view, update, or add properties to the `appserver.properties` file.
- You cannot use the config action to update existing values or add new values to the `jvm.properties` file

Display or modify the server features of an instance (feature)

Purpose

View, enable, or disable the server features contained in the `/conf/server.xml` file of an instance.

When you run `tcman.sh feature` with no parameters, it displays a list of the features (and their current status) that you can enable or disable. You can also display the status of a single server feature. After viewing the status of a feature, you can use `tcman.sh feature` to change its setting.

Syntax

```
tcman.sh feature [general_options] [feature_name [={on | off}]]
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run `tcman.sh help feature` to see which general options are appropriate.

feature_name

Specify one of the features defined in an instance's `conf/server.xml` file. Running `tcman.sh feature` without *feature_name* displays a list of all the features.

`on`

Enables the named feature.

`off`

Disables the named feature.

Example

Display the list of server feature settings for `acme1`, enable AJP13 (Apache JServ Protocol. version 1.3), and verify that the feature is enabled:

```
$: /psc/acme1/bin/tcman.sh feature
SecurityListener=off
JMXLifecycle=off
PSCRegistry=on
HTTP=onHTTPS=on
AJP13=off
Cluster=off
UserDatabase=on
JAASRealm=off
LDAPRealm=off
PASInstrument=off
RemoteHostValve=on
RemoteAddrValve=onSingleSignOn=on
AccessLog=on
CrawlerSessionManager=on
StuckSessionValve=on

$: /psc/acme1/bin/tcman.sh feature AJP13=on


$: /psc/acme1/bin/tcman.sh feature AJP13
AJP13=on
```

Notes

- Server features for instances are set in `$CATALINA_BASE/conf/server.xml`. You can change feature status by manually editing this file. However, it is safer to use `tcman.sh feature` to avoid corrupting the file with erroneous entries.
- Run `tcman.sh feature` when the instance is offline.

See also

Display and manage an instance's configuration (config) on page 104
Stop an instance (stop) on page 117

The tcman command on page 84

Clean up or archive server log files (clean)

Purpose

Truncate, move, or delete the log files located in the `/logs` directory of the core server or instance. If the server is running, clean truncates log files to zero length. If the server is not running, clean deletes the log files from the file system.

You have the option to save log files to a subdirectory of `/logs`.

Syntax

```
tcman.sh clean [general_options] [-A]
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run `tcman.sh help clean` to see which general options are appropriate.

`-A`

Archive log files to a subdirectory of `$CATALINA_BASE/logs`. The directory is automatically named with a month-day-year-second (*MM-DD-YYYY-ss*) time-stamp format. If the server is not running, the files in `$CATALINA_BASE/logs` are deleted.

Example

Archive the log files of `acme1`, an instance of the core server `pashome`, and save to a file:

```
/psc/pashome/tcman.sh clean -I acme1 -A
```

See also

The tcman command on page 84

Display server instances (instances)

Purpose

Show the names and locations of the instances created from the PAS installed in `$CATALINA_HOME` by displaying the contents of the file where instances are registered for tracking.

By default, instances are registered for tracking `$CATALINA_HOME/conf/instances.{windows|.unix}`. The file name extension indicates the OS platform where the PAS server is installed.

Syntax

```
tcman.sh instances [general_options]
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run `tcman.sh help instances` to see which general options are appropriate.

Output format

The following is the format of the output from a TCMAN `instances` action:

```
alias-name | full-file-path | type | state
```

alias-name

The user-defined name for the instance.

full-file-path

The location, in the OS file system, of the instance's root directory.

type

The designation of the server instance type (for example: `instance`, `service`, . . .).

state

An indication of the instance's validity. `OK` is returned for a valid server and `invalid` is returned for a corrupted or non-existant server.

Example

Display the instances of the core server installed in `/psc/pashome`:

```
/psc/pashome/bin/tcman.sh instances
acme1 | /psc/wrk/acme1 | instance | ok
acme2 | /psc/wrk/acme2 | instance | ok
```

Notes

- By default, instances are registered when you execute a `$CATALINA_HOME/bin/tcman{.sh|.bat} create` action, which automatically adds instance entries to an `instances` file. TCMAN removes instance entries from the file when you execute a `delete` action.

  You can manually add or remove instance entries from `instances` by using the `register` or `unregister` actions.

- By default, the name and location of the file where instances are registered is `$CATALINA_HOME/conf/instances.{windows|.unix}`.

  You can change the location of the instance registration file by adding and setting the `psc.as.instdir` property in the `appserver.properties` file. Use the TCMAN config action as in the following example:

  ```
  tcman.sh config '+psc.as.instdir=PATH'
  ```

  where *`PATH`* is a path name or an environment variable.

  You can also change the location and/or name of instance registration files by setting the environment variables, `PAS_AS_INSTANCE_DIR`, and `PAS_AS_INSTANCE_FILE`.

See also

Register an instance for tracking (register) on page 111
Stop tracking an instance (unregister) on page 120
The tcman command on page 84

List process ids (plist)

Purpose

List the process ids for all the processes that are running under an instance.

Syntax

```
tcman.sh plist [general_options] [-f]
```

Parameters

`general_options`

Specify one or more of the options that can be used with any TCMAN action. Run `tcman.sh help plist` to see which general options are appropriate.

`-f`

Display verbose output. The output is indented and uses the plus (+) character to indicate parent-child relationships.

Examples

Display process id's for the running instance, `acme1` using the `-v` and `-f` options:

```
/psc/acme1/bin/tcman.sh plist -v
info: showing process ids for server 5942
5942 5963 5975 5988 6001 6015

/psc/acme1/bin/tcman.sh plist -f
5942
 +5963
 +5975
 +5988
 +6001
 +6015
```

Notes

The plist action is useful for administrative tasks such as:

- Checking if an instance is stopped or running. If it is running, it returns the process id of the instance. If it is stopped, 0 is returned.
- Using the output (which is easily parsable) in administrative scripts.
- If the instance's applications have started sub-processes, multiple process ids will be returned.
- If multiple process ids are listed, the instance is always listed first.

See also

Show Windows process information (showproc) on page 114

Register an instance for tracking (register)

Purpose

Register an instance for tracking purposes.

Note:

Instances are automatically registered for tracking when you execute a `create` action. You use the `register` action to restart tracking on instances after tracking was stopped.

A typical use for unregistering and then re-registering an instance is to make configuration changes when moving instances from one location (core server) to another. The `register` action enables tracking and also updates the value of `CATALINA_HOME` in all of the executable scripts in the instance's `/bin` directory to refer to the new core server.

Syntax

```
tcman.sh register alias_name instance_path
```

Parameters

alias_name

Specify a meaningful name for the instance. The alias name must be unique in the instances file.

instance_path

Specify the OS file system path to where the instance exists. This value will be expanded into a fully qualified OS directory path and will be verified to exist.

Example

Track `test1`, which is an alias for the instance `/psc/acme1`:

```
/psc/pashome/bin/tcman.sh register test1 /psc/acme1
```

Notes

When you register an instance for tracking or create a new instance with the `create` command, an entry is created in the core Progress Application Server's `$CATALINA_HOME/conf/instances.[unix|windows]` file.

The `instances.[unix|windows]` file is a simple text file, which can be manually edited (with care) in the event that it becomes out of date. The format for entries is:

```
instance_name = base_path
```

An `instances.unix` file uses Unix OS file path syntax (forward slashes), and an `instances.windows` file uses Windows OS file path syntax (backslashes) to specify *base_path*.

Also note that in an instances file:

- Any line starting with a pound-sign (#) is a comment line.
- Blank lines are skipped.
- Instance names cannot contain spaces or any of the following characters: "`[ . # | ] $ ? + = { / , }`"

See also

Stop tracking an instance (unregister) on page 120
The tcman command on page 84

Register and manage an instance as a Windows service (service)

Purpose

(Windows only) Registers or unregisters an instance as a Windows service. After an instance is registered, you can start, stop, or check the status of the service with this action.

Note: Before you register an instance as a Windows service, you must install JDK 1.8 (for example, jdk1.8.0_66) and set the environment variable `JAVA_HOME=C:\Program Files\Java\jdk1.8.0_66`.

Syntax

```
tcman.bat service [general_options] alias_name { register | unregister | start
| stop | status }
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run `tcman.bat help service` to see which general options are appropriate.

alias_name

A required parameter that specifies the name of a PAS instance that was created using `tcman create`.

`register`

Create a new Windows service that runs the named PAS instance *alias_name*..

Set the `PR_DISPLAYNAME` and/or `PR_DESCRIPTION` variables to change the display name and description of the PAS instance service that appears in the Windows Service utility (**Services** tab of the Task Manager). The defaults for these variables are:

Variable	Default
`PR_DISPLAYNAME`	Progress Application Server *alias_name*
`PR_DESCRIPTION`	Progress Application Server (Tomcat 7) – http://www.progress.com

Set these variables before you register the instance. For example, if you wanted to change the defaults for `oepas1`:

```
set PR_DISPLAYNAME=PAS ROOT Server
set PR_DESCRIPTION=Progress Application Server
tcman service oepas1 register
```

`unregister`

Delete the Windows service that runs the named PAS instance *alias_name*

`start`

Start the Windows service that corresponds to the named PAS instance *alias_name*. The Windows service may also be started using the Windows service console or the SC command line utility.

`stop`

Stop the Windows service that corresponds to the named PAS instance *`alias_name`*. The Windows service may also be stopped using the Windows service console or the SC command line utility.

`status`

The registration status of the Windows service corresponding to the named PAS instance *`alias_name`*. The Windows service's status may be monitored using the Windows service console or SC command line utility

Example

Register the default instance oepas1 as a Windows, then start, check status, stop, and unregister:

```
tcman service oepas1 register
  oepas1 service is registered

tcman service oepas1 start
  oepas1 started

tcman service oepas1 status
  Service oepas1 is running

tcman service oepas1 stop
  oepas1 is stopped

tcman service oepas1 unregister
  oepas1 is unregistered
```

Note

Be sure that the instance is not running before you attempt to register/unregister it.

Show Windows process information (showproc)

Purpose

(Windows only) Show information about the process specified by a process id.

Syntax

```
tcman.bat showproc [general_options] [process-id]
```

Parameters

`general_options`

Specify one or more of the options that can be used with any TCMAN action. Run `tcman.sh help showproc` to see which general options are appropriate.

process-id

The numerical identifier of a Windows process. You can obtain a list of process ids by running the TCMAN `plist` action.

Examples

Display process id's for the running instance, `acme1`, then specify process ids to show detailed information.

```
/psc/acme1/bin/tcman.bat plist -v
info: showing process ids for server with window title 13332
13332 14240

/psc/acme1/bin/tcman.bat showproc 13332
ProcesName                : java
SessionId                 : 2
StartTime                 : 10/04/2015 16:29:42
Threads                   : 26
TotalProcessorTime        : 00:00:19.9213277
UserProcessTime           :
CPU (seconds)             : 19.9213277
Description               : Java(TM) Platform SE binary
Path                      : C:\Progress\OpenEdge\jdk\bin\java.exe

/psc/acme1/bin/tcman.bat showproc 14240
ProcesName                : _mproapsv
SessionId                 : 2
StartTime                 : 10/04/2015 16:29:54
Threads                   : 7
TotalProcessorTime        : 00:00:00.3744024
UserProcessTime           :
CPU (seconds)             : 0.3744024
Description               : OpenEdge AppServer (Multi-thread)
Path                      : C:\Progress\OpenEdge\bin\_mproapsv.exe
```

See also

List process ids (plist) on page 110

Start an instance (start)

Purpose

Start an instance of a PAS, optionally in debug mode.

Syntax

```
tcman.sh start [general_options] [-D|-J]
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run `tcman.sh help start` to see which general options are appropriate.

`-D`

Start the server in Tomcat debug mode. `-D` overrides the `-J` option.

`-J`

Start the server in debug mode using the JDPA (Java Platform Debugger Architecture) APIs for debugging. `-J` cannot be used if the `-D` option is specified.

Before you run a server with the `-J` option, you must define a port for the JDPA debugger by setting the `JDPA_ADDRESS` environment variable to a unique TCP network port number.

Example

Start the server in `/psc/acme1`, which is an instance of the core server in `/psc/pashome`:

```
/psc/acme1/bin/tcman.sh start
Using CATALINA_BASE:   /psc/acme1
Using CATALINA_HOME:   /psc/pashome
Using CATALINA_TMPDIR: /psc/acme1/temp
Using JRE_HOME:        /tools/linuxx86_64/java64/jdk1.7.0_02/
Using CLASSPATH:       /psc/pashome/bin/bootstrap.jar:/psc/pashome/bin/tomcat-juli.jar
Using CATALINA_PID:    /psc/acme1/temp/catalina.pid
```

Notes

- When the TCMAN utility starts the server, it verifies the creation of the OS process and then records the server's process-id in a `.pid` file. The location of the `.pid` file is:

OS	PID File Path
UNIX	`$CATALINA_BASE/temp/catalina-`*`instance_name`*`.pid`
Windows	`$CATALINA_BASE\logs\catalina-`*`instance_name`*`.pid`

- You can obtain the process id of a server by running the TCMAN `env` action.

See also

Stop an instance (stop) on page 117
The tcman command on page 84

Stop an instance (stop)

Purpose

Stop a running instance, either gracefully or forcibly.

Note: TCMAN supports stopping a server instance that is not configured with a shutdown port.

On UNIX platforms stopping the running server instance is accomplished by sending a UNIX signal to the PAS process. Therefore, the administrator's process must have the UNIX permissions to signal the PAS process. On Windows platforms, the instance is identified using an OS process id that is used to stop server processes.

Syntax

```
tcman.sh stop [general_options] [-F[-w seconds]]
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run `tcman.sh help stop` to see which general options are appropriate.

`-F`

Kill the sever process if it does not stop after a default wait time (5 seconds on UNIX, 10 seconds on Windows). Change the default wait interval by using the `-w` option.

`-w` *seconds*

Optionally specify the number of seconds to wait before killing a server process.

Example

Stop the server in `/psc/acme1`, which is an instance of the core server in `/psc/pashome`:

```
/psc/acme1/bin/tcman.sh stop
Using CATALINA_BASE:   /psc/acme1
Using CATALINA_HOME:   /psc/pashome
Using CATALINA_TMPDIR: /psc/acme1/temp
Using JRE_HOME:        /tools/linuxx86_64/java64/jdk1.7.0_02/
Using CLASSPATH:       /psc/pashome/bin/bootstrap.jar:/psc/pashome/bin/tomcat-juli.jar
Using CATALINA_PID:    /psc/acme1/temp/catalina.pid
```

Notes

- TCMAN supports stopping a server instance that is not configured with a shutdown port.

On UNIX platforms stopping the running server instance is accomplished by sending a UNIX signal to the PAS process. Therefore, the administrator's process must have the UNIX permissions to signal the PAS process. On Windows platforms, the instance is identified using an OS process id that is used to stop server processes.

The following is an example a message you would see after a forced shut down with no shut down port:

```
Sep 23, 2013 4:10:47 PM org.apache.catalina.startup.Catalina stopServer
SEVERE: No shutdown port configured. Shut down server through OS signal.
Server not shut down.
Killing Tomcat with the PID: 14230
```

- Process ids are stored in the following locations:

OS	PID File Path
UNIX	`$CATALINA_BASE/temp/catalina-`*`instance_name`*`.pid`
Windows	`$CATALINA_BASE\logs\catalina-`*`instance_name`*`.pid`

- You can also obtain the process id of a server by running the TCMAN `env` action.

See also

Start an instance (start) on page 115
The tcman command on page 84

Display server, OS, and runtime version information (version)

Purpose

Show the Apache Tomcat runtime version and OS information for an instance.

To execute this action, the instance cannot be running

Syntax

```
tcman.sh version [general_options]
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run `tcman.sh help version` to see which general options are appropriate.

Example

Display the server and runtime information for `acme1`, an instance of the core server installed in `/psc/pashome`:

```
$: /psc/pashome/bin/tcman.sh version -I acme1
Using CATALINA_BASE:   /psc/acme1
Using CATALINA_HOME:   /psc/pashome
Using CATALINA_TMPDIR: /psc/acme1/temp
Using JRE_HOME:        /tools/linuxx86_64/java64/jdk1.7.0_02/
Using CLASSPATH:
/psc/pashome/bin/bootstrap.jar:/users/doc/agarbacz/psc/pashome/bin/tomcat-juli.jar
Using CATALINA_PID:    /psc/acme1/temp/catalina.pid
Server version: Apache Tomcat/7.0.42
Server built:   Jul 2 2013 08:57:41
Server number:  7.0.42.0
OS Name:        Linux
OS Version:     2.6.18-164.el5
Architecture:   amd64
JVM Version:    1.7.0_02-b13
JVM Vendor:     Oracle Corporation
```

See also

Display OS and server information (info) on page 90
The tcman command on page 84

Test a server configuration (test)

Purpose

Displays information on the configuration and environment of an instance. It also displays information about error conditions.

The `test` action starts a server (instance), loads all the configuration files, and then displays information. The instance is stopped, exiting gracefully even if there is an error condition.

To execute this action, the instance cannot be running.

Syntax

```
tcman.sh test [general_options]
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run `tcman.sh help test` to see which general options are appropriate.

Example

Run a test of the configuration of `acme1`, which is an instance of the core server installed at `/psc/pashome`:

```
$: /psc/pashome/bin/tcman.sh -I acme1 test
Using CATALINA_BASE:   /psc/acme1
Using CATALINA_HOME:   /psc/pashome
Using CATALINA_TMPDIR: /psc/acme1/temp
Using JRE_HOME:        /tools/linuxx86_64/java64/jdk1.7.0_02/
Using CLASSPATH:       /psc/pashome/bin/bootstrap.jar:/psc/pashome/bin/tomcat-juli.jar
Using CATALINA_PID:    /psc/acme1/temp/catalina.pid
. . .
```

Notes

The `test` action is particularly useful for testing to verify that a server will start and run properly after you make changes to configuration and properties files.

See also

The tcman command on page 84

Stop tracking an instance (unregister)

Purpose

Stop tracking an instance by removing the instance's entry from the `$CATALINA_HOME/conf/instances.[unix|windows]` file.

Note:

You use the `register` action to restart tracking on instances after tracking was stopped with `unregister` .

A typical use for unregistering and then re-registering an instance, is to make configuration changes when moving instances from one location, or core server, to another. The `register` action not only enables tracking, it also updates the value of `CATALINA_HOME` in all of the executable scripts in the instance's `/bin` directory to refer to the new core server.

Syntax

```
tcman.sh unregister alias_name
```

Parameters

alias_name

Specify the alias name of the instance that you want to stop tracking. The alias name must exist in an `instances.[unix|windows]` file.

Example

Stop tracking `test1`, which is an instance of `/psc/pashome`:

```
/psc/pashome/bin/tcman.sh unregister test1
```

See also

Register an instance for tracking (register) on page 111
The tcman command on page 84

Create a Tomcat worker configuration file (workers)

Purpose

Create a preliminary `worker.properties` file that supports the configuration of supporting servers (workers) for a Progress Application Server (PAS) instance.

in the Apache Reference Guide, a worker is defined as an "instance that is waiting to execute servlets or any other content on behalf of some web server." In the context of the Progress Application Server, a worker is a server that is called by a PAS instance to perform a specific task. Typically,you would define worker instances to manage proxies, load balancing, clusters, or status monitoring. (For links to information on this functionality, see the Apache Tomcat Documentation Index.) There are probably other situations where you could improve the performance of a server instance by configuring worker instances to handle specific processing tasks.

In Apache Tomcat, workers are configured in a `worker.properties` file. The protocol implemented for communication between servers and workers is the Apache JServ Protocol (version 1.3, referred to as AJP13).

In TCMAN, the `workers` action adds the definitions of registered PAS instances to the content of the `$CATALINA_HOME/extras/workers.template` file and puts the result in `$CATALINA_HOME/temp/worker.properties`. The template file supplies a set of common directives that are referenced by all of the defined PAS instances. Individual instance definitions contain only the properties that are unique to the instance, such as the AJP13 network connection port. (See Table 28: worker.properties example on page 123.)

The `/temp/worker.properties` created by the `workers` action is a preliminary configuration file that you will probably need to modify to implement your deployment. See The Apache Tomcat Connector-Reference Guide for more information about configuring workers.

Syntax

```
$CATALINA_HOME/tcman.sh workers [general_options] [worker_list]
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run `tcman.sh help worker` to see which general options are appropriate.

`worker_list`

A comma separated list of instance names and/or keywords. The keywords are:

status	Include an instance that has been implemented as a status server
lb	Include an instance that has been implemented as a load balance server
home	Include the CATALINA_HOME core server
all	Include all registered instances

If no *`worker_list`* is specified, the default worker list (all instances registered to CATALINA_HOME) will be added. If no instances have been created, then the default *`worker_list`* is CATALINA_HOME.

Examples

Assume there are:

- Two registered instances (`piw1` and `piw2`) that serve Web applications
- A Tomcat load balancer instance (`jklb`) that distributes the workload between `piw1` and `piw2`
- A status instance (`jkstatus`) that is used to monitor the runtime status of `piw1` and `piw2`

The following are examples of worker-lists showing various combinations of keywords and instances, and the resulting content in `$CATALINA_HOME/temp/worker.properties`:

Table 27: worker-list keywords

worker-list	Resulting content in worker.properties
`blank`	Default entries from `worker.template` plus entries for `piw1` and `piw2`
`piw1`	Default entries from `worker.template` plus an entry for `piw1`
`all`	Default entries from `worker.template` plus entries for `piw1` and `piw2`
`home`	Default entries from `worker.template` plus an entry for core server (CATALINA_HOME)
`home, all`	Default entries from `worker.template` plus an entry for core server, (CATALINA_HOME), and entries for for `piw1` and `piw2`
`lb, status`	Default entries from `worker.template` plus entries for `jklb`, `jstatus`, `piw1`, and `piw2` **Note:** When no registered instance is specified, all registered instances are automatically included.
`lb, status, home, all`	Default entries from `worker.template` plus entries for `jklb`, `jstatus`, the core server (CATALINA_HOME),`piw1`, and `piw2`

The following is an example `workers.properties` file that includes entries for instances `piw1` and `piw2`:

Table 28: worker.properties example

```
# worker properties for instances rooted at U:\vobs\dlc\servers\pasw
# build date: 03/24/2014 11:09:23

# List of worker server instances
worker.list=piw1,piw2

#
# Global properties
#
# worker.maintain=60

#
# Common worker properties referenced by individual workers
#
worker.common.type=ajp13
worker.common.host=${psc.as.host.name}
worker.common.socket_timeout=10
worker.common.connect_timeout=10000
worker.common.socket_keepalive=true
worker.common.ping_mode=I
worker.common.ping_timeout=10000
worker.common.connect_timeout=0
worker.common.retry_interval=100
worker.common.recovery_options=7
# worker.common.connection_ping_interval=10000
# worker.common.fail_on_status=0# worker.common.max_packet_size=8192
# worker.common.recover_time=60

# properties for alias piw1 with jvmRoute piw1
worker.piw1.port=9996
worker.piw1.reference=worker.common

# properties for alias piw2 with jvmRoute piw2
worker.piw2.port=9996
worker.piw2.reference=worker.common
```

Notes

- The `tcman workers` action must be run from the PAS installation's `$CATALINA_HOME/bin directory`.
- The `/extras/workers.template` file can be modified to adjust existing properties or to add additional static information. However, you cannot replace the common properties with a unique set of properties for each defined server.

General actions

This section details the actions available for displaying help and server runtime environment information.

See also

The tcman command on page 84

Display help (help)

Purpose

Display summary or detailed help for all TCMAN actions, property names, and server features.

Syntax

```
tcman.sh help [action | property | feature]
```

Parameters

action

Show the syntax and options of the specified action. If no action is specified, show a list of all actions and the general options.

property

Show the settings for specified property.

feature

Show if the specified feature is enabled or disabled.

Example

Display the usage help for the `create` action:

```
$: tcman.sh help create
usage: tcman create [options] -p <http-port> [instance-opts] <new-base-path>
    instance-opts:
       [-s <shutdown-port>]
       [-P <https-port>]
       [-j <ajp13-port>]
       [-W <web-apps-dir>]
       [-N <inst-alias-name>]
       [-U <file-owner> -G <file-group>]

   general options:
      -u uid:pwd  pass uid and pwd for HTTP BASIC authentication
      -v          print verbose output
      -M url      override the CATALINA_BASE manager's URL with
                  <{http|https}://<host>:<port>/<mgr-app>
      -B          override CATALINA_BASE environment setting
      -n          debug run action but do not execute changes
```

Display runtime environment information (env)

Purpose

Show details about a server's state.

Syntax

```
tcman.sh env [general_options] [keyword]
```

Parameters

general_options

Specify one or more of the general TCMAN options. Run `tcman.sh help env` to see which general options are appropriate.

keyword

Specify one or more keywords that represent the name of the state that you want to view. If no keyword is specified, then all of the state information is displayed.

Keywords include:

Keyword	Description
`running`	Indicate if a server is running (`1`) or not running (`0`).
`mgrurl`	Display the URL of the manager application.
`type`	Display the server type.
`alias`	Display the server's alias.
`parent`	Display the pathname of the parent of an instance.
`tracking`	Indicate if tracking is on (`1`) or off (`0`).
`http`	Display the server's http port number.
`https`	Display the server's https port number.
`shut`	Display the server's shutdown port number. A value of `-1` indicates that there is no shutdown port.
`pid`	Display the server's process id. A hyphen (`-`) indicates that the server is not running.

Example

Display all of the state information for the instance created in `/psc/acme1`:

```
/psc/acme1/bin/tscman.sh env
catalina home:     /psc/pashome
catalina base:     /psc/acme1
java home:         /tools/linuxx86_64/java64/jdk1.7.0_02/
jre home:
manager http port: 8501
manager https port:8601
manager shut port: 8701
manager URL:       http://localhost:8501/manager
config type:       instance
config alias:      acme1
config parent:     /psc/pashome
server running:    0
instance tracking: 1
instance file:     /psc/pashome/conf/instances.unix
server process-id: -
```

Index

A

ABL application logging *53*
ABL application metrics *50*
ABL application startup environment *51*
ABL application's advanced settings *57*
ABL service network environment *56*
APSV transport *74*

C

Catalina log file monitor *30*
Catalina log file viewer *31*
clean action *108*
config action *104*
Configure ABL application *51*
Configure properties *62*, *69*
Configuring properties
    Nest index terms for sub entries *74*, *78*
create *101*
Customize log file monitors *30*

D

Default rules *27*
delete *103*
deploy *91*
disable *98*

E

enable *97*
env *124*

F

feature action *106*

H

help *124*

I

info *90*
instances *108*

L

leaks *96*
list *89*
log file monitor *31*
log file monitors *30*

M

Manage APSV transports *73*
Manage REST transports *59*
Manage SOAP transports *66*
Manage WEB transports *77*
Managing *80*
Monitoring plans *27*

P

PAS Agents Utilization High rule *28*
PAS rules *28*
PAS Sessions Utilization High rule *28*
plist *110*

R

register *111*
reload *94*
resources *99*
REST properties *62*
Rule sets *27*
Rules for default monitoring plan *28*
Rules with polled alerts *28*

S

service *112*
sessions *100*
showproc *114*
SOAP transport properties *69*
start *115*
status *95*
stop *117*

T

tcman *84*
TCMAN
    clean *108*
    config *104*
    create *101*
    customizing *86*
    delete *103*
    deploy *91*

TCMAN *(continued)*
disable *98*
enable *97*
env *124*
extending *86*
feature *106*
general actions *123*
help *124*
info *90*
instances *108*
leaks *96*
list *89*, *110*
manager actions *88*
register *111*
reload *94*
resources *99*
server actions *100*
service *112*
sessions *100*
showproc *114*
start *115*
status *95*
stop *117*
test *119*
undeploy *93*
TCMAN *(continued)*
unregister *120*
using *83*
version *118*
worker *121*
test *119*
Tuning parameters and limits *54*

U

undeploy *93*
unregister *120*

V

version *118*
View metrics *50*
Viewing metrics *79*

W

WEB handlers *80*
WEB transport *78*
WEB transports *79*
worker *121*