PROGRESS OPENEDGE

OpenEdge Management and OpenEdge Explorer: Configuration
Third party acknowledgements — See the table of contents for the "Third Party Acknowledgements" appendix.
1. Introduction ............................................... 21
   OpenEdge Management and OpenEdge Explorer .......................... 22
      OpenEdge Management ........................................ 22
      OpenEdge Explorer ......................................... 23
   Getting started .............................................. 24
      Comparing OpenEdge Management and OpenEdge Explorer ............. 24

2. Configuring OpenEdge Databases ................................. 27
   Database configuration and administration ................................ 28
      Viewing the database log file .................................. 29
      Database configuration awareness .................................. 29
   Managing databases ............................................. 30
   Viewing or modifying database properties ................................. 31
   Viewing the various database configuration properties .................. 33
      Editing the database default configuration ........................ 46
      Creating a database configuration ................................ 46
      Assigning a different default database configuration ................ 47
   Viewing the server group ........................................ 48
      Editing a database server group .................................. 51
      Creating a database server group .................................. 52
   Starting or stopping the database ................................... 53
      Starting a database automatically ................................. 53
   Using the background writers and watchdog processes .................... 54
      Background writers ............................................. 54
      The watchdog process .......................................... 54
   Viewing database status ......................................... 55
      Viewing database process details ................................ 56
   Deleting a database, database configuration, or database server group .... 58
      Deleting a database ............................................. 58
      Deleting a database configuration ................................ 58
      Deleting a database server group ................................ 59
   Configuring an OpenEdge Replication-enabled database .................. 60
### Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3. <strong>Configuring DataServers.</strong></td>
<td>71</td>
</tr>
<tr>
<td>ODBC DataServer configuration and administration</td>
<td>72</td>
</tr>
<tr>
<td>Working with the ODBC DataServer.</td>
<td>72</td>
</tr>
<tr>
<td>Configuring an ODBC DataServer broker</td>
<td>73</td>
</tr>
<tr>
<td>Editing an ODBC DataServer broker configuration</td>
<td>79</td>
</tr>
<tr>
<td>Creating an ODBC DataServer broker</td>
<td>79</td>
</tr>
<tr>
<td>Deleting an ODBC DataServer broker</td>
<td>80</td>
</tr>
<tr>
<td>Starting or Stopping an ODBC DataServer broker</td>
<td>80</td>
</tr>
<tr>
<td>Starting an ODBC DataServer broker automatically.</td>
<td>80</td>
</tr>
<tr>
<td>Viewing the status of an ODBC DataServer broker</td>
<td>81</td>
</tr>
<tr>
<td>Viewing the ODBC DataServer log files</td>
<td>83</td>
</tr>
<tr>
<td>Oracle DataServer configuration and administration</td>
<td>85</td>
</tr>
<tr>
<td>Working with the Oracle DataServer broker</td>
<td>85</td>
</tr>
<tr>
<td>Configuring an Oracle DataServer broker</td>
<td>86</td>
</tr>
<tr>
<td>Editing an Oracle DataServer broker configuration</td>
<td>93</td>
</tr>
<tr>
<td>Creating an Oracle DataServer broker</td>
<td>93</td>
</tr>
<tr>
<td>Deleting an Oracle DataServer broker</td>
<td>94</td>
</tr>
<tr>
<td>Starting or Stopping an Oracle DataServer broker</td>
<td>94</td>
</tr>
<tr>
<td>Starting an Oracle DataServer broker automatically.</td>
<td>94</td>
</tr>
<tr>
<td>Viewing the status of an Oracle DataServer broker</td>
<td>95</td>
</tr>
<tr>
<td>Viewing the Oracle DataServer broker log files</td>
<td>97</td>
</tr>
<tr>
<td>MS SQL Server DataServer configuration and administration</td>
<td>99</td>
</tr>
<tr>
<td>Working with the MS SQL Server DataServer broker</td>
<td>99</td>
</tr>
<tr>
<td>Configuring an MS SQL Server DataServer broker</td>
<td>100</td>
</tr>
<tr>
<td>Editing an MS SQL Server DataServer broker configuration</td>
<td>106</td>
</tr>
<tr>
<td>Creating an MS SQL Server DataServer broker</td>
<td>107</td>
</tr>
<tr>
<td>Deleting an MS SQL Server DataServer broker</td>
<td>107</td>
</tr>
<tr>
<td>Starting or Stopping an MS SQL Server DataServer broker</td>
<td>108</td>
</tr>
<tr>
<td>Starting an MS SQL Server DataServer broker automatically</td>
<td>108</td>
</tr>
<tr>
<td>Viewing the status of an MS SQL Server DataServer broker</td>
<td>109</td>
</tr>
<tr>
<td>Viewing the MS SQL Server DataServer log files</td>
<td>111</td>
</tr>
<tr>
<td>4. <strong>Configuring NameServers.</strong></td>
<td>113</td>
</tr>
<tr>
<td>NameServer configuration and administration</td>
<td>114</td>
</tr>
<tr>
<td>NameServer features</td>
<td>114</td>
</tr>
<tr>
<td>Working with the NameServer</td>
<td>115</td>
</tr>
<tr>
<td>NameServer as an optional client connection request handler.</td>
<td>115</td>
</tr>
<tr>
<td>Local and remote NameServers</td>
<td>115</td>
</tr>
<tr>
<td>Configuring a NameServer</td>
<td>116</td>
</tr>
<tr>
<td>Configuring local instances</td>
<td>116</td>
</tr>
<tr>
<td>Configuring remote instances</td>
<td>123</td>
</tr>
<tr>
<td>Editing a NameServer configuration</td>
<td>124</td>
</tr>
<tr>
<td>Setting or deleting NameServer environment variables</td>
<td>124</td>
</tr>
<tr>
<td>Creating a NameServer</td>
<td>126</td>
</tr>
<tr>
<td>Deleting a NameServer</td>
<td>127</td>
</tr>
<tr>
<td>Configuring fault tolerance and load balancing</td>
<td>128</td>
</tr>
<tr>
<td>Connection-level fault tolerance</td>
<td>128</td>
</tr>
<tr>
<td>Server-level fault tolerance</td>
<td>128</td>
</tr>
<tr>
<td>Load balancing</td>
<td>129</td>
</tr>
</tbody>
</table>
5. Configuring AppServers and AppServer Internet Adapters

AppServer configuration and administration ............................................. 136
   Understanding AppServer configuration ........................................... 136
   AppServer features ........................................................................... 137
   Working with the AppServer ............................................................... 137
   AppServer configuration properties ..................................................... 138
   Configuring an AppServer ................................................................. 139
   Editing an AppServer configuration .................................................... 163
   Creating an AppServer ..................................................................... 164
   Starting and stopping an AppServer .................................................... 164
   Adding or trimming AppServer agents ............................................... 166
   Viewing the status of an AppServer .................................................... 167
   Viewing AppServer process details ..................................................... 171
   Stopping an AppServer agent ............................................................. 172
   Killing an AppServer agent ............................................................... 173
   Deleting an AppServer ..................................................................... 173
   Viewing the AppServer log files ......................................................... 173
For additional AppServer information ..................................................... 174

AppServer Internet Adapter configuration and administration ... 175
   Working with the AppServer Internet Adapter .................................... 175
   Configuring an AppServer Internet Adapter ....................................... 176
   Editing an AppServer Internet Adapter configuration ........................ 182
   Creating an AppServer Internet Adapter ............................................. 183
   Viewing the AppServer Internet Adapter log file ............................... 183
   Enabling or disabling an AppServer Internet Adapter broker .......... 185
   Deleting an AppServer Internet Adapter broker .............................. 185

6. Configuring OpenEdge SOAP Web Services ........................................ 187
Configuring and managing SOAP Web services .................................... 188
   SOAP-AVM gateway ....................................................................... 188
   Web Service Application Server ....................................................... 188
   Managing SOAP Web services with the WSA .................................... 189
   SOAP Web service session models and the WSA ............................... 189
Managing Web Services Adapter (for OpenEdge SOAP Web Services) instances ........................................... 191
   Creating a new Web Services Adapter (for OpenEdge SOAP Web Services) instance ........................................... 192
   Configuring a Web Services Adapter (for OpenEdge SOAP Web Services) instance ........................................... 193
   Editing a Web Services Adapter (for OpenEdge SOAP Web Services) instance configuration ........................................... 204
   Starting Web Services Adapter (for OpenEdge SOAP Web Services) instances ................................................... 205
   Setting SOAP Web service defaults ................................................ 206
Deploying a SOAP Web service to a Web Services Adapter (for OpenEdge SOAP Web Services) instance ..................... 207
Exporting a SOAP Web service ........................................................ 209
Importing a SOAP Web service to a Web Services Adapter (for OpenEdge SOAP Web Services) instance ..................... 210
<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>OpenEdge Explorer functionality by resource</td>
<td>23</td>
</tr>
<tr>
<td>2</td>
<td>OEM and OEE feature comparison</td>
<td>25</td>
</tr>
<tr>
<td>3</td>
<td>Database properties</td>
<td>31</td>
</tr>
<tr>
<td>4</td>
<td>General database configuration properties</td>
<td>34</td>
</tr>
<tr>
<td>5</td>
<td>Background writers database configuration properties</td>
<td>36</td>
</tr>
<tr>
<td>6</td>
<td>After Image Management database configuration properties</td>
<td>38</td>
</tr>
<tr>
<td>7</td>
<td>International database configuration properties</td>
<td>38</td>
</tr>
<tr>
<td>8</td>
<td>Advanced database configuration properties</td>
<td>40</td>
</tr>
<tr>
<td>9</td>
<td>Statistics Collection Configuration database properties</td>
<td>44</td>
</tr>
<tr>
<td>10</td>
<td>SQL-92 Configuration database properties</td>
<td>44</td>
</tr>
<tr>
<td>11</td>
<td>SSL Configuration database properties</td>
<td>45</td>
</tr>
<tr>
<td>12</td>
<td>Agent database configuration property</td>
<td>45</td>
</tr>
<tr>
<td>13</td>
<td>General database server group properties</td>
<td>49</td>
</tr>
<tr>
<td>14</td>
<td>Advanced database server group properties</td>
<td>50</td>
</tr>
<tr>
<td>15</td>
<td>Process statistics data</td>
<td>57</td>
</tr>
<tr>
<td>16</td>
<td>ODBC DataServer broker properties</td>
<td>73</td>
</tr>
<tr>
<td>17</td>
<td>ODBC DataServer server properties</td>
<td>77</td>
</tr>
<tr>
<td>18</td>
<td>ODBC DataServer broker status summary</td>
<td>81</td>
</tr>
<tr>
<td>19</td>
<td>ODBC DataServer broker status details</td>
<td>82</td>
</tr>
<tr>
<td>20</td>
<td>Oracle DataServer broker properties</td>
<td>86</td>
</tr>
<tr>
<td>21</td>
<td>Oracle DataServer server properties</td>
<td>91</td>
</tr>
<tr>
<td>22</td>
<td>Oracle DataServer broker property Status summary</td>
<td>95</td>
</tr>
<tr>
<td>23</td>
<td>Oracle DataServer broker Status details</td>
<td>96</td>
</tr>
<tr>
<td>24</td>
<td>MS SQL Server DataServer broker properties</td>
<td>100</td>
</tr>
<tr>
<td>25</td>
<td>MS SQL Server DataServer server properties</td>
<td>104</td>
</tr>
<tr>
<td>26</td>
<td>MS SQL Server DataServer broker property status summary</td>
<td>109</td>
</tr>
<tr>
<td>27</td>
<td>MS SQL Server DataServer broker status details</td>
<td>110</td>
</tr>
<tr>
<td>28</td>
<td>NameServer properties</td>
<td>117</td>
</tr>
<tr>
<td>29</td>
<td>NameServer status summary properties</td>
<td>131</td>
</tr>
<tr>
<td>30</td>
<td>NameServer status detail properties</td>
<td>132</td>
</tr>
<tr>
<td>31</td>
<td>Ubroker.properties file structure</td>
<td>138</td>
</tr>
<tr>
<td>32</td>
<td>AppServer Broker General properties</td>
<td>140</td>
</tr>
<tr>
<td>33</td>
<td>AppServer Broker Owner Information properties</td>
<td>140</td>
</tr>
<tr>
<td>34</td>
<td>AppServer Broker Controlling NameServer properties</td>
<td>140</td>
</tr>
<tr>
<td>35</td>
<td>AppServer Broker AppService Name List properties</td>
<td>141</td>
</tr>
<tr>
<td>36</td>
<td>AppServer Broker Logging Setting properties</td>
<td>142</td>
</tr>
<tr>
<td>37</td>
<td>AppServer Broker Advanced Features properties</td>
<td>146</td>
</tr>
<tr>
<td>38</td>
<td>AppServer Agent General properties</td>
<td>151</td>
</tr>
<tr>
<td>39</td>
<td>AppServer Agent Logging Setting properties</td>
<td>152</td>
</tr>
<tr>
<td>40</td>
<td>AppServer Agent Pool Range properties</td>
<td>156</td>
</tr>
<tr>
<td>41</td>
<td>AppServer Agent Advanced Features properties</td>
<td>156</td>
</tr>
<tr>
<td>42</td>
<td>AppServer SSL General properties</td>
<td>158</td>
</tr>
<tr>
<td>43</td>
<td>AppServer SSL Advanced Features properties</td>
<td>159</td>
</tr>
<tr>
<td>44</td>
<td>AppServer Messaging properties</td>
<td>161</td>
</tr>
<tr>
<td>45</td>
<td>Process statistics data</td>
<td>172</td>
</tr>
<tr>
<td>46</td>
<td>AIA General properties</td>
<td>177</td>
</tr>
<tr>
<td>47</td>
<td>AIA Controlling NameServer properties</td>
<td>177</td>
</tr>
<tr>
<td>48</td>
<td>AIA Logging Setting properties</td>
<td>179</td>
</tr>
<tr>
<td>49</td>
<td>AIA SSL properties</td>
<td>181</td>
</tr>
<tr>
<td>50</td>
<td>AIA Advanced Features properties</td>
<td>182</td>
</tr>
<tr>
<td>51</td>
<td>Web Services Adapter (for OpenEdge SOAP Web Services) Location properties</td>
<td>195</td>
</tr>
<tr>
<td>52</td>
<td>Web Services Adapter (for OpenEdge SOAP Web Services) Proxy Server Setting properties</td>
<td>196</td>
</tr>
<tr>
<td>53</td>
<td>Web Services Adapter (for OpenEdge SOAP Web Services) WSDL properties</td>
<td>196</td>
</tr>
</tbody>
</table>
Table 54: Web Services Adapter (for OpenEdge SOAP Web Services) Logging Setting properties ............................................................... 197
Table 55: Web Services Adapter (for OpenEdge SOAP Web Services) Security properties ............................................................... 200
Table 56: Web Services Adapter (for OpenEdge SOAP Web Services) Advanced Features properties ................................................. 202
Table 57: WebSpeed Messenger General properties ............................................................... 225
Table 58: WebSpeed Messenger Controlling NameServer properties ............................................................... 225
Table 59: WebSpeed Messenger Logging Setting properties ............................................................... 227
Table 60: WebSpeed Messenger SSL properties ............................................................... 228
Table 61: WebSpeed Messenger Advanced properties ............................................................... 229
Table 62: WebSpeed Broker General properties ............................................................... 233
Table 63: WebSpeed Broker Owner Information properties ............................................................... 233
Table 64: WebSpeed Broker Controlling NameServer properties ............................................................... 234
Table 65: WebSpeed Broker AppService Name List properties ............................................................... 235
Table 66: WebSpeed Broker Logging Setting properties ............................................................... 235
Table 67: WebSpeed Broker Advanced Features properties ............................................................... 239
Table 68: WebSpeed Agent General properties ............................................................... 243
Table 69: WebSpeed Agent Logging Setting properties ............................................................... 245
Table 70: WebSpeed Agent Pool Range properties ............................................................... 248
Table 71: WebSpeed Agent Advanced Features properties ............................................................... 249
Table 72: WebSpeed SSL General properties ............................................................... 251
Table 73: WebSpeed SSL Advanced Features properties ............................................................... 252
Table 74: WebSpeed Messaging properties ............................................................... 252
Table 75: SonicMQ Adapter Broker General properties ............................................................... 265
Table 76: SonicMQ Adapter Broker Owner Information properties ............................................................... 265
Table 77: SonicMQ Adapter Broker Controlling NameServer properties ............................................................... 266
Table 78: SonicMQ Adapter Broker AppServer Name List property ............................................................... 266
Table 79: SonicMQ Adapter Broker Logging Setting properties ............................................................... 267
Table 80: SonicMQ Adapter Broker Advanced Features properties ............................................................... 268
Table 81: SonicMQ Adapter Server General property ............................................................... 270
Table 82: SonicMQ Adapter Server Logging Settings properties ............................................................... 270
Table 83: SonicMQ Adapter Server Pool Range properties ............................................................... 271
Table 84: SonicMQ Adapter SSL General properties ............................................................... 272
Table 85: SonicMQ Adapter SSL Advanced Features properties ............................................................... 273
Table 86: OE Web Server Location properties ............................................................... 290
Table 87: OE Web Server Proxy Server Setting properties ............................................................... 291
Table 88: OE Web Server Security properties ............................................................... 291
Preface

This Preface contains the following sections:

- Purpose
- Audience
- Organization
- Using this manual
- Typographical conventions
- Examples of syntax descriptions
- OpenEdge messages
Purpose

This manual describes how to establish property and configuration settings for OpenEdge databases, DataServers (for ODBC, Oracle, and MS SQL Server), NameServers, AppServers, AppServer Internet Adapters, Web Services Adapters (for OpenEdge SOAP Web Services), REST Web Services, WebSpeed® Transaction Servers and Messengers, and SonicMQ® Adapters in OpenEdge® Management and OpenEdge Explorer. In addition, the manual also provides detail about viewing status and log files.

Audience

This manual is intended for OpenEdge Management and OpenEdge Explorer users as well as OpenEdge Management database and system administrators.

Organization

Chapter 1, “Introduction”

Introduces OpenEdge Management and OpenEdge Explorer and describes their features.

Chapter 2, “Configuring OpenEdge Databases”

Describes how to establish and modify property and configuration settings for a database, database configuration, or database server group; add a database instance; start and stop a database; use the background writers and watchdog process; view status; view the log file; and delete a database.

Chapter 3, “Configuring DataServers”

Describes how to establish and modify property and configuration settings for an ODBC, Oracle, or MS SQL Server DataServer; add a DataServer instance; start and stop a DataServer; view status; view the log files; and delete a DataServer.

Chapter 4, “Configuring NameServers”

Describes how to establish and modify property and configuration settings for a NameServer; add a NameServer instance; start and stop a NameServer; work with fault tolerance and load balancing; view status; view the log files; and delete a NameServer.

Chapter 5, “Configuring AppServers and AppServer Internet Adapters”

Describes how to establish and modify property and configuration settings for an AppServer or AppServer Internet Adapter; add an AppServer or AppServer Internet Adapter instance; add or trim AppServer agents; start and stop an AppServer; enable or disable an AppServer Internet Adapter; view status; view the log files; and delete an AppServer or AppServer Internet Adapter.
Chapter 6, “Configuring OpenEdge SOAP Web Services”

Describes how to configure and manage SOAP Web services; manage Web Services Adapter instances (for OpenEdge SOAP Web Services); and work with SOAP Web services.

Chapter 7, “Configuring WebSpeed Messengers and WebSpeed Transaction Servers”

Describes how to establish and modify property and configuration settings for a WebSpeed Transaction Server or WebSpeed Messenger; create or delete a WebSpeed Transaction Server instance; start and stop a WebSpeed Transaction Server; add or trim WebSpeed Transaction Server agents; view status; and view the log files.

Chapter 8, “Configuring SonicMQ Adapters”

Describes how to establish and modify property and configuration settings for a SonicMQ Adapter; create or delete a SonicMQ Adapter instance; start and stop a SonicMQ Adapter; view status; and view the log file.

Chapter 9, “Configuring OE Web Servers”

Describes how to configure and manage OE Web Servers, REST Web services; manage REST Web Services instances; and work with OE WebServers.

Chapter A, “Third Party Acknowledgements”

Using this manual

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 see the OpenEdge Product Documentation Overview page on PSDN:

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.”
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 manual uses the following typographical conventions:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Bold</strong></td>
<td>Bold typeface indicates commands or characters the user types, provides emphasis, or the names of user interface elements.</td>
</tr>
<tr>
<td><em>Italic</em></td>
<td>Italic typeface indicates the title of a document, or signifies new terms.</td>
</tr>
<tr>
<td><strong>SMALL, BOLD CAPITAL LETTERS</strong></td>
<td>Small, bold capital letters indicate OpenEdge key functions and generic keyboard keys; for example, <code>GET</code> and <code>CTRL</code>.</td>
</tr>
<tr>
<td><strong>KEY1+KEY2</strong></td>
<td>A plus sign between key names indicates a <strong>simultaneous</strong> key sequence: you press and hold down the first key while pressing the second key. For example, <code>CTRL+X</code>.</td>
</tr>
<tr>
<td><strong>KEY1 KEY2</strong></td>
<td>A space between key names indicates a <strong>sequential</strong> key sequence: you press and release the first key, then press another key. For example, <code>ESCAPE H</code>.</td>
</tr>
</tbody>
</table>

Syntax:
<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fixed width</td>
<td>A fixed-width font is used in syntax, code examples, system output, and filenames.</td>
</tr>
<tr>
<td>Fixed-width italics</td>
<td>Fixed-width italics indicate variables in syntax.</td>
</tr>
<tr>
<td>Fixed-width bold</td>
<td>Fixed-width bold italic indicates variables in syntax with special emphasis.</td>
</tr>
<tr>
<td>UPPERCASE fixed width</td>
<td>ABL keywords in syntax and code examples are almost always shown in uppercase. Although shown in uppercase, you can type ABL keywords in either uppercase or lowercase in a procedure or class.</td>
</tr>
<tr>
<td>Period . or colon :</td>
<td>All statements except <strong>DO</strong>, <strong>FOR</strong>, <strong>FUNCTION</strong>, <strong>PROCEDURE</strong>, and <strong>REPEAT</strong> end with a period. <strong>DO</strong>, <strong>FOR</strong>, <strong>FUNCTION</strong>, <strong>PROCEDURE</strong>, and <strong>REPEAT</strong> statements can end with either a period or a colon.</td>
</tr>
<tr>
<td>[ ]</td>
<td>Large brackets indicate the items within them are optional.</td>
</tr>
<tr>
<td>[ ]</td>
<td>Small brackets are part of ABL.</td>
</tr>
<tr>
<td>{ }</td>
<td>Large braces indicate the items within them are required. They are used to simplify complex syntax diagrams.</td>
</tr>
<tr>
<td>{ }</td>
<td>Small braces are part of ABL. For example, a called external procedure must use braces when referencing arguments passed by a calling procedure.</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
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.
Introduction

OpenEdge® Management is a system management center that provides visibility, analysis, and proactive monitoring of critical information assets. OpenEdge Management optimizes the availability and performance of OpenEdge-based applications through system monitoring, alerting, and automatic handling of corrective actions. OpenEdge Management empowers Progress Software customers to become more efficient, decrease the cost of managing the OpenEdge® environment, and ensure high availability and performance.

OpenEdge Explorer is a subset of OpenEdge Management and runs within the OpenEdge Management console in a Web browser. OpenEdge Explorer enables you to set configuration properties, to start or stop, and to view the status or log files for various OpenEdge resources. OpenEdge Explorer is supported on all the UNIX platforms that support OpenEdge.

This chapter introduces:

- OpenEdge Management and OpenEdge Explorer
- Getting started
OpenEdge Management and OpenEdge Explorer

You can use OpenEdge Management or OpenEdge Explorer to configure your licensed OpenEdge resources.

Note: The AdminServer referenced in the following sections is a background process that provides a common point of management for the resources managed by OpenEdge Management or OpenEdge Explorer. For more information on the AdminServer, see OpenEdge Getting Started: Installation and Configuration.

OpenEdge Management

OpenEdge Management provides database administrators and systems operations managers with the performance tools and processes required to configure, monitor, diagnose, and manage the OpenEdge environment. This is all done from an easy-to-use graphical interface known as the management console, which runs in a Web browser.

Using OpenEdge Management, you can monitor the following resources running under a local AdminServer or under an AdminServer on a remote machine:

- OpenEdge databases (including those that are enabled for OpenEdge® Replication)
- System resources (CPU, disk, memory, file system)
- File resources
- OpenEdge servers, such as the AppServer, NameServer, WebSpeed® Transaction Server, and supported DataServers (ODBC, Oracle, and MS SQL Server)
- WebSpeed Messengers
- AppServer Internet Adapters, Web Services Adapters (for OpenEdge SOAP Web Services), OE Web Servers, and SonicMQ® Adapters
- TCP-based network services

OpenEdge Management’s deep level of monitoring provides information and details about your environment, enabling you to proactively manage operations.

In addition, OpenEdge Management includes the Database Administration Console, which allows you to manage and work with databases.
OpenEdge Explorer

OpenEdge Explorer allows you to set and modify properties from within the OpenEdge Management console. The functionality that you can use for the various resources in OpenEdge Explorer is provided in Table 1.

Table 1: OpenEdge Explorer functionality by resource

<table>
<thead>
<tr>
<th>OpenEdge Resource instance</th>
<th>Set config. props</th>
<th>Edit config. props</th>
<th>Start</th>
<th>Stop</th>
<th>Delete</th>
<th>Create</th>
<th>View generic status</th>
<th>View log file</th>
</tr>
</thead>
<tbody>
<tr>
<td>AppServer</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>AppServer Internet Adapter</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Database</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Messengers</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>MSS DataServer</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>NameServer</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>ODBC DataServer</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Oracle DataServer</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>SonicMQ Adapter</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>WebSpeed Transaction Server</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Web Services Adapter (for OpenEdge SOAP Web Services)</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>OE Web Server</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

OpenEdge Explorer also includes the Database Administration Console, which allows you to manage and work with databases.
Getting started

This guide provides details about how to use OpenEdge Management and OpenEdge Explorer to configure OpenEdge resources.

For introductory information about using the OpenEdge Management and OpenEdge Explorer console, see *OpenEdge Management and OpenEdge Explorer: Getting Started*.

Comparing OpenEdge Management and OpenEdge Explorer

OpenEdge Management and OpenEdge Explorer are packaged with OpenEdge and no longer require a separate installation; however, OpenEdge Management still requires a separate license. When OpenEdge Management is licensed, the OpenEdge Explorer functionality is included. If you did not purchase OpenEdge Management, however, its options are grayed out and unavailable in the management console when you are using OpenEdge Explorer.

In general, the settings you can establish and the options you can access when using OpenEdge Management and OpenEdge Explorer are determined both by the platform you are using and the products that you are licensed to use. If you notice that a particular link is dimmed in the OpenEdge Explorer console and not available for modification, it is because that link has no meaning within your operating environment or your licensed configuration.

**Note:** The settings (properties) for most of the resources managed by OpenEdge Management and OpenEdge Explorer (except for OpenEdge databases) are stored in the `ubroker.properties` file installed with OpenEdge. These properties are stored as name-value pairs. The names of these properties shown in OpenEdge Management and OpenEdge Explorer are long forms of the short names for the same properties stored in the `ubroker.properties` file. For more information about this file, see *OpenEdge Getting Started: Installation and Configuration*. For information on property storage for OpenEdge databases, see the "Database configuration and administration" section on page 28.
Feature comparison

Table 2 provides a comparison between the functionality available to you if you have purchased and installed OpenEdge Management (with OpenEdge Explorer), and the functionality available to you if you are working solely with OpenEdge Explorer.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Available in OpenEdge Explorer</th>
<th>Available in OpenEdge Management (with OpenEdge Explorer)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuring or modifying default property settings</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Starting/stopping resource instances (if applicable)</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Viewing status (if applicable)</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Viewing log files</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Adding and configuring new resource instances</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Supporting resource and log file monitoring for:</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>• Databases</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• DataServers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• AppServers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• NameServers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• WebSpeed Transaction Servers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Web Speed messengers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Network components (TCP, UDP, PING, HTTP)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• System components (CPU, disk, file system, memory)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• File components</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• SonicMQ Adapters</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Web Services Adapters (for OpenEdge SOAP Web Services)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• OE Web Servers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Supporting log file monitoring for:</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>• AppServer Internet Adapters</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• WebSpeed Messengers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Creating jobs for supported resources</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Creating reports for supported resources</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>
## Table 2: OEM and OEE feature comparison

<table>
<thead>
<tr>
<th>Feature</th>
<th>Available in OpenEdge Explorer</th>
<th>Available in OpenEdge Management (with OpenEdge Explorer)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using Library functions, such as rule sets, for supported resources</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Alerts for supported resources</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Scheduling and polling for supported resources</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Graphing for supported resources</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Trending for supported resources</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>SNMP support</td>
<td>No</td>
<td>Yes (with SNMP license)</td>
</tr>
<tr>
<td>The Dashboard</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>My Collections</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>
You can use OpenEdge Management and OpenEdge Explorer to configure and administer the database configuration of an existing OpenEdge RDBMS.

This chapter provides the following information about database configuration:

- Database configuration and administration
- Managing databases
- Viewing or modifying database properties
- Viewing the various database configuration properties
- Viewing the server group
- Starting or stopping the database
- Using the background writers and watchdog processes
- Viewing database status
- Deleting a database, database configuration, or database server group
- Configuring an OpenEdge Replication-enabled database

For details about managing and working with databases enabled for multi-tenancy, see *OpenEdge Management and OpenEdge Explorer: Configuring Multi-tenancy.*
Database configuration and administration

Using OpenEdge Management or OpenEdge Explorer, you can set properties for a database, its various configurations, and its associated server groups. Any configuration settings that you establish for a managed database, database configuration, or database server group in OpenEdge Management or OpenEdge Explorer are automatically reflected in the OpenEdge database Configuration Manager property file (conmgr.properties), and vice versa.

**Note:** You cannot create a new physical database from within OpenEdge Management or OpenEdge Explorer. For more information about creating a database, see *OpenEdge Data Management: Database Administration*.

A database recognized and managed by the AdminServer is called a *managed database*. A database not managed by an AdminServer is called a *scripted database*. Scripted databases are administered outside the AdminServer using parameter files (.pf) and operating system-dependent scripts.

A managed OpenEdge database starts up by using the default configuration information stored in the conmgr.properties file. The file contains the property settings for all managed databases on the host where OpenEdge is installed.

The conmgr.properties file resides in the properties subdirectory of the OpenEdge install directory. Each database is represented in the file by the following groups:

- **Database** — The group that describes the physical database. There is one database group for each physical database.

- **Configuration** — One or more groups that describe the database arguments and server groups for an active instance of the database. Only one database configuration can be active at one time. The active configuration is the default configuration identified in the server group.

- **ServerGroup** — One or more groups that describe brokering behavior. The first server group within a configuration defines the primary broker and each subsequent group defines a secondary broker. The properties in a server group define the arguments that are specific to that primary or secondary broker.

You can edit the conmgr.properties file manually; any changes that you make to it are automatically reflected in OpenEdge Management or OpenEdge Explorer.

**Note:** Although you can make manual edits to the conmgr.properties file, Progress Software recommends that you use the mergeprop utility, OpenEdge Management, or OpenEdge Explorer to make property changes. For more information about the mergeprop utility, see *OpenEdge Getting Started: Installation and Configuration*.
From OpenEdge Management or OpenEdge Explorer, you can perform the following database-related actions:

- Add an existing database
- Create a new database configuration
- Edit an existing or new database configuration
- Use the default server group
- Create a new database server group
- Edit a new or existing database server group
- Start a database
- Start a database automatically
- Stop a database
- Use the background writers and watchdog processes
- View the status of a database
- Delete a database
- Delete a database configuration (provided there is more than one)
- Delete a database server group (provided there is more than one)
- Configure the properties of an OpenEdge Replication-enabled database

You can also manage and work with databases enabled for multi-tenancy. For details, see OpenEdge Management and OpenEdge Explorer: Configuring Multi-tenancy.

**Viewing the database log file**

You can use the log file viewer to review the database log file. For more information, see the section about using the log file viewer in OpenEdge Management: Database Management.

**Database configuration awareness**

You can use OpenEdge Management or OpenEdge Explorer to establish database settings and options. These settings and options are determined both by the platform you are using and the products that you are licensed to use. If a property is not applicable to the database, its setting is either ignored or, when appropriate, an error or warning is issued.
Managing databases

You can use OpenEdge Management or OpenEdge Explorer to manage your database. You cannot physically create a database in the management console; however, you can create the definition of a database so that you can configure, start, stop, or view status for the primary broker and all auxiliary processes associated with that broker, and view status for the log file for the database.

Database configuration information is stored in the conmgr.properties file. For a complete list of how the database properties map to the command-line .pf arguments, see the conmgr.properties.README. You can locate the README file in the OpenEdge-install\properties directory.

A database recognized and managed by the AdminServer is called a managed database. A database not managed by an AdminServer is called a scripted database. Scripted databases are administered outside the AdminServer using parameter files (.pf) and operating system-dependent scripts. You can migrate a scripted database to a managed database.

For details about adding a managed or a scripted database, see OpenEdge Management: Database Management.

For details about managing and working with databases enabled for multi-tenancy, see OpenEdge Management and OpenEdge Explorer: Configuring Multi-tenancy.
Viewing or modifying database properties

Each database definition has a set of properties. You can view and modify the properties if you want to establish new values.

To view or modify the database properties:

1. Click Resources from the OpenEdge Management console menu. The Resources page appears with all the resources managed by your console, in a grid list.

2. From the resources list, select a database whose default configuration properties you want to modify.

   Note: You can also search for a specific database by entering the database name in the Search field or selecting Database from the resources type drop-down.

3. Click the Edit icon. The Database details page appears.

4. In the Command and control section of the page, click Configuration. The Database Configuration page appears.

   From the Database Configuration page, you can do the following:

   • View a read-only display of the database properties as described in Table 3.
   • Click Edit to modify the database properties. When you finish, click Save.
   • Click the relevant Configuration and Server Group link to access the database default configuration properties or the database server group properties. For details about these properties, see the “Viewing the various database configuration properties” section on page 33 and the “Viewing the server group” section on page 48.
   • Set OpenEdge Replication Server or Agent properties, if the database is replication-enabled or if an associated <db-name>.repl.properties file exists in the physical database directory. For details, see the “Configuring an OpenEdge Replication-enabled database” section on page 60.

Table 3: Database properties (1 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database name</td>
<td>Identifies the full path to the database.</td>
</tr>
<tr>
<td>Autostart</td>
<td>Indicates whether the database starts automatically when the AdminServer starts.</td>
</tr>
</tbody>
</table>
Table 3: Database properties (2 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster enabled</td>
<td>Indicates whether you have already configured the database as cluster-enabled (by using the PROUTIL DESCRIBE command). This allows the database to register with the cluster manager at start time.</td>
</tr>
<tr>
<td>Default configuration</td>
<td>Provides the name of the default configuration that is used when the database starts.</td>
</tr>
</tbody>
</table>
Viewing the various database configuration properties

OpenEdge Management and OpenEdge Explorer provide a predefined default database configuration. Once you add a database instance, you can use the property settings established in the default configuration, you can modify the default settings, or you can create and modify additional configurations.

Note that only one database default configuration can be selected for a database at a time.

To view the database default configuration properties:

1. Click Resources from the OpenEdge Management console menu. The Resources page appears with all the resources managed by your console, in a grid list.

2. From the resources list, select a database whose default configuration properties you want to view.

   Note: You can also search for a specific database by entering the database name in the Search field or selecting Database from the resources type drop-down.

3. Click the Edit icon. The Database details page appears.

4. In the Command and control section of the page, click Configuration. The Database Configuration page appears.

5. Click configuration.<databasename>.defaultconfiguration. The Database Configuration page appears, displaying the default configuration properties in the following categories: General, Background Writers, After Image Management, International, Advanced, Statistics Collection Configuration, SQL-92 Configuration, SSL Configuration, and Agent. These properties are described in Table 4 through Table 12, which follow this procedure.

   From the Database Configuration page, you can do the following:

   - View a read-only display of the database default configuration properties as described in Table 4 through Table 12.
   - Click Edit to modify the default database properties. For details, see the "Editing the database default configuration" section on page 46.
   - Click the relevant link to access the associated database server group properties. For details about these properties, see the "Viewing the server group" section on page 48.
### Table 4: General database configuration properties (1 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Blocks in database buffers (primary)</strong></td>
<td>The number of blocks in the database buffers. The optimum value for this property depends on your application.</td>
</tr>
<tr>
<td></td>
<td>The default value is 0.</td>
</tr>
<tr>
<td><strong>Alternate buffer pool</strong></td>
<td>The number of blocks in the alternate buffer pool. The optimum value depends on your application.</td>
</tr>
<tr>
<td></td>
<td>If you specify Alternate buffer pool blocks for a Replication target database, the property setting is ignored.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Alternate buffer pool functionality requires an Enterprise database license. For more information about alternate buffer pools, see OpenEdge Data Management: Database Administration.</td>
</tr>
<tr>
<td><strong>Hash table entries</strong></td>
<td>The number of hash table entries to use for the buffer pool.</td>
</tr>
<tr>
<td></td>
<td>The default value is 0.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Do not edit this property unless directed to do so by Progress Software Technical Support.</td>
</tr>
<tr>
<td><strong>Lock table entries</strong></td>
<td>The number of entries in the record locking table.</td>
</tr>
<tr>
<td></td>
<td>If you specify a value that is not a multiple of 32, the value is rounded to the next highest multiple of 32. Each record that is accessed by any user takes one entry.</td>
</tr>
<tr>
<td></td>
<td>The default value is 8192.</td>
</tr>
<tr>
<td><strong>Tenant Lock Table Governors(%)</strong></td>
<td>Specifies the percentage of the lock table that any tenant can occupy. The valid values are 1 through 100. The default value is 100.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> This parameter applies to only multi-tenant databases. In addition, this applies to only the default tenant and regular tenant. The super tenants do not use the governors parameters.</td>
</tr>
<tr>
<td></td>
<td>You can also monitor this parameter using PROMON utility.</td>
</tr>
<tr>
<td><strong>Max users</strong></td>
<td>The maximum number of users on the system for the database.</td>
</tr>
<tr>
<td></td>
<td>After the maximum number of users have connected to the database, additional user connect attempts are rejected.</td>
</tr>
<tr>
<td></td>
<td>The default value is 20.</td>
</tr>
</tbody>
</table>
### Table 4: General database configuration properties (2 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Tenant Max. User Governor** | Specifies the maximum number of users that may login for any tenant. The valid values are 1 through the maximum number of users (32000). The default value is the maximum number of users (-n).

**Note:** This parameter applies to only multi-tenant databases. In addition, this applies to only the default tenant and regular tenant. The super tenants do not use the governors parameters. You can also monitor this parameter using the PROMON utility. |
| **Max servers** | The maximum number of remote client servers that can be started by the broker process. Use the **Max servers** property to limit the number of remote user servers that can be started on the system. The performance trade-off to consider is swapping overhead for many servers versus overloading (slowing down) a server with too many clients.

**Note:** This parameter pertains only to the OpenEdge® Workgroup RDBMS and the OpenEdge® Enterprise RDBMS. In addition, this parameter applies only to databases that are accessed from remote network nodes. |
| **Maximum private database buffers per user** | Controls the number of private buffers a connection can request through the Private Read-only Buffers (-Bp) parameter. |
| **Direct I/O** | Enables the Direct I/O property, which opens all files in unbuffered mode. Direct I/O enables the database to use an I/O technique that bypasses the operating system buffer pool and transfers data directly from a buffer to disk.

This technique has some advantages over buffered reads and writes, such as avoiding the overhead of maintaining the operating system buffer pool, and eliminating competition for operating system buffers between the database and other programs. The operating system buffer-pool algorithms are designed for efficient sequential file access; the OpenEdge buffer-pool algorithms are more efficient for access to an OpenEdge RDBMS.

The **Direct I/O** check box is cleared by default. |
<p>| <strong>Other arguments</strong> | Any other arguments for which there is no defined property available. |</p>
<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Watchdog process</td>
<td>Start the Watchdog process automatically when the database starts. The check box is selected by default.</td>
</tr>
<tr>
<td>Asynchronous page writers</td>
<td>The number of Asynchronous Page Writer (APW) processes to start.</td>
</tr>
<tr>
<td></td>
<td>A database can have between zero and nine APWs running simultaneously. The optimal number is dependent on your application and environment.</td>
</tr>
<tr>
<td></td>
<td>To start, use one page writer for each disk where the database resides. If this is insufficient, add more.</td>
</tr>
<tr>
<td></td>
<td>For an application that performs many updates, start one APW for each disk containing your database, and add one additional APW. Applications that perform fewer changes to a database require fewer APWs. If an application performs no updates, no page writers are required.</td>
</tr>
<tr>
<td></td>
<td>The default value is 1.</td>
</tr>
<tr>
<td>Before image process</td>
<td>Start a Before-imaging process automatically when the database starts. The check box is selected by default.</td>
</tr>
<tr>
<td>Before image stall</td>
<td>Send a message to the log file when the recovery log threshold is reached, preventing emergency shutdown. (For details about the -bistall Startup parameter, see OpenEdge Data Management: Database Administration.)</td>
</tr>
<tr>
<td></td>
<td>The check box is cleared by default.</td>
</tr>
<tr>
<td>Before image buffers</td>
<td>The number of before-image buffers to use.</td>
</tr>
<tr>
<td></td>
<td>This property is useful only when running the before-image writer (BIW). The BIW continually writes the filled before-image buffers to disk, making the buffers available to other client and server processes. Without a BIW writing the buffers, any gain from increasing the number of buffers is negligible.</td>
</tr>
<tr>
<td></td>
<td>The default value is 5.</td>
</tr>
</tbody>
</table>
### Table 5: Background writers database configuration properties (2 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Before image threshold</strong></td>
<td>The maximum size to which BI files can grow as a percentage of the largest possible recovery log file. (For details about the -bithold startup parameter, see <em>OpenEdge Data Management: Database Administration.</em>) The default value is 3.</td>
</tr>
<tr>
<td><strong>Before image delay writes</strong></td>
<td>A positive value that you specify to delay OpenEdge from synchronously writing out to disk the last before-image (BI) file records at the end of each transaction. On UNIX systems using shared memory, the value also specifies the interval at which the broker process wakes up to make sure all BI file changes have been written to disk. The default value is 3.</td>
</tr>
<tr>
<td><strong>Before image cluster age</strong></td>
<td>The number of seconds before OpenEdge reuses a before-image cluster. The default value is 60.</td>
</tr>
<tr>
<td><strong>Group delay</strong></td>
<td>The number of milliseconds a transaction waits before committing.</td>
</tr>
<tr>
<td><strong>After image process</strong></td>
<td>Start an After-imaging process automatically when the database starts. The check box is cleared by default.</td>
</tr>
<tr>
<td><strong>After image stall</strong></td>
<td>Suspend database activity if all the after-image (AI) files are filled. By default, if all the AI files are filled and the database cannot switch to an empty one, the database shuts down. With AI stall, you can suspend database activity, back up the full after-image file, and mark it as empty. The check box is cleared by default.</td>
</tr>
<tr>
<td><strong>After image buffers</strong></td>
<td>The number of after-image buffers to use. This parameter is useful only when running the after-image writer (AIW) process because the AIW writes the filled after-image buffers to disk, making the buffers available to other client and server processes. The default value is 5.</td>
</tr>
</tbody>
</table>
### Table 6: After Image Management database configuration properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>After-image archival directory list</td>
<td>The directories where the AI File Management utility writes the archived after-image files.</td>
</tr>
<tr>
<td>Create after-image archival directories</td>
<td>Directs the AI File Management utility to create the specified directories.</td>
</tr>
<tr>
<td>After-image archival interval</td>
<td>Specifies on-demand mode archiving or the extent switch interval for timed-mode archiving. The minimum value is 120, and the maximum value is 86400. The default value is 120.</td>
</tr>
</tbody>
</table>

### Table 7: International database configuration properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collation table (-cpcoll)</td>
<td>A collation table in the convmap.cp file that you use for collation rules.</td>
</tr>
<tr>
<td></td>
<td>The collation table you specify is used with the code page OpenEdge uses in memory, which is specified by the Internal Code Page (-cpinternal) parameter.</td>
</tr>
<tr>
<td></td>
<td>The default value is BASIC.</td>
</tr>
<tr>
<td>Case table name (-cpcase)</td>
<td>A case table in the convmap.cp file that you use for case rules.</td>
</tr>
<tr>
<td></td>
<td>Case rules are used by the 4GL CAPS and LC functions and by the exclamation point (!) formatting character, which you use to tell OpenEdge to convert all characters to uppercase during input. The case table you specify is used to establish case rules for the code page used in OpenEdge's memory. The memory code page is set with the Internal Code Page (-cpinternal) parameter.</td>
</tr>
<tr>
<td></td>
<td>The default value is BASIC.</td>
</tr>
<tr>
<td>Conversion map (-convmap)</td>
<td>The conversion map file that OpenEdge uses for code page conversions, collation orders, and case conversions.</td>
</tr>
<tr>
<td></td>
<td>By default, OpenEdge uses OpenEdge-install-dir\convmap.cp in Windows and OpenEdge-install-dir/convmap.cp on UNIX.</td>
</tr>
</tbody>
</table>
Viewing the various database configuration properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Server code page (-cpinternal)** | The code page that OpenEdge uses in memory.  
For GUI clients, the `-cpinternal` code page should be the same code page the operating system uses.  
If you do not specify a value, OpenEdge uses the iso8859-1 code page by default. |
| **Log code page (-cplog)**    | The code page that the database uses when it writes to a log file. By default, the database uses the internal code page.  
Choose a code page that agrees with your operating system and its devices. If you have a Unicode database, you might want to specify the log character set as Unicode so information is properly logged.  
If you do not specify a value, OpenEdge uses the iso8859-1 code page by default. |
Table 8: Advanced database configuration properties  

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Database service communication area size</strong></td>
<td>The value (which corresponds to the \texttt{-pica} command-line option) that sets the size of the database service communications area. This area is used for storage of after-image block write notifications intended for OpenEdge Replication. The minimum value is 4 and the maximum value is 8192; the value is entered in Kb units.</td>
</tr>
<tr>
<td><strong>Windows Event Level</strong></td>
<td>The level of information written to the Windows Application Event Log. Choose from the following options:</td>
</tr>
<tr>
<td></td>
<td>• <strong>None</strong> — No events are written to the Event Log.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Brief</strong> — Error and Warning messages are written to the Event Log.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Normal</strong> — Error and Warning messages are written to the Event Log along with any message that is normally written to the log file (\texttt{.lg}). This is the default value.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Full</strong> — Error, Warning, and Informational messages are written to the Event Log along with any messages generated by the Message Statement.</td>
</tr>
<tr>
<td><strong>Minimum heap size</strong></td>
<td>The minimum value is 0, and the maximum value is 65535. The minimum size for temporary storage.</td>
</tr>
<tr>
<td><strong>Lock table hash table size</strong></td>
<td>The size of the hash table that controls access to the lock table.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Progress Software Corporation recommends changing the value of this property only after contacting Technical Support.</td>
</tr>
<tr>
<td><strong>Maximum number of database areas</strong></td>
<td>The highest area number available for use during the time the database is online.</td>
</tr>
<tr>
<td><strong>Nap interval</strong></td>
<td>The minimum nap time in the spin lock algorithm. The default value is 1.</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Nap time increment</strong></td>
<td>The nap increment value. (Enterprise database only)</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Do not edit this property unless directed to do so by Progress Software Technical Support.</td>
</tr>
<tr>
<td><strong>Nap max interval</strong></td>
<td>The maximum nap time in the spin lock algorithm before the nap time resets to the minimum.</td>
</tr>
<tr>
<td></td>
<td>The default value is 1.</td>
</tr>
<tr>
<td><strong>Nap time steps between nap time</strong></td>
<td>The steps between the nap increment. (Enterprise database only)</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Do not edit this property unless directed to do so by Progress Software Technical Support.</td>
</tr>
<tr>
<td><strong>Page writer max buffers</strong></td>
<td>The maximum number of modified database buffers to write to disk during a buffer scan cycle.</td>
</tr>
<tr>
<td></td>
<td>The default value is 25.</td>
</tr>
<tr>
<td><strong>Page writer queue delay</strong></td>
<td>The number of milliseconds between scans of the page writer queue.</td>
</tr>
<tr>
<td></td>
<td>The default value is 100.</td>
</tr>
<tr>
<td><strong>Page writer queue minimum</strong></td>
<td>The minimum number of buffers required on the page writer queue before an APW writes them to disk.</td>
</tr>
<tr>
<td></td>
<td>The default value is 1.</td>
</tr>
<tr>
<td><strong>Page writer scan</strong></td>
<td>The number of buffers that an APW should scan during a buffer scan cycle.</td>
</tr>
<tr>
<td></td>
<td>The default value is 1.</td>
</tr>
<tr>
<td><strong>Page writer scan delay</strong></td>
<td>The number of seconds of delay between each scan of the database buffers by the APW.</td>
</tr>
<tr>
<td></td>
<td>The default value is 1.</td>
</tr>
<tr>
<td><strong>Pending connection timeout</strong></td>
<td>The amount of time a client has to connect to a server before the broker clears the client's reservation.</td>
</tr>
</tbody>
</table>
Semaphore sets
The number of semaphore sets available to the database broker. On single-processor systems, OpenEdge uses semaphores to synchronize the activities of server and self-service client processes that are connected to a database. By default, each database has an array of semaphores, one for each user or server. Each process uses its semaphore when it must wait for a shared resource. Semaphores are not used for single-user sessions or for client sessions connecting to a remote database on a server system. By default, this value is set to 1. When more than 1,000 users connect to a single database, there might be high contention for the semaphore set. If there is a lot of semaphore contention on a system, using multiple semaphore sets helps alleviate this contention and improves performance on high user counts.

Shared memory overflow size
In kilobytes, the size of the shared-memory overflow area. This parameter replaces the default value of the shared-memory overflow area; it does not increase it. The overflow area is appended to the shared-memory area. The default value is 0.

Pin shared memory segments
Prevents the database engine from swapping shared memory contents to disk.

Maximum shared-memory segment size
The maximum number of bytes, specified in MB or GB, in a shared memory segment.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semaphore sets</td>
<td>The number of semaphore sets available to the database broker. On single-processor systems, OpenEdge uses semaphores to synchronize the activities of server and self-service client processes that are connected to a database. By default, each database has an array of semaphores, one for each user or server. Each process uses its semaphore when it must wait for a shared resource. Semaphores are not used for single-user sessions or for client sessions connecting to a remote database on a server system. By default, this value is set to 1. When more than 1,000 users connect to a single database, there might be high contention for the semaphore set. If there is a lot of semaphore contention on a system, using multiple semaphore sets helps alleviate this contention and improves performance on high user counts.</td>
</tr>
<tr>
<td>Shared memory overflow size</td>
<td>In kilobytes, the size of the shared-memory overflow area. This parameter replaces the default value of the shared-memory overflow area; it does not increase it. The overflow area is appended to the shared-memory area. The default value is 0.</td>
</tr>
<tr>
<td>Pin shared memory segments</td>
<td>Prevents the database engine from swapping shared memory contents to disk.</td>
</tr>
<tr>
<td>Maximum shared-memory segment size</td>
<td>The maximum number of bytes, specified in MB or GB, in a shared memory segment.</td>
</tr>
</tbody>
</table>
### Spin lock retries
The number of times a process tries to acquire a latch before pausing.

If the process cannot acquire the resource's latch, it continues the attempt. This iterative process is called spinning.

By default, this value is set to 0.

If the value of this property is greater than zero, a spin lock algorithm is used for shared-memory data structures. When a process has to lock a shared-memory structure, the process attempts to acquire the latch for that structure up to the specified number of times.

If the process has not acquired the latch in the specified number of attempts, then the process pauses, or naps. The length of the pause increases gradually if the process repeatedly fails to acquire a latch. After the allotted nap time, the process wakes up and attempts to acquire the latch again. If it fails to acquire the latch, it retries up to the number of attempts specified.

### Storage object cache size
The size of the object cache for all database objects.

### TXE commit lock skip limit
Sets the maximum SHARE/UPDATE locks that can skip the queued COMMIT locks.

**Note:** Do not edit this property unless directed to do so by Progress Software Technical Support.

### User MUX latches
A value that controls the granularity of access to large database resources in shared memory.

**Note:** Do not edit this property unless directed to do so by Progress Software Technical Support.
<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base index number</td>
<td>The starting index number in the range of indexes for which you want to track access statistics</td>
</tr>
<tr>
<td>Index range size</td>
<td>The number of indexes for which you want to track access statistics</td>
</tr>
<tr>
<td>Base table number</td>
<td>The starting table number in the range of tables for which you want to track access statistics</td>
</tr>
<tr>
<td>Table range size</td>
<td>The number of tables for which you want to track access statistics</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Java classpath</td>
<td>The pathname of the classpath</td>
</tr>
<tr>
<td>Number of open SQL cursors</td>
<td>The number of open cursors per connection</td>
</tr>
<tr>
<td>SQL stack size</td>
<td>The size, in MB, of the SQL Stack</td>
</tr>
<tr>
<td>SQL statement cache size</td>
<td>The number of statements allowed in the SQL statement cache</td>
</tr>
<tr>
<td>SQL sorting memory</td>
<td>The size of the temporary table buffer in memory</td>
</tr>
<tr>
<td>SQL sorting on disk</td>
<td>The size of the temporary table for backup storage</td>
</tr>
<tr>
<td>SQL temporary data page size</td>
<td>The size of the temporary table data page</td>
</tr>
<tr>
<td>Maximum number of JTA transactions</td>
<td>The number of JTA transactions simultaneously allowed</td>
</tr>
</tbody>
</table>
Table 11: SSL Configuration database properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable SSL for remote connections</td>
<td>Specifies that all connections to this server socket must use SSL.</td>
</tr>
<tr>
<td>Key alias name</td>
<td>Sets the alias name within the keystore of the private key and digital certificate entry to use to authenticate all connections to this server socket. If the key alias name is not specified, the server socket uses the default_server server certificate alias.</td>
</tr>
<tr>
<td>Key alias password</td>
<td>Sets the password to use for accessing the private key and digital certificate.</td>
</tr>
<tr>
<td></td>
<td>You must specify a password when you specify the -keyalias option, and the password must be encrypted. You must specify the password as an encrypted value that you can obtain using the genpassword utility located in the bin directory of your OpenEdge installation.</td>
</tr>
<tr>
<td></td>
<td>If you use the default_server server certificate, it also has a default password that you do not need to specify.</td>
</tr>
<tr>
<td>Disable SSL session cache</td>
<td>Disables session caching. (Session caching allows a client to reuse a previously established session if it reconnects prior to the session cache timeout expiring.)</td>
</tr>
</tbody>
</table>

Table 12: Agent database configuration property

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monitored</td>
<td>Starts the database monitoring agent automatically. This option is selected by default.</td>
</tr>
</tbody>
</table>
Chapter 2: Configuring OpenEdge Databases

Editing the database default configuration

OpenEdge Management and OpenEdge Explorer provide a default database configuration. You can modify the existing configuration and then either save it as a replacement for the default configuration or create a new configuration.

**Note:** When you create a new configuration, the configuration automatically uses the pre-defined OpenEdge defaults.

To edit a database default configuration:

1. Click **Resources** from the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the database whose default configuration you want to edit.
3. Click the Edit icon. The **Database** details page appears.
4. In the **Command and Control** section of the page, click **Configuration**. The **Database Configuration** page appears.
5. Click the **configuration.<databasename>.defaultconfiguration**. The **Database Configuration** page appears, with the default configuration properties organized into the categories described in Table 4 through Table 12.
6. Click **Edit** and make the modifications.
7. Click **Save**.

Creating a database configuration

OpenEdge Management and OpenEdge Explorer provide a predefined default database configuration; however, you can create a new configuration by modifying the default settings.

To create a database configuration:

1. Click **Resources** from the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the database whose default settings you want to modify to create a new configuration.
3. Click the Edit icon. The **Database** details page appears.
4. In the **Command and control** section of the page, click **Configuration**. The **Database Configuration** page appears.
5. Click **configuration.<databasename>.defaultconfiguration**. The **Database Configuration** page appears and displays the default configuration properties.
6. Click **Create**.
7. Type a new configuration group name in the field provided, and click Save. The Database Configuration page appears.

8. Click Edit to set the properties of the new database configuration. (Refer to Table 4 through Table 12 for a description of each property.)

9. When you finish, click Save.

To make this new configuration the default, you must select it as the default on the Database Configuration page.

To assign the new configuration to a database, see the “Assigning a different default database configuration” section on page 47.

**Assigning a different default database configuration**

OpenEdge Management and OpenEdge Explorer provide a default database configuration. You can modify that default configuration, or you can create one or more additional configurations and assign one of them as the default.

To assign a different default database configuration:

1. Click Resources from the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the database to which you want to assign a different default configuration.

3. Click the Edit icon. The Database details page appears.

4. In the Command and control section of the page, click Configuration. The Database Configuration page appears.

5. Click Edit.

6. From the Default configuration dropdown, select the default configuration you want to use.

7. Click Save.
Chapter 2: Configuring OpenEdge Databases

Viewing the server group

The server group specifies the logical collection of server processes to start.

You can apply properties to a database server group by either using the default configuration or creating and then specifying a configuration of your own choosing.

You can set or edit any of the database server group properties from OpenEdge Management or OpenEdge Explorer. The properties are organized into the General and Advanced categories.

To view the server group:

1. Click Resources from the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the database whose server group you want to view.

3. Click the Edit icon. The Database details page appears.

4. In the Command and control section of the page, click Configuration. The Database Configuration page appears.

5. Click servergroup.<databasename>.defaultconfiguration.defaultservergroup. The Database Configuration page for the server group appears.

From the Database Configuration page, you can do the following:

- View a read-only display of the General and Advanced database server group properties as described in Table 13 and Table 14, respectively.

- Click Edit to modify the database server group properties. For details, see the “Editing a database server group” section on page 51.

- Click Create to create or modify a server group. For details, see the “Creating a database server group” section on page 52.

- Click Delete to delete a server group, provided it is not running. For details, see the “Deleting a database server group” section on page 59.
Table 13: General database server group properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Service name or port** | The service name or port number through which clients may connect to the database.  
Notes: The service name or port number must represent a non-zero value in order to start a networked database broker.  
When you define multiple server groups for the same configuration, you must assign a unique service name or port number to each server group. |
| **Number of servers**  | The maximum number of server processes that can be started to service clients.  
The default value is 0.                                                                                                                      |
| **Message buffer size** | The standard message buffer size in bytes.  
OpenEdge uses message buffers to move records (messages) between servers and remote clients. Records (plus 40-byte headers) larger than the message buffer size are fragmented into multiple messages. If your database records are large, increase this parameter to avoid record fragmentation.  
However, if the network works more efficiently with small messages, reduce the **Message Buffer size** value and fragment larger records.  
The default value is set dynamically based on your platform (UNIX or Windows) and the type of database (Personal, Workgroup, or Enterprise). |
| **Reporting interval**  | The interval, in minutes, at which the server polls for license usage statistics that are reported in the usage report file.  
The default value is 1.                                                                                                                      |
Table 13: General database server group properties (2 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TCP/IP Version</strong></td>
<td>The TCP/IP version—IPv4 or IPv6—that the database broker uses to open the database port. The default value is IPv4.</td>
</tr>
<tr>
<td><strong>Client Type</strong></td>
<td>One of the following client types:</td>
</tr>
<tr>
<td></td>
<td>• <strong>4GL only</strong> — The server group can service requests only from ABL clients. The settings are the same as those for the <strong>Both 4GL and SQL</strong> option.</td>
</tr>
<tr>
<td></td>
<td>• <strong>SQL only</strong> — The server group can service requests only from SQL clients. The settings are the same as those for the <strong>Both 4GL and SQL</strong> option, except that the <strong>Message buffer size</strong> and <strong>Reporting interval</strong> settings are unavailable.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Both 4GL and SQL</strong> — The server group can service requests from both ABL and SQL clients.</td>
</tr>
</tbody>
</table>

**Note:** Remember that the client type choice you make here is honored only when you have installed the proper licenses for that client. For example, if you selected the **4GL only** option during the OpenEdge installation, choosing the **SQL only** option or the **Both 4GL and SQL** option in the management console has no effect. However, if you chose the **Both 4GL and SQL** option during the OpenEdge installation, choosing **SQL only** in the management console is valid.

Table 14: Advanced database server group properties (1 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Maximum clients per server</strong></td>
<td>The maximum number of remote users per database server.</td>
</tr>
<tr>
<td></td>
<td>The default value is 0.</td>
</tr>
<tr>
<td><strong>Minimum clients per server</strong></td>
<td>The minimum number of remote users on a server before the broker starts another server (up to the maximum number of servers).</td>
</tr>
<tr>
<td></td>
<td>The default value is 1.</td>
</tr>
<tr>
<td><strong>Maximum dynamic port</strong></td>
<td>The port number that is the highest in a specified range of server port numbers accessible to a client.</td>
</tr>
<tr>
<td></td>
<td>You specify the lowest port number with the <strong>Minimum dynamic port</strong> property.</td>
</tr>
<tr>
<td></td>
<td>The default value is set dynamically based on your platform (UNIX or Windows) and the type of database (Personal, Workgroup, or Enterprise).</td>
</tr>
</tbody>
</table>
Editing a database server group

For a database instance you added to OpenEdge Management or OpenEdge Explorer, you can edit any of the database server group properties.

To edit a database server group:

1. Click Resources from the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the database whose server group you want to edit.

3. Click the Edit icon. The Database details page appears.

4. In the Command and control section of the page, click Configuration. The Database Configuration page appears.

5. Click the servergroup.<databasename>.defaultconfiguration.defaultservergroup you want to edit. The Database Configuration page for the server group appears.

6. Click Edit and make the modifications. (For details about the properties in the General and Advanced categories, refer to Table 13 and Table 14, respectively.)

7. Click Save.

Table 14: Advanced database server group properties (2 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum dynamic port</td>
<td>The port number that is the lowest in a specified range of server port numbers accessible to a client. You specify the higher port number with the Maximum dynamic port property. Ports below 1025 are usually reserved for system services. The default value is set dynamically based on your platform (Unix or Windows) and the type of database (Personal, Workgroup, or Enterprise).</td>
</tr>
<tr>
<td>Host</td>
<td>The database’s host machine.</td>
</tr>
</tbody>
</table>
Creating a database server group

You can create a new database server group from OpenEdge Management or OpenEdge Explorer.

To create a database server group:

1. Click Resources from the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the database where you want to create a new server group.
3. Click the Edit icon. The Database details page appears.
4. In the Command and control section of the page, click Configuration. The Database Configuration page appears.
5. Click the servergroup.<databasename>.defaultconfiguration.defaultservergroup you want to edit. The Database Configuration page for the server group appears.
6. Click Create.
7. Type a new server group name in the field provided, and click Save. The Database Configuration page appears.

You can click Edit to set the properties of the new database server group. (For details about the properties in each category, refer to Table 13 and Table 14.) When you finish, click Save.
Starting or stopping the database

OpenEdge Management and OpenEdge Explorer support starting and stopping both local databases and databases running under a remote AdminServer on another machine. A local database resides on the same machine as OpenEdge Management or OpenEdge Explorer.

Before you can start or stop a database, you must first connect to the AdminServer that contains a definition of the database.

To start or stop a database, refer to the Starting or Stopping OpenEdge resources section in OpenEdge Management: Servers, DataServers, Messengers, and Adapters. The database agent works only if the database is running. Therefore, when you stop a database, the agent status automatically changes to Not Running. However, when you start a local database whose Agent Monitored property is disabled, the agent does not automatically start. You must specifically click Start Agent in the Database control page.

Notes: You can run only one configuration for a database at a time.

You can set the database configuration to start automatically when the AdminServer starts. For details, see the “Starting a database automatically” section on page 53.

Starting a database automatically

You can set a database to start automatically when the AdminServer starts.

To start a database automatically:

1. Click Resources from the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the database you want to start automatically.
3. Click the Edit icon. The Database details page appears.
4. In the Command and control section of the page, click Configuration. The Database Configuration page appears.
5. Click Edit.
6. Select the Autostart option by clicking in the check box.
7. Click Save.
Using the background writers and watchdog processes

On shared-memory systems, setting database background writer and watchdog processes can improve database performance. Background writers continuously perform overhead functions in the background.

Background writers

There are three types of background writers:

- **Asynchronous Page Writer (APW)** — The APW improves performance by performing overhead operations in the background. These operations provide available buffers, reduce the number of buffers that OpenEdge reads before writing to disk, and reduce the overhead associated with before-image checkpoints (the process of synchronizing the buffer pool of modified blocks to the database). You can start and stop an APW at any time without shutting down the database.

- **Before-image Writer (BIW)** — The BIW improves performance by continually writing before-image buffers to disk. These writes occur in the background. BIWs are optional; however, they are highly recommended for improving I/O performance. You can run only one BIW per database. You can start and stop a BIW at any time without shutting down the database.

- **After-image Writer (AIW)** — The AIW improves performance by continually writing after-image (AI) buffers to disk soon after OpenEdge fills the buffers. The after-imaging feature lets you recover a database that was damaged when a failure caused the loss of the database or primary recovery (BI) area. When you enable after-imaging, OpenEdge writes notes to the after-image files that contain a description of all changes to the database. You can run only one AIW process per database. You can start and stop an AIW at any time without shutting down the database.

The watchdog process

The watchdog process improves performance by cleaning up after improperly terminated processes. It releases locks, backs out any live transactions, and releases shared-memory locks. It also disconnects and cleans up the server’s remote clients.
Viewing database status

You can view the status of a database's configurations, server groups, and auxiliary processes through OpenEdge Management or OpenEdge Explorer. You can also view the status of the database monitoring agent.

Note: Most of the broker instances provide a status view, enabling you to monitor the status of multiple brokers simultaneously. You can monitor instances located on the same or different servers.

To view the status of a database:

1. Click Resources from the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the database whose status you want to view.
3. Click the Edit icon. The Database details page appears.
4. In the Command and control section of the page, click Control. The Database Control page appears.
5. Review the following status information for the database:
   - Database information — The database name, the active port or ports, and the name of the configuration being used by the database.
   - Database status — The status for the database and monitoring agent.
   - Database auxiliary status — The status of the After image page writer, the Before image page writer, the Auxiliary page writers, and the Watchdog process for this server group.
   - Properties — The status of the Monitor and Enabled settings: enabled (checked) or disabled.
Viewing database process details

You can access real-time details and statistics that provide you with snapshot information about a running database at the point you access this information from the Control page. You can review this information to help you assess the database’s performance.

To access processing details:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the database whose status you want to view.
3. Click the Edit icon. The Database details page appears.
4. In the Command and control section of the page, click Control. The Database Control page appears.
5. Click a PID link. The following sections appear:

- The Process summary section identifies the Process name and Process start time. User id and Group id values appear when UNIX-based data is shown. The Parent pid provides the identifier number associated with the process that spawned this current process.
- The Process statistics section presents details about the database’s real-time operational status. Values presented without parentheses identify that the processing time determined since the last scheduled polling interval, as noted, has occurred. Values presented within parentheses have been calculated based on information obtained since the start of the process.
Table 15 identifies and describes the fields of information presented in the Process statistics section.

Table 15: Process statistics data

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resident size</td>
<td>The physical size of the process as defined by the host system.</td>
</tr>
<tr>
<td>Virtual size</td>
<td>The virtual size of the process as defined by the host system.</td>
</tr>
<tr>
<td>CPU</td>
<td>The percentage of time spent using the CPU in either the user or kernel mode since the last scheduled poll.</td>
</tr>
<tr>
<td>Weighted CPU</td>
<td>The percentage of time spent using the CPU in either the user or kernel mode since the last scheduled poll divided by the number of CPU processors on the system. This value appears only when there is more than one CPU process on the system where the process is running.</td>
</tr>
<tr>
<td>User time</td>
<td>The amount of CPU time spent in the user mode since the last scheduled poll.</td>
</tr>
<tr>
<td>Kernel time</td>
<td>The amount of CPU time spent in the kernel mode since the last scheduled poll.</td>
</tr>
<tr>
<td>Process time</td>
<td>The sum of the values that appear in the User time and Kernel time fields.</td>
</tr>
</tbody>
</table>
Deleting a database, database configuration, or database server group

You can delete a database, provided it is not running, and you can delete a database configuration or database server group, provided it is not currently in use by a database.

Deleting a database

When you no longer need a particular database, you can delete the database and all associated configuration information from the management console. When you delete a database from the console, you are not deleting the actual, physical database files; you are only deleting the database definition from the `conmgr.properties` file. After you delete the database, it no longer appears in the console.

You cannot delete a database that is running. You must first stop the database and then delete it.

To delete a database from the management console:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the database whose status you want to delete.
3. Click the Edit icon. The Database details page appears.
4. Click Delete and then click OK to confirm the deletion.

Deleting a database configuration

You can delete a database configuration from the management console.

If a configuration is running, you cannot delete it. You must first stop the configuration, and then delete it.

To delete a database configuration from the management console:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the database you want to delete.
3. Click the Edit icon. The Database details page appears.
4. In the Command and control section of the page, click Configuration. The Database Configuration page appears.
5. In the Configuration and Server Group Links section of the page, click the database configuration you want to delete. The configuration information appears.
6. Click Delete, and then click OK to confirm the deletion.
Deleting a database server group

You can delete a database server group from the management console.

If the configuration associated with the server group is in use, you cannot delete it. You must first stop the configuration, and then delete the server group.

To delete a database server group:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the database whose server group you want to delete.
3. Click the Edit icon. The Database details page appears.
4. In the Command and control section of the page, click Configuration. The Database Configuration page appears.
5. In the Configuration and Server Group Links section of the page, click the server group you want to delete. The server group information appears.
6. Click Delete, and then click OK to confirm the deletion.
Chapter 2: Configuring OpenEdge Databases

Configuring an OpenEdge Replication-enabled database

To configure your source database and target database for OpenEdge Replication, you must configure the replication property files for both the OpenEdge Replication source database and the OpenEdge Replication target database.

In the `OpenEdge-install-dir/properties` folder, OpenEdge Replication provides two sample property files: `source.repl.properties` and `target.repl.properties`. You can save these files and modify them, using either OpenEdge Management or OpenEdge Explorer, or a text editor, so that you can use them with your source and target OpenEdge Replication databases. Because `source.repl.properties` and `target.repl.properties` are samples, some of the properties and values are generic; it is up to you to customize the properties to reflect the settings you want to use with your own source and target databases.

When you use OpenEdge Management or OpenEdge Explorer to make configuration changes to the Replication server or Replication target properties, the changes are reflected in the applicable property file.

Rules for setting properties for a Replication-enabled database

From the management console, you can configure various properties for a database enabled for Replication. The options you have in the console for configuring these Replication-related properties depend upon the following rules:

- For a database **not enabled** for Replication:
  - If the Replication property file **does not** exist, no Replication links are available.
  - If the Replication property file **does** exist but is empty, a **Server (create new)** link and an **Agent (create new)** link appear.

  **Note:** You can configure transition only when one of these groups exists.

  - If the Replication property file **does** exist and is **not** empty, then all groups defined in the property file appear.

- For a database **enabled** for Replication:
  - If the Replication property file **does not** exist, Replication-related links still appear. However, you can create only the group for which the database is enabled. In other words, you can create the Server group for source-enabled databases and the Agent group for target-enabled databases.
  - If the Replication property file **does** exist, only the groups that are applicable to the database enablement appear. For source enablement, **Server**, **ctrl-agents**, and **transition** appear. For target enablement, **Agent** and **transition** appear.
– If the Transition group database-role is set to reverse, you either see the existing group for the reverse role or are allowed to create a new group for the reverse role.

– If you specify Alternate buffer pool blocks for a Replication target database, the property setting is ignored.

### Setting Replication-related properties

If you are familiar with OpenEdge Replication and its configuration, see the following topics for details about the properties you can configure through the management console:

- Setting OpenEdge Replication Agent properties
- Setting OpenEdge Replication Control Agent properties
- Setting OpenEdge Replication Server properties
- Setting OpenEdge Replication Transition properties

If you are not familiar with OpenEdge Replication and its configuration, see OpenEdge Replication: User Guide for details about setting the properties.

### Setting OpenEdge Replication Agent properties

You set OpenEdge Replication Agent properties to define the configuration for the local agent running for the target database.

**Note:** If you are not familiar with OpenEdge Replication and its configuration, see OpenEdge Replication: User Guide.

To set the OpenEdge Replication Agent properties:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the database whose Replication Agent properties you want to set.

3. Click the Edit icon. The Database details page appears.

4. In the Command and control section of the page, click Configuration. The Database Configuration page appears.

5. In the OpenEdge Replication Group Links section, click Agent. The Database Replication Configuration page opens.
6. Set the following properties:

- **Agent name** — Specifies the OpenEdge Replication agent name. By convention, the agent name should match the name specified in the source database properties file.

  ALL is not allowed as an agent name. Each agent must have a unique name.

- **Database** — Specifies the source database name.

- **Connection Timeout** — Specifies how many seconds the OpenEdge Replication agent will wait for connection from the OpenEdge Replication server before the Replication agent shuts itself down.

  The seconds value must be greater than or equal to 120, and less than or equal to 86,400.

  Using this property means you do not have to do a forced shutdown on your target database. If the OpenEdge Replication agent does not receive a connection attempt from the OpenEdge Replication server before the number of seconds specified have elapsed, the OpenEdge Replication agent will terminate and allow some limited system-level target database connections.

- **Polling delay minimum** — Specifies the minimum value, in milliseconds, for a polling delay.

  By default, the polling delay starts at 5 milliseconds and automatically increases during periods of inactivity to a maximum of 500 milliseconds.

  The milliseconds value must be greater than or equal to one, and less than or equal to ten.

- **Polling delay maximum** — Specifies the maximum value, in milliseconds, for a polling delay.

  By default, the polling delay starts at 5 milliseconds and automatically increases during periods of inactivity to a maximum of 500 milliseconds.

  The milliseconds value must be greater than or equal to 500 and less than or equal to 1000.

- **Agent minimum port** — Specifies the minimum TCP port number. The agent selects a port in a range between the values specified by the minimum port and the maximum port.

  The minimum port number value must be greater than 1024 and less than the value of the maximum port.
• **Agent maximum port** — Specifies the maximum TCP port number for the agent.

  The maximum port number value must be greater than the minimum port + 1, and less than the maximum allowable port number on the system.

  For UNIX the maximum port number is 65534. For Windows the maximum port number is 32765.

• **Other arguments** — Any other arguments for which there is no defined property available.

7. Click **Save**.

### Setting OpenEdge Replication Server properties

You set the OpenEdge Replication Server properties to specify server and agent property values.

**Note:** If you are not familiar with OpenEdge Replication and its configuration, see [OpenEdge Replication: User Guide](#) before setting the properties.

The server properties are divided into the following categories:

- General
- Advanced

To set the OpenEdge Replication Server properties:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the database whose Replication Server properties you want to set.

3. Click the Edit icon. The **Database** details page appears.

4. In the **Command and control** section of the page, click **Configuration**. The **Database Configuration** page appears.

5. In the **OpenEdge Replication Group Links** section, click **Server**. The **Database Replication Configuration** page opens.
6. Set the following **General** properties:

- **Database** — Specifies the source database name.

- **Defer agent startup** — Specifies for how long, in minutes, the server attempts to connect to an agent if the first connection attempt is unsuccessful. The minutes value must be greater than or equal to zero and less than or equal to 10080.

- **Keep alive** — Specifies a timeout period for communications between a server and its agents. If a connection between the server and the agent is not verified before the timeout expires, failure recovery begins.

  By default, this property is enabled and has a value of 300 seconds. The minimum value is 60 seconds; there is no maximum value.

- **Server transition timeout** — Specifies the number of seconds the target database waits before it performs auto-transition. This property is ignored when transition is set to manual.

  The value is incremented by the sum of the connect-timeout for all configured agents.

- **Server transition** — Specifies how to transition the target database to a normal database.

  If a synchronous agent or an asynchronous critical agent cannot reconnect with the source database within the time specified in the transition-timeout property, the following occurs:

  - **auto** — The agent automatically transitions the target database.

  - **manual** — The agent listens for the server until it reconnects with the source database or the DSRUTIL command is executed. The `DSRUTIL db-name -C transition` command transitions the target database.

7. Set the following **Advanced** properties:

- **Other arguments** — Any other arguments for which there is no defined property available.

- **Schema lock action** — Specifies the action an agent takes if an exclusive schema lock is not granted. Possible actions are:

  - **wait** — The agent waits until the exclusive schema lock is granted. The server blocks until the exclusive schema lock is granted.

  - **force** — The agent attempts to acquire the exclusive schema lock five times. If the fifth attempt fails, the agent disconnects all users from the target and makes another attempt. If the last attempt fails, the server and all agents terminate. When schema update activity completes, the server and target can be restarted.
• **Polling delay minimum** — Specifies the minimum value, in milliseconds, for a polling delay. By default, the polling delay starts at 5 milliseconds and automatically increases during periods of inactivity to a maximum of 500 milliseconds.

    The value must be greater than or equal to one and less than or equal to ten.

• **Polling delay maximum** — Specifies the maximum value, in milliseconds, for a polling delay. By default, the polling delay starts at 5 milliseconds and automatically increases during periods of inactivity to a maximum of 500 milliseconds.

    The value must be greater than 500 and less than 1000.

• **Server port minimum** — The minimum port available.

• **Server port maximum** — The maximum port available.

• **Agent shutdown action** — Specifies the action an agent takes during a shutdown when the Replication server ends. Possible actions are:

    – **recovery** — The agent remains active but in a standby state waiting for the replication server to reconnect.

    – **normal** — The agent terminates; the target database stays up.

Then click **Save**.

### Setting OpenEdge Replication Control Agent properties

You set the OpenEdge Replication control agent properties to define for the server which agents it will contact, where it will contact them, and how the agent should perform. Each server group supports up to two control agents.

**Note:** If you are not familiar with OpenEdge Replication and its configuration, see *OpenEdge Replication: User Guide* before setting the properties.

#### To set the OpenEdge Replication Control Agent properties:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the database whose Replication Control Agent properties you want to set.

3. Click the Edit icon. The **Database** details page appears.

4. In the **Command and control** section of the page, click **Configuration**. The **Database Configuration** page appears.

5. In the **OpenEdge Replication Group Links** section, click **Control-Agent**. The **Database Replication Configuration** page opens.

6. Click **Edit**.
7. Set the properties as follows:

- **Database** — Specifies the source database name.

- **Host** — Specifies to the server which host the agent will start on.

- **Port** — Specifies which port the server uses to connect to the target database. The port number specified must be the same port specified with the `-s` parameter when the target database broker was started.

- **Connection Timeout** — Specifies how many seconds the OpenEdge Replication agent will wait for a connection from the OpenEdge Replication server before the Replication Agent shuts itself down.

  The seconds value must be greater than or equal to 120, and less than or equal to 86,400.

  Using this property means you do not have to do a forced shutdown on your target database. If the OpenEdge Replication agent does not receive a connection attempt from the OpenEdge Replication server before the number of seconds specified elapses, the OpenEdge Replication agent will terminate and allow some limited system-level target database connections.

  This property is also used by the server when reconnecting to the agent after communication has been lost.

- **Maximum message** — The maximum number of bytes used for the TCP/IP communication messages.

- **Replication method** — Asynchronous or synchronous.

- **Critical agent** — Specifies whether the agent is critical. (Select the check box for a critical agent.)

  A critical agent is an asynchronous agent for the target database that can become the source database if the source database becomes unavailable.

- **Other arguments** — Any other arguments for which there is no defined property available.

8. Click **Save**.
Setting OpenEdge Replication Transition properties

You set the Transition properties to specify the transition process values, which are used when the target database becomes a normal database after a failure.

**Note:** If you are not familiar with OpenEdge Replication and its configuration, see *OpenEdge Replication: User Guide* before setting the properties.

The transition properties are divided into the following categories:

- General
- Start Up
- After Image
- Back Up

To set the OpenEdge Replication Transition properties:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the database whose Replication Transition properties you want to set.

3. Click the **Edit** icon. The **Database** details page appears.

4. In the **Command and control** section of the page, click **Configuration**. The **Database Configuration** page appears.

5. In the **OpenEdge Replication Group Links** section, click **Transition**. The **Database Replication Configuration** page opens.

6. Set the following **General** properties:

- **Database role:**
  - **reverse** — The role of the database is reversed: A source database becomes a target database, and a target database becomes a source database.
  - **normal** — The role of the database becomes that of a normal database; the database is no longer enabled for replication once the transition is performed. This is the default value.

- **Responsibility:**
  - **primary** — This is the primary database.
  - **secondary** — This is the secondary database.
7. Set the following **Start Up** properties:

- **Transition to agents** — Transitions to the first agent in the list when a failure occurs. If the first agent is not available, transition to the second agent in the list.

  A valid value for this property is any configured agent name; separate the names by a comma if you are listing more than one. For example, `agent1,agent2`.

- **Restart after transition** — The database can be automatically restarted after transition is performed. When you select this option, the `*-startup-arguments` properties must be supplied, or the database startup will fail:

  - If the database role is normal, you must specify the `normal-startup-arguments`.

  - If the database role is reversed, you must specify the `source-startup-arguments` and the `target-startup-arguments`.

- **Source startup arguments** — If the database is transitioned to a source database, these arguments are used when the database is started. The arguments are appended to the PROSERVE command (used to start the database).

  In most cases, the only argument specified here should be `-pf` followed by a parameter file name; for example, `-pf db-name.source.pf`.

  Because the database is a source, you must also specify the `-DBService replserv` argument, as an indication to the broker to start the Replication server.

- **Target startup arguments** — If the database is transitioned to a target database, these arguments are used when the database is started. The arguments will be appended to the PROSERVE command.

  In most cases, the only argument specified here should be `-pf` followed by a parameter file name; for example, `-pf db-name.target.pf`.

  Because the database is a target, you must also specify the `-DBService replagent` arguments, as an indication to the broker to start the Replication agent and to listen on the TCP/IP port specified with `-S port-number` or `service-name`.

- **Normal startup arguments** — If the database is transitioned to a normal database, these arguments are used when the database is started. The arguments will be appended to the PROSERVE command.

  In most cases, the only argument specified here should be `-pf` followed by a parameter file name; for example, `-pf db-name.normal.pf`. 
8. Set the following After image properties:

   - **Begin after imaging after transition** — Select the option to automatically begin AI after a target-to-source database transition. After-imaging can be started for a database that has AI areas.
   
   - **Automatically add after imaging extents** — This property instructs the transition process to automatically add AI areas to the database if:
     
     - It is transitioned to a source database.
     
     - There are currently no AI areas for the database.
     
     If you do not select the option, AI areas are not added to the database. If you do select the option, AI areas are automatically added to the database by using the structure file specified in the `ai-structure-file` property.

   - **Structure file containing after image area definitions** — The name of the structure file that contains the list of AI areas to add.

9. Set the following Back Up properties:

   - **Backup method** — Specifies the backup method performed before AI is enabled. The following values are valid:
     
     - **mark** — Marks the database backed up by using the `rfutil db-name -C mark backedup` command. Marking the database as backed up does not allow future AI extents to be used when recovering from a disaster.
     
     - **full off-line** — Backs up the database offline by using the OpenEdge PROBKUP utility. The backup is performed in two steps. The first backup is a full backup, which is performed before AI is enabled for the database. The second backup is an incremental backup, which is performed after AI is enabled, and after the role of the database is changed.
     
     - **full on-line** — This backup is performed after the database is restarted, which occurs after the database has been transitioned.

   - **Full on-line and off-line backup arguments** — The arguments required for the full online and offline backups that are performed for the database. At a minimum, the target file or device must be specified in these arguments for both online and offline backups.
     
     To avoid overwriting a backup, do not use the same target file or device for both the backup and the incremental backup.
     
     Do not use backup validation parameters (such as `-vp` and `-vf`).
     
     The arguments should begin with `device-name`.

   - **Incremental backup arguments** — The arguments required for the offline incremental backup performed after AI is enabled and the database’s role is reversed.
     
     The arguments should begin with `device-name`.
You can configure DataServer property settings and add new DataServer instances in OpenEdge Management and OpenEdge Explorer. You can also start, stop, and delete DataServer instances, as well as view their status and log files.

This chapter provides the following information:

- ODBC DataServer configuration and administration
- Oracle DataServer configuration and administration
- MS SQL Server DataServer configuration and administration
ODBC DataServer configuration and administration

The OpenEdge® DataServer for ODBC allows the OpenEdge Application Development Environment (ADE) and applications created with OpenEdge in Windows to access the ODBC-compliant databases DB2 and Sybase.

You can use OpenEdge Management or OpenEdge Explorer to configure and administer the configuration of an OpenEdge DataServer for ODBC. The DataServer for ODBC installation provides one predefined DataServer broker (odbbroker1) and one predefined NameServer (NS1). You can use these predefined components as a starting point for creating and configuring additional DataServer brokers, and, if needed, NameServers. Each broker is referred to as an instance.

Working with the ODBC DataServer

To work with the ODBC DataServer, double-click the ODBC DataServer folder from the management console list frame. There is one pre-defined default ODBC DataServer broker (osbroker1).

The following sections describe these ODBC DataServer actions:

- Configuring an ODBC DataServer broker
- Editing an ODBC DataServer broker configuration
- Creating an ODBC DataServer broker
- Starting or Stopping an ODBC DataServer broker
- Starting an ODBC DataServer broker automatically
- Viewing the status of an ODBC DataServer broker
- Deleting an ODBC DataServer broker
- Viewing the status of an ODBC DataServer broker
Configuring an ODBC DataServer broker

You can view or modify the following default configuration properties of an ODBC DataServer broker from the management console:

- Broker
- Server
- Environment variables

You can also create a new ODBC DataServer broker, view a broker's status, delete a broker you no longer need, or view the broker's log files.

Each of these tasks is described in the following sections.

Viewing or modifying ODBC DataServer broker properties

You can view or modify ODBC DataServer Broker properties in the following categories: general, owner information, controlling NameServer, Data Service, logging setting, and advanced features.

To view or modify broker properties for an ODBC DataServer:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the ODBC DataServer broker whose properties you want to view or modify.
3. Click the ODBC DataServer broker that you want to edit. The ODBC DataServer details page appears.
4. In the Command and control section of the page, click Configuration. The ODBC DataServer Configuration page appears.

From this page, you can:

- View a read-only display of the ODBC broker properties as described in Table 16.

- Click Edit to modify the broker properties. For details, see the "Editing an ODBC DataServer broker configuration" section on page 79.

<table>
<thead>
<tr>
<th>Table 16: ODBC DataServer broker properties (1 of 4)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Property</strong></td>
</tr>
<tr>
<td><strong>General</strong></td>
</tr>
<tr>
<td>Working directory</td>
</tr>
<tr>
<td>Operating mode</td>
</tr>
</tbody>
</table>
The type of the TCP/IP listening port that the DataServer broker uses. The default value is 3050.

Starts the DataServer automatically when the AdminServer starts. The Auto start check box is cleared by default.

N/A

Your user name. The default is blank.

Your password. The default is blank.

A check box that indicates whether to register the DataServer with a controlling NameServer.

Identifies the NameServer that the DataServer uses to find a broker that supports the correct application services. (If you did not select to register the DataServer with a NameServer, the field is dimmed and unavailable.)

How the broker specifies its hostname when registering with its controlling NameServer. This hostname information is passed onto a client application when it attempts to connect to a Data Service that the broker supports. The choices are:

- **Register IP** — (Default) This setting is the most efficient mechanism, and can be used in most cases. It registers with the IP address of the machine where the broker is located.
- **Register-LocalHost** — The broker registers with the hostname of the machine that it runs on. Use this setting when the broker runs on a machine with a single hostname and more than one IP address.
- **Register-HostName** — The broker registers with the values specified in the hostName property. Use this setting when your clients need a fully qualified host name to connect to a broker in a different DNS domain.
<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registration host name</td>
<td>Identifies the hostname value to be used if the Registration mode property is set to Register-HostName.</td>
</tr>
<tr>
<td>Data service name list</td>
<td>Data services are services that are provided by a selected broker's servers. The default service name is the broker name. The default value is blank.</td>
</tr>
<tr>
<td>Supports default service</td>
<td>A check box that indicates if the ODBC broker supports the default service. Select the check box to have the broker support the default service, or click to clear the check box and cancel support for the default service. The Supports default service check box is cleared by default.</td>
</tr>
<tr>
<td>Broker log filename</td>
<td>The general broker log filename. The default is @{WorkPath}\oddbroker1.broker.log.</td>
</tr>
<tr>
<td>Broker logging level</td>
<td>The amount of information written to the broker log. The possible values are:</td>
</tr>
<tr>
<td></td>
<td>• None — No log file is written.</td>
</tr>
<tr>
<td></td>
<td>• Error Only — Only error-related information is written to the log file, which reduces the size of the log file.</td>
</tr>
<tr>
<td></td>
<td>• Basic — Only some information is written to the log file.</td>
</tr>
<tr>
<td></td>
<td>• Verbose — All broker activity is written to the log file. This might produce large files that use up disk space.</td>
</tr>
<tr>
<td></td>
<td>• Extended — Information in addition to Verbose.</td>
</tr>
<tr>
<td>Append to broker log file</td>
<td>Deselect to create a new broker log file each time a broker is started, even if the broker log file specified already exists. The Append to broker log file check box is selected by default.</td>
</tr>
<tr>
<td>Broker logging entry types</td>
<td>A single entry or comma delimited list of logging entry types.</td>
</tr>
<tr>
<td>Broker log file threshold size</td>
<td>Specifies a limit on how big the log file can get (in bytes). When the log is full, the broker creates a new log file with a sequence number.</td>
</tr>
</tbody>
</table>
### Advanced Features

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum number of broker log files</td>
<td>Limit on how many broker log files will be kept on the system when there is a threshold limit. It represents the number of log files including the current one and includes any rolled over log files that already exist when the process started.</td>
</tr>
<tr>
<td>Maximum client instances</td>
<td>The maximum number of client connections that the broker can support concurrently. The default value of 512 that is set for this parameter is high enough to ensure that the number of client connections is unlimited. Realistically, however, the system-level resources needed to support an unlimited number of clients might be exhausted before this limit is reached. If you see broker log entries that contain Exception Messages for &quot;out of space&quot; or &quot;OutOfMemory&quot; you might need to reduce the maximum number of client connections to a more reasonable value. Note, however, that lowering this value can cause some client requests to be rejected (&quot;Exceeded Max Clients&quot;). Configuring (and starting) multiple brokers to handle higher client loads alleviates the problem.</td>
</tr>
<tr>
<td>Priority weight (0-100) (dynamic property)</td>
<td>An integer value between 0 and 100 that influences the share of the workload that the selected broker receives. The larger the value, the heavier the load that is distributed to the broker. Using the Priority weight value, the NameServer distributes client requests to the same DataServer service across all DataServer brokers. The default value is 0. You can update this property dynamically. Any changes will affect all current and new brokers.</td>
</tr>
<tr>
<td>Registration retry (in seconds)</td>
<td>After registering with its controlling NameServer, the DataServer broker periodically sends &quot;keep-alive&quot; messages to the NameServer to let the NameServer know that the broker is still active. The Registration retry value is the number of seconds that pass between &quot;keep-alive&quot; messages. The default value is 30.</td>
</tr>
</tbody>
</table>
**Viewing or modifying ODBC DataServer broker server properties**

You can view or modify ODBC DataServer broker Server properties in the following categories: general and logging setting.

**To view or modify server properties for an ODBC broker:**

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the ODBC DataServer broker whose server properties you want to view or modify.

3. Click the ODBC DataServer broker that you want to edit. The **ODBC DataServer** details page appears.

4. In the **Command and control** section of the page, click **Configuration**. The **ODBC DataServer Configuration** page appears. From this page, you can:

   - View a read-only display of the ODBC server properties as described in Table 17.

   - Click **Edit** to modify the default server properties. For details, see the “Starting or Stopping an Oracle DataServer broker” section on page 94.

**Table 17: ODBC DataServer server properties (1 of 2)**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>General</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Server executable file</strong></td>
<td>The pathname of the DataServer executable that the broker runs. The default for the DataServer for ODBC in Windows is:</td>
</tr>
<tr>
<td>(dynamic property)</td>
<td><code>@{Startup\DLC}bin\odbsrv.exe</code></td>
</tr>
<tr>
<td></td>
<td>You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated.</td>
</tr>
<tr>
<td><strong>Server startup parameters</strong></td>
<td>The startup parameters for the server. You should not modify the default parameters, which are:</td>
</tr>
<tr>
<td>(dynamic property)</td>
<td><code>-svub -S -X -N TCP -U X -P x -hs 0 -s 40</code></td>
</tr>
<tr>
<td></td>
<td>You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated.</td>
</tr>
<tr>
<td><strong>PROPATH</strong></td>
<td>A list of directories OpenEdge searches to find procedures.</td>
</tr>
</tbody>
</table>
Setting ODBC DataServer broker environment variables

You can set environment variables for the DataServer broker for ODBC. You can set one or more environment variables that:

- Are accessible from your ABL application
- Affect the environment in which your application runs

For the ODBC DataServer, there are no required environment variables in Windows.

To set ODBC DataServer broker environment variables:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the ODBC DataServer broker whose environment variables you want to set.
3. Click the ODBC DataServer broker that you want to edit. The ODBC DataServer details page appears.
4. In the Command and control section of the page, click Configuration. The ODBC DataServer Configuration page appears.
5. Click Edit.
6. Click Environment Variables.

7. Enter each variable name and then provide its value in the format name=value.

8. Click Save when you finish adding the variables and their values.

**Editing an ODBC DataServer broker configuration**

You can edit an ODBC DataServer broker configuration from the management console.

To edit an ODBC DataServer broker configuration:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the ODBC DataServer broker whose configuration you want to edit.

3. Click the ODBC DataServer broker you want to edit. The ODBC DataServer details page appears.

4. In the Command and control section of the page, click Configuration. The ODBC DataServer Configuration page appears.

5. Click Edit. (For details about the broker and server properties, see Table 16 and Table 17, respectively.)

6. Make the configuration changes, and then click Save.

**Creating an ODBC DataServer broker**

OpenEdge Management and OpenEdge Explorer provide a predefined default ODBC DataServer configuration; however, you can create a new configuration by modifying the default settings.

To create an ODBC DataServer broker:

1. From the drop-down for Resources on the management console menu, click New OpenEdge Resource→ODBC DataServer. The ODBC DataServer Configuration page appears.

2. Type the name of the new ODBC DataServer broker in the field provided.

3. Click Save. The ODBC DataServer Configuration page appears, allowing you to establish the broker’s properties.
Deleting an ODBC DataServer broker

You can delete an inactive ODBC DataServer broker from the management console.

If a broker is running, you cannot delete it. (The Delete button is not available.) You must first stop the broker and then delete it.

To delete an ODBC DataServer broker:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the ODBC DataServer broker.
3. From the list of ODBC DataServer brokers, click the ODBC DataServer that you want to delete. The ODBC DataServer details page appears.
4. Click Delete, and then click OK to confirm the deletion.

Starting or Stopping an ODBC DataServer broker

You can start or stop an ODBC DataServer broker from the management console, or you can choose to start the broker automatically when the AdminServer starts.

To start or stop an ODBC DataServer broker, refer to the Starting or Stopping OpenEdge resources section in OpenEdge Management: Servers, DataServers, Messengers, and Adapters

Starting an ODBC DataServer broker automatically

You can set an ODBC DataServer broker to start automatically when the AdminServer starts.

To start an ODBC DataServer broker automatically:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the ODBC DataServer broker you want to start automatically.
3. From the list of ODBC DataServer brokers, click the ODBC DataServer that you want to edit. The ODBC DataServer details page appears.
4. In the Command and control section of the page, click Configuration. The ODBC DataServer Configuration page appears.
5. Click Edit.
6. Select the Auto start option by clicking in the check box.
7. Click Save.
Viewing the status of an ODBC DataServer broker

You can view the status of any active DataServer broker, server groups, and auxiliary processes through the management console. Most of the broker instances provide a status view, enabling you to monitor the status of multiple brokers simultaneously.

To see the status of a broker, the broker must either be running or have been running.

To view the status of an ODBC DataServer broker:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the ODBC DataServer broker whose status you want to view.
3. Click the Edit icon. The ODBC DataServer details page appears.
4. In the Operations views section of the page, click Status.

A Status summary and status details appear and provide the information described in Table 18 and Table 19, respectively.

<table>
<thead>
<tr>
<th>Table 18: ODBC DataServer broker status summary (1 of 2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Property</td>
</tr>
<tr>
<td>--------------------------------</td>
</tr>
<tr>
<td>Host</td>
</tr>
<tr>
<td>Broker Name</td>
</tr>
<tr>
<td>Operating Mode</td>
</tr>
<tr>
<td>Broker Status</td>
</tr>
<tr>
<td>Broker Port</td>
</tr>
<tr>
<td>Broker PID</td>
</tr>
<tr>
<td>Active Servers</td>
</tr>
<tr>
<td>Busy Servers</td>
</tr>
<tr>
<td>Locked Servers</td>
</tr>
<tr>
<td>Available Servers</td>
</tr>
<tr>
<td>Active Clients (now, peak)</td>
</tr>
</tbody>
</table>
Chapter 3: Configuring DataServers

Table 18: ODBC DataServer broker status summary  

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client Queue Depth (cur, max)</td>
<td>The number of clients waiting for agents to become available to service their request. The current value (cur) represents the number of waiting clients at the moment the status is displayed, and the maximum value (max) represents the largest number of clients waiting concurrently since the broker started.</td>
</tr>
<tr>
<td>Total Requests</td>
<td>N/A</td>
</tr>
<tr>
<td>Rq Wait (max, avg)</td>
<td>N/A</td>
</tr>
<tr>
<td>Rq Duration (max, avg)</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Table 19: ODBC DataServer broker status details

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PID</td>
<td>The process id of the server process</td>
</tr>
<tr>
<td>State</td>
<td>The current state of the server process</td>
</tr>
<tr>
<td>Port</td>
<td>The TCP/IP port number used by the server</td>
</tr>
<tr>
<td>nRq</td>
<td>N/A</td>
</tr>
<tr>
<td>nRcvd</td>
<td>N/A</td>
</tr>
<tr>
<td>nSent</td>
<td>N/A</td>
</tr>
<tr>
<td>Started</td>
<td>The time stamp indicating when the server process is started</td>
</tr>
<tr>
<td>Last Change</td>
<td>The time stamp indicating when the server process last changed state</td>
</tr>
</tbody>
</table>
Viewing the ODBC DataServer log files

You can view the log files for an ODBC DataServer by using the log file viewer. The log file viewer allows you to examine the ODBC DataServer log files through an HTML interface.

To access the log file viewer:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the ODBC DataServer broker whose log file viewer you want to access.

3. From the ODBC DataServer brokers list, click the ODBC DataServer that you want view. The ODBC DataServer details page appears.

4. In the Command and control section of the page, Click Log File Viewer of Broker or Log File Viewer of Servers.

You can use the Log File Viewer in the following ways:

- Use the Show field to control how many database log file entries display at one time. The number entered into the Show field cannot 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 cannot be more 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.

- Click Reload after changing the values in either the Show field or the Overlap field. If you do not reload, the viewer continues to display the previous values.

- Click Go To to control which numbered entry in the log file the viewer display begins. For example, a value of 10 entered into the Go To field will begin the display from the tenth log file entry.

**Note:** You must click Go To after entering a value in the Go To field, or the viewer does not update its display.

The default display of entries is in ascending order; choose Descending to change the display. Note that the Show field dictates the number of entries shown, regardless of whether they display in ascending or descending order.

- Click First to display the first \( x \) entries, where \( x \) is the value in the Show field.

- Click Previous 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 last $x$ entries, where $x$ is the value in the Show field.

• To view additional log file entries without changing your current starting log file entry, leave the Go To field blank, change the value in the Show field, and click Reload.

• If the contents of the log file have changed since you opened the viewer, the log file viewer indicates this in the Log file status field.

• OpenEdge Management and OpenEdge Explorer consider a viewer that has been inactive for more than four hours stale and at that point release ninety-five percent of any memory being held. If you try to use a stale viewer, OpenEdge Management and OpenEdge Explorer automatically reload the file. Because additional resource activity might have occurred during the viewer's inactivity, the reloaded log file view might not match the previous log file view of that resource.

• OpenEdge Management and OpenEdge Explorer consider a viewer that has been inactive for forty-eight hours dead. Once a viewer dies, OpenEdge Management and OpenEdge Explorer release all of its memory. To return to the log file displayed in a dead view, you must renavigate to it, even if you pinned up the view or saved a link to it before the viewer died.
Oracle DataServer configuration and administration

The OpenEdge® DataServer for Oracle allows the OpenEdge Application Development Environment (ADE) and applications created with OpenEdge in Windows to access certain Oracle databases.

The DataServer for Oracle is also supported on several UNIX and LINUX platforms. For details, see http://communities.progress.com/pcom/community/psdn.

You can use OpenEdge Management or OpenEdge Explorer to establish and administer the configuration of a Progress OpenEdge DataServer for Oracle. The DataServer for Oracle installation provides one predefined DataServer broker (orabroker1). You can use this predefined component as a starting point for creating and configuring additional DataServer brokers. Each broker is referred to as an instance.

Working with the Oracle DataServer broker

To work with the Oracle DataServer, double-click the Oracle DataServer folder from the management console list frame. There is one pre-defined default Oracle DataServer (orabroker1).

You can perform the following Oracle DataServer actions:

- Configuring an Oracle DataServer broker
- Editing an Oracle DataServer broker configuration
- Creating an Oracle DataServer broker
- Starting or Stopping an Oracle DataServer broker
- Starting an Oracle DataServer broker automatically
- Viewing the status of an Oracle DataServer broker
- Deleting an Oracle DataServer broker
- Viewing the ODBC DataServer log files
Configuring an Oracle DataServer broker

You can view or modify the following default configuration properties of an Oracle DataServer broker from OpenEdge Management or OpenEdge Explorer:

- Broker
- Server
- Environment variables

You can also create a new Oracle DataServer broker, view a broker’s status, delete a broker you no longer need, or view the broker’s log files.

Each of these tasks is described in the following sections.

**Viewing or modifying Oracle DataServer broker properties**

You can view or modify Oracle DataServer broker properties in the following categories: general properties, owner information, controlling NameServer information, Data Service, setting logging properties, and advanced features.

To view or modify broker properties for an Oracle broker:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the Oracle DataServer broker whose properties you want to view or modify.
3. Click the Oracle DataServer broker that you want to view. The **Oracle DataServer** details page appears.
4. In the **Command and control** section of the page, Click **Configuration**. The **Oracle DataServer Configuration** page appears.

From this page, you can:

- View a read-only display of the Oracle broker properties as described in Table 20.
- Click **Edit** to modify the broker properties. For details, see the “Editing an Oracle DataServer broker configuration” section on page 93.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>General</strong></td>
<td></td>
</tr>
<tr>
<td>Working directory</td>
<td>Your general working directory. The default value is @{WorkPath}.</td>
</tr>
<tr>
<td>Operating mode</td>
<td>For DataServer brokers, the operating mode is state-aware. The default value is state-aware.</td>
</tr>
</tbody>
</table>
Table 20: Oracle DataServer broker properties (2 of 5)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Port number</td>
<td>The type of the TCP/IP listening port that the DataServer broker uses. The default value is 3050.</td>
</tr>
<tr>
<td>Auto start</td>
<td>Starts the DataServer automatically when the AdminServer starts. The Auto start check box is cleared by default.</td>
</tr>
<tr>
<td>Owner Information</td>
<td></td>
</tr>
<tr>
<td>Group name</td>
<td>(UNIX only) The name of the group.</td>
</tr>
<tr>
<td>Username</td>
<td>Your user name. The default is blank.</td>
</tr>
<tr>
<td>Password</td>
<td>Your password. The default is blank.</td>
</tr>
<tr>
<td>Controlling NameServer</td>
<td></td>
</tr>
<tr>
<td>Register with NameServer</td>
<td>A check box that indicates whether to register the DataServer with a controlling NameServer.</td>
</tr>
<tr>
<td>Controlling NameServer</td>
<td>Identifies the NameServer that the DataServer uses to find a broker that supports the correct application services. (If you did not select to register the DataServer with a NameServer, the field is dimmed and unavailable.)</td>
</tr>
<tr>
<td>Registration mode</td>
<td>How the broker specifies its hostname when registering with its controlling NameServer. This hostname information is passed onto a client application when it attempts to connect to a Data Service that the broker supports. The choices are:</td>
</tr>
<tr>
<td></td>
<td>• Register IP — (Default) This setting is the most efficient mechanism, and can be used in most cases. It registers with the IP address of the machine where the broker is located.</td>
</tr>
<tr>
<td></td>
<td>• Register-LocalHost — The broker registers with the hostname of the machine that it runs on. Use this setting when the broker runs on a machine with a single hostname and more than one IP address.</td>
</tr>
<tr>
<td></td>
<td>• Register-HostName — The broker registers with the values specified in the hostName property. Use this setting when your clients need a fully qualified host name to connect to a broker in a different DNS domain.</td>
</tr>
</tbody>
</table>
### Chapter 3: Configuring DataServers

**Table 20: Oracle DataServer broker properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registration host name</td>
<td>Identifies the hostname value to be used if the <strong>Registration mode</strong> property is set to <strong>Register-HostName</strong>.</td>
</tr>
<tr>
<td>Data Service</td>
<td></td>
</tr>
<tr>
<td>Data service name list</td>
<td>Data services are services that are provided by a selected broker's servers. The default service name is the broker name. The default value is blank.</td>
</tr>
<tr>
<td>Supports default service</td>
<td>A check box that indicates whether the ODBC broker supports the default service. Select the check box to have the broker support the default service, or click to clear the check box and cancel support for the default service. The <strong>Supports default service</strong> check box is cleared by default.</td>
</tr>
<tr>
<td>Logging Setting</td>
<td></td>
</tr>
<tr>
<td>Broker log filename</td>
<td>The general broker log filename. The default is @{WorkPath}\orabroker1.broker.log.</td>
</tr>
<tr>
<td>Broker logging level</td>
<td>The amount of information written to the broker log. The possible values are as follows:</td>
</tr>
<tr>
<td></td>
<td>• <strong>None</strong> — No log file is written.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Error Only</strong> — Only error-related information is written to the log file, which reduces the size of the log file.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Basic</strong> — Only some information is written to the log file.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Verbose</strong> — All broker activity is written to the log file. This might produce large files that use up disk space.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Extended</strong> — Information in addition to <strong>Verbose</strong>.</td>
</tr>
<tr>
<td>Append to broker log file</td>
<td>Create a new broker log file each time a broker is started, even if the broker log file specified already exists. The <strong>Append to broker log</strong> file check box is selected by default.</td>
</tr>
<tr>
<td>Broker logging entry types</td>
<td>A single entry or comma delimited list of logging entry types.</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Broker log file threshold size</td>
<td>Specifies a limit on how big the log file can get (in bytes). When the log is full, the broker creates a new log file with a sequence number.</td>
</tr>
<tr>
<td>Maximum number of broker log files</td>
<td>Limit on how many broker log files will be kept on the system when there is a threshold limit. It represents the number of log files including the current on and includes any rolled over log files that already exist when the process started.</td>
</tr>
</tbody>
</table>

**Advanced Features**

| Maximum client instances | The maximum number of client connections that the broker can support concurrently.  
The default value of 512 that is set for this parameter is high enough to ensure that the number of client connections is unlimited. Realistically, however, the system-level resources needed to support an unlimited number of clients might be exhausted before this limit is reached. If you see broker log entries that contain Exception Messages for "out of space" or "OutOfMemory" you might need to reduce the maximum number of client connections to a more reasonable value. Note, however, that lowering this value can cause some client requests to be rejected ("Exceeded Max Clients"). Configuring and starting multiple brokers to handle higher client loads alleviates the problem. |

---

*Table 20: Oracle DataServer broker properties (4 of 5)*
### Table 20: Oracle DataServer broker properties (5 of 5)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Priority weight (0-100)</strong> (dynamic property)</td>
<td>An integer value between 0 and 100 that influences the share of the workload that the selected broker receives. The larger the value, the heavier the load that is distributed to the broker. Using the <strong>Priority weight</strong> value, the NameServer distributes client requests to the same DataServer service across all DataServer brokers. The default value is 0. You can update this property dynamically. Any changes will affect all current and new brokers.</td>
</tr>
<tr>
<td><strong>Registration retry (in seconds)</strong></td>
<td>After registering with its controlling NameServer, the DataServer broker periodically sends &quot;keep-alive&quot; messages to the NameServer to let the NameServer know that the broker is still active. The <strong>Registration retry</strong> value is the number of seconds that pass between &quot;keep-alive&quot; messages. The default value is 30.</td>
</tr>
</tbody>
</table>

---

**Viewing or modifying Oracle DataServer server properties**

You can view or modify Oracle DataServer server properties in the following categories: **General** and **Logging Setting**.

**To view or modify server properties for an Oracle broker:**

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the Oracle DataServer broker whose server properties you want to view or modify.

3. Click the Oracle DataServer broker that you want to view. The **Oracle DataServer** details page appears.

4. In the **Command and control** section of the page, click **Configuration**. The **Oracle DataServer Configuration** page appears.

From this page, you can do the following:

- View a read-only display of the Oracle server properties as described in Table 21.

- Click **Edit** to modify the server properties. For details, see the “Editing an Oracle DataServer broker configuration” section on page 93.
Table 21: Oracle DataServer server properties

<table>
<thead>
<tr>
<th>General</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Server executable file**<br>(dynamic property) | The pathname of the DataServer executable that the broker runs. The defaults for the DataServer for Oracle on the Windows and UNIX platforms are as follows:
- **Windows:** `{Startup}\DLC\bin\orasrv.exe`
- **UNIX:** `OpenEdge-install-dir/_orasrv`
You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated. |
| **Server startup parameters**<br>(dynamic property) | The startup parameters for the server. You should not modify the default parameters, which are:

```
-svub -S -X -N TCP -U X -P x -hs 0 -s 40
```

You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated. |
| **PROPATH** | A list of directories OpenEdge searches to find procedures. |
| **Minimum port number**<br>(dynamic property) | The minimum port number that you want to make available for each DataServer broker. The default is blank.
You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated. |
| **Maximum port number**<br>(dynamic property) | The maximum port number that you want to make available for each DataServer broker. The default is blank.
You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated. |
Setting Oracle DataServer broker environment variables

You can set environment variables for the DataServer broker for Oracle. You can set one or more environment variables that:

- Are accessible from your ABL application
- Affect the environment in which your application runs

For the Oracle DataServer, there are no required environment variables in Windows.

To set Oracle DataServer broker environment variables:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the Oracle DataServer broker whose environment variables you want to set.
3. Click the Oracle DataServer broker that you want to view. The **Oracle DataServer** details page appears.
4. In the **Command and control** section of the page, click **Configuration**. The **Oracle DataServer Configuration** page appears.
5. Click **Edit**.
6. Click **Environment Variables**.
7. Enter each variable name and then provide its value in the format **name=value**.
8. Click **Save** when you finish adding the variables and their values.

### Table 21: Oracle DataServer server properties (2 of 2)

<table>
<thead>
<tr>
<th>Server properties</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Logging Setting</strong></td>
<td>The DataServer log filename. Enter a valid path name. Installed Default: @(\text{WorkPath})\orabroker1.server.log</td>
</tr>
</tbody>
</table>

| Server log filename | The DataServer log filename. Enter a valid path name. Installed Default: @\(\text{WorkPath}\)\orabroker1.server.log |
Editing an Oracle DataServer broker configuration

You can edit an Oracle DataServer configuration from the management console.

To edit an Oracle DataServer broker configuration:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the Oracle DataServer broker whose configuration you want to edit.
3. Click the Oracle DataServer broker. The Oracle DataServer details page appears.
4. In the Command and control section of the page, click Configuration. The Oracle DataServer Configuration page appears.
5. Click Edit. (For details about the broker or server properties, see Table 20 and Table 21, respectively.)
6. Make the configuration changes, and then click Save.

Creating an Oracle DataServer broker

OpenEdge Management and OpenEdge Explorer provide a predefined default Oracle DataServer configuration; however, you can create a new configuration by modifying the default settings.

To create an Oracle DataServer broker:

1. From the drop-down for Resources on the management console menu, click New OpenEdge Resource → Oracle DataServer. The Oracle DataServer Configuration page appears.
2. Type the name of the new Oracle DataServer in the field provided.
3. Click Save. The Oracle DataServer Configuration page appears, allowing you to configure the broker’s properties.
Deleting an Oracle DataServer broker

You can delete an inactive Oracle DataServer broker from the management console.

If a broker is running, you cannot delete it. (The Delete button is not available.) You must first stop the broker and then delete it.

To delete an Oracle DataServer broker:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the Oracle DataServer broker you want to delete.
3. From the Oracle DataServer brokers list, click the broker that you want to delete. The Oracle DataServer details page appears.
4. Click Delete, and then click OK to confirm the deletion.

Starting or Stopping an Oracle DataServer broker

You can start or stop an Oracle DataServer broker from the management console.

To start or stop an Oracle DataServer broker, refer to the Starting or Stopping OpenEdge resources section in OpenEdge Management: Servers, DataServers, Messengers, and Adapters

Starting an Oracle DataServer broker automatically

You can set a Oracle DataServer broker to start automatically when the AdminServer starts.

To start an Oracle DataServer broker automatically:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the Oracle DataServer brokers.
3. From the search list, click the Oracle DataServer broker that you want to start automatically. The Oracle DataServer details page appears.
4. In the Command and control section of the page, click Configuration. The Oracle DataServer Configuration page appears.
5. Click Edit.
6. Select the Auto start check box.
7. Click Save.
Viewing the status of an Oracle DataServer broker

You can view the status of any active Oracle DataServer broker, server groups, and auxiliary processes through the management console. Most of the broker instances provide a status view, enabling you to monitor the status of multiple brokers simultaneously.

To see the status of a broker, the broker must be running or have been running.

To view the status of an Oracle DataServer broker:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the Oracle DataServer broker you want to start automatically.
3. Click the Oracle DataServer broker. The Oracle DataServer details page appears.
4. In the Operations views section of the page, click Status.

A Status summary and summary details appear and provide the information described in Table 22 and Table 23.

Table 22: Oracle DataServer broker property Status summary (1 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>The broker’s host machine.</td>
</tr>
<tr>
<td>Broker Name</td>
<td>The name of the broker whose status you are viewing.</td>
</tr>
<tr>
<td>Operating Mode</td>
<td>For DataServer brokers, the operating mode is state-aware.</td>
</tr>
<tr>
<td>Broker Status</td>
<td>The current state of the broker.</td>
</tr>
<tr>
<td>Broker Port</td>
<td>The TCP/IP port number that the broker listens to.</td>
</tr>
<tr>
<td>Broker PID</td>
<td>The process ID of the broker.</td>
</tr>
<tr>
<td>Active Servers</td>
<td>The number of active servers.</td>
</tr>
<tr>
<td>Busy Servers</td>
<td>N/A</td>
</tr>
<tr>
<td>Locked Servers</td>
<td>N/A</td>
</tr>
<tr>
<td>Available Servers</td>
<td>N/A</td>
</tr>
<tr>
<td>Active Clients (now, peak)</td>
<td>N/A</td>
</tr>
</tbody>
</table>
Table 22: **Oracle DataServer broker property Status summary** (2 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client Queue Depth (cur, max)</td>
<td>The number of clients waiting for agents to become available to service their request. The current value (cur) represents the number of waiting clients at the moment the status is displayed, and the maximum value (max) represents the largest number of clients waiting concurrently since the broker was started.</td>
</tr>
<tr>
<td>Total Requests</td>
<td>N/A</td>
</tr>
<tr>
<td>Rq Wait (max, avg)</td>
<td>N/A</td>
</tr>
<tr>
<td>Rq Duration (max, avg)</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Table 23: **Oracle DataServer broker Status details**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PID</td>
<td>The process id of the server process</td>
</tr>
<tr>
<td>State</td>
<td>The current state of the server process</td>
</tr>
<tr>
<td>Port</td>
<td>The TCP/IP port number used by the server</td>
</tr>
<tr>
<td>nRq</td>
<td>N/A</td>
</tr>
<tr>
<td>nRcvd</td>
<td>N/A</td>
</tr>
<tr>
<td>nSent</td>
<td>N/A</td>
</tr>
<tr>
<td>Started</td>
<td>The time stamp indicating when the server process is started</td>
</tr>
<tr>
<td>Last Change</td>
<td>The time stamp indicating when the server process last changed state</td>
</tr>
</tbody>
</table>
Viewing the Oracle DataServer broker log files

You can view the log files for an Oracle DataServer broker by using the log file viewer. The log file viewer allows you to examine the Oracle DataServer log files through an HTML interface.

To access the log file viewer:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the Oracle DataServer broker whose log file viewer you want to access.
3. From the list, click the Oracle DataServer broker. The Oracle DataServer details page appears.
4. In the Command and control section of the page, Click Log File Viewer of Broker or Log File Viewer of Servers.

You can use the Log File Viewer in the following ways:

- Use the Show field to control how many database log file entries display at one time. The number entered into the Show field cannot 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 cannot be more 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.

- Click Reload after changing the values in either the Show field or the Overlap field. If you do not reload, the viewer continues to display the previous values.
- Click Go To to control which numbered entry in the log file the viewer display begins. For example, a value of 10 entered into the Go To field will begin the display from the tenth log file entry.

**Note:** You must click Go To after entering a value in the Go To field or the viewer will not update its display.

The default display of entries is in ascending order; choose Descending to change the display. Note that the Show field dictates the number of entries shown, regardless of whether they display in ascending or descending order.

- Click First to display the first x entries, where x is the value in the Show field.
- Click Previous 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 last \( x \) entries, where \( x \) is the value in the **Show** field.

• To view additional log file entries without changing your current starting log file entry, leave the **Go To** field blank, change the value in the **Show** field, and click **Reload**.

• If the contents of the log file have changed since you opened the viewer, the log file viewer indicates this in the **Log file status** field.

• OpenEdge Management and OpenEdge Explorer consider a viewer that has been inactive for more than four hours stale and at that point release ninety-five percent of any memory being held. If you try to use a stale viewer, OpenEdge Management and OpenEdge Explorer automatically reload the file. Because additional resource activity might have occurred during the viewer’s inactivity, the reloaded log file view might not match the previous log file view of that resource.

• OpenEdge Management and OpenEdge Explorer consider a viewer that has been inactive for forty-eight hours dead. Once a viewer dies, OpenEdge Management and OpenEdge Explorer release all of its memory. To return to the log file displayed in a dead view, you must renavigate to it, even if you pinned up the view or saved a link to it before the viewer died.
MS SQL Server DataServer configuration and administration

The OpenEdge® DataServer for MS SQL Server allows the OpenEdge Application Development Environment (ADE) and applications created with OpenEdge in Windows to access Microsoft SQL Server.

You can use OpenEdge Explorer to configure and administer the configuration of an OpenEdge DataServer for SQL Server. The DataServer for SQL Server installation provides one predefined DataServer broker (mssbroker1). You can use this predefined component as a starting point for creating and configuring additional DataServer brokers. Each broker is referred to as an instance.

Working with the MS SQL Server DataServer broker

To work with the MS SQL Server DataServer, double-click the MSS DataServer folder from the OpenEdge Explorer list frame. There is one pre-defined default MS SQL Server DataServer (mssbroker1).

You can perform the following MS SQL Server actions:

- Configuring an MS SQL Server DataServer broker
- Editing an MS SQL Server DataServer broker configuration
- Creating an MS SQL Server DataServer broker
- Starting or Stopping an MS SQL Server DataServer broker
- Starting an MS SQL Server DataServer broker automatically
- Viewing the MS SQL Server DataServer log files
- Deleting an MS SQL Server DataServer broker
- Viewing the MS SQL Server DataServer log files
Chapter 3: Configuring DataServers

Configuring an MS SQL Server DataServer broker

You can view or modify the following configuration properties of an MS SQL Server DataServer from the management console:

- Broker
- Server
- Environment variables

You can also create a new MS SQL Server DataServer broker, view a broker’s status, or delete a broker you no longer need.

Each of these tasks is described in the following sections.

Viewing or modifying MS SQL Server DataServer broker properties

You can view or modify MS SQL Server DataServer Broker properties in the following categories: general properties, owner information, controlling NameServer information, Data Service, setting logging properties, and advanced features.

To view or modify broker properties for an MS SQL Server broker:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the MS SQL DataServer broker whose properties you want to view or modify.
3. Click the Edit icon. The MSS DataServer details page appears.
4. In the Command and control section of the page, click Configuration. The MSS DataServer Configuration page appears.

From this page, you can:

- View a read-only display of the MS SQL Server broker properties as described in Table 24.
- Click Edit to modify the broker properties. For details, see the “Editing an MS SQL Server DataServer broker configuration” section on page 106.

Table 24: MS SQL Server DataServer broker properties (1 of 4)

<table>
<thead>
<tr>
<th>Property category or property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>General</td>
<td></td>
</tr>
<tr>
<td>Working directory</td>
<td>Your general working directory. The default value is @{WorkPath}.</td>
</tr>
<tr>
<td>Operating mode</td>
<td>For DataServer brokers, the operating mode is state-aware. The default value is state-aware.</td>
</tr>
</tbody>
</table>
### Table 24: MS SQL Server DataServer broker properties (2 of 4)

<table>
<thead>
<tr>
<th>Property category or property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Port number</strong></td>
<td>The type of the TCP/IP listening port that the DataServer broker uses. The default value is 3050.</td>
</tr>
<tr>
<td><strong>Auto start</strong></td>
<td>Starts the DataServer automatically when the AdminServer starts. The <strong>Auto start</strong> check box is cleared by default.</td>
</tr>
</tbody>
</table>

**Owner Information**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Group name</strong></td>
<td>(UNIX only) The name of the group.</td>
</tr>
<tr>
<td><strong>Username</strong></td>
<td>Your user name. The default is blank.</td>
</tr>
<tr>
<td><strong>Password</strong></td>
<td>Your password. The default is blank.</td>
</tr>
</tbody>
</table>

**Controlling NameServer**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Register with NameServer</strong></td>
<td>A check box that indicates whether to register the DataServer with a controlling NameServer.</td>
</tr>
<tr>
<td><strong>Controlling NameServer</strong></td>
<td>Identifies the NameServer that the DataServer uses to find a broker that supports the correct application services. (If you did not select to register the DataServer with a NameServer, the field is dimmed and unavailable.)</td>
</tr>
<tr>
<td><strong>Registration mode</strong></td>
<td>How the broker specifies its hostname when registering with its controlling NameServer. This hostname information is passed onto a client application when it attempts to connect to a Data Service that the broker supports. The choices are:</td>
</tr>
<tr>
<td></td>
<td>- <strong>Register IP</strong> — (Default) This setting is the most efficient mechanism, and can be used in most cases. It registers with the IP address of the machine where the broker is located.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Register-LocalHost</strong> — The broker registers with the hostname of the machine that it runs on. Use this setting when the broker runs on a machine with a single hostname and more than one IP address.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Register-HostName</strong> — The broker registers with the values specified in the hostName property. Use this setting when your clients need a fully qualified host name to connect to a broker in a different DNS domain.</td>
</tr>
<tr>
<td></td>
<td><strong>Registration host name</strong> (UNIX only) The name of the group.</td>
</tr>
<tr>
<td>Property category or property</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Data Service</strong></td>
<td></td>
</tr>
<tr>
<td>Data service name list</td>
<td>Data services are services that are provided by a selected broker's servers. The default service name is the broker name. The default value is blank.</td>
</tr>
<tr>
<td>Supports default service</td>
<td>A check box that indicates whether the MS SQL Server broker supports the default service. Select the check box to have the broker support the default service, or click to clear the check box and cancel support for the default service. The Supports default service check box is cleared by default.</td>
</tr>
<tr>
<td><strong>Logging Setting</strong></td>
<td></td>
</tr>
<tr>
<td>Broker log filename</td>
<td>The general broker log filename. The default is @WorkPath\mssbroker1.broker.log.</td>
</tr>
<tr>
<td>Broker logging level</td>
<td>The amount of information written to the broker log. The possible values are as follows:</td>
</tr>
<tr>
<td></td>
<td>• <strong>None</strong> — No log file is written.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Error Only</strong> — Only error-related information is written to the log file, which reduces the size of the log file.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Basic</strong> — Only some information is written to the log file.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Verbose</strong> — All broker activity is written to the log file. This might produce large files that use up disk space.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Extended</strong> — Information in addition to <strong>Verbose</strong>.</td>
</tr>
<tr>
<td>Append to broker log file</td>
<td>Create a new broker log file each time a broker is started, even if the broker log file specified already exists. The Append to broker log file check box is selected by default.</td>
</tr>
<tr>
<td>Broker logging entry types</td>
<td>A single entry or comma delimited list of logging entry types.</td>
</tr>
<tr>
<td>Broker log file threshold size</td>
<td>Specifies a limit on how big the log file can get (in bytes). When the log is full, the broker creates a new log file with a sequence number.</td>
</tr>
</tbody>
</table>
### Maximum number of broker log files

Limit on how many broker log files will be kept on the system when there is a threshold limit. It represents the number of log files including the current on and includes any rolled over log files that already exist when the process started.

### Advanced Features

#### Maximum client instances

The maximum number of client connections that the broker can support concurrently.

The default value of 512 that is set for this parameter is high enough to ensure that the number of client connections is unlimited. Realistically, however, the system-level resources needed to support an unlimited number of clients might be exhausted before this limit is reached. If you see broker log entries that contain Exception Messages for "out of space" or "OutOfMemory" you may need to reduce the maximum number of client connections to a more reasonable value. Note, however, that lowering this value can cause some client requests to be rejected ("Exceeded Max Clients"). Configuring and starting multiple brokers to handle higher client loads alleviates the problem.

#### Priority weight (0-100) (dynamic property)

An integer value between 0 and 100 that influences the share of the workload that the selected broker receives. The larger the value, the heavier the load that is distributed to the broker. Using the Priority weight value, the NameServer distributes client requests to the same DataServer service across all DataServer brokers. The default value is 0.

You can update this property dynamically. Any changes will affect all current and new brokers.

#### Registration retry (in seconds)

After registering with its controlling NameServer, the DataServer broker periodically sends "keep-alive" messages to the NameServer to let the NameServer know that the broker is still active. The Registration retry value is the number of seconds that pass between "keep-alive" messages. The default value is 30.
Viewing or modifying MS SQL Server DataServer server properties

You can view or modify MS SQL Server DataServer properties in the following categories: **General** and **Logging Setting**.

**To view or modify server properties for an MS SQL DataServer broker:**

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the MS SQL DataServer broker whose server properties you want to view or modify.

3. From the list, click the MS SQL DataServer broker. The **MSS DataServer** details page appears.

4. In the **Command and control** section of the page, click **Configuration**. The **MSS DataServer Configuration** page appears.

   From this page, you can:

   • View a read-only display of the MS SQL Server server properties as described in Table 25.

   • Click **Edit** to modify the server properties. For details, see the “Editing an MS SQL Server DataServer broker configuration” section on page 106.

   • View the MS SQL Server DataServer log file. For details, see the “Viewing the MS SQL Server DataServer log files” section on page 111.

**Table 25: MS SQL Server DataServer server properties (1 of 2)**

<table>
<thead>
<tr>
<th>Property category or property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>General</strong></td>
<td></td>
</tr>
</tbody>
</table>
| **Server executable file**   | The pathname of the DataServer executable that the broker runs. The default for the DataServer for MS SQL Server in Windows is:  
                                @\{Startup\DLC\}bin\_msssrv.exe  
                                You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated. |
Setting MS SQL Server DataServer broker environment variables

You can set environment variables for the DataServer broker for MS SQL Server. You can set one or more environment variables that:

- Are accessible from your ABL application
- Affect the environment in which your application runs

For the MS SQL Server DataServer, there are no required environment variables in Windows.

### Table 25: MS SQL Server DataServer server properties (2 of 2)

<table>
<thead>
<tr>
<th>Property category or property</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Server startup parameters** (dynamic property) | The startup parameters for the server. You should not modify the default parameters, which are: `-svub -S -X -N TCP -U X -P x -hs 0 -s 40`
You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated. |
| PROPATH | A list of directories OpenEdge searches to find procedures. |
| **Minimum port number** (dynamic property) | The minimum port number that you want to make available for each DataServer broker. The default is blank. You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated. |
| **Maximum port number** (dynamic property) | The maximum port number that you want to make available for each DataServer broker. The default is blank. You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated. |
| **Logging Setting** | |
| **Server log filename** | The DataServer log filename. Enter a valid path name.
Installed Default: `@{WorkPath}\mssbroker1.server.log` |
To set MS SQL Server DataServer environment variables:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the MS SQL DataServer whose environment variables you want to set.

3. From the list, click the MS SQL DataServer broker. The **MSS DataServer** details page appears.

4. In the **Command and control** section of the page, click **Configuration**. The **MSS DataServer Configuration** page appears.

5. Click **Edit**.

6. Click **Environment Variables**.

7. Enter each variable name and then provide its value in the format `name=value`.

8. Click **Save** when you finish adding the variables and their values.

**Editing an MS SQL Server DataServer broker configuration**

You can edit an MS SQL Server DataServer broker configuration from the management console.

To edit an MS SQL Server DataServer configuration:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the MS SQL DataServer whose configuration you want to edit.

3. From the list, click the MS SQL DataServer broker. The **MSS DataServer** details page appears.

4. In the **Command and control** section of the page, click **Configuration**. The **MSS DataServer Configuration** page appears.

5. Click **Edit**. (For details about the broker and server properties, see Table 24 and Table 25, respectively.)

6. Make the configuration changes, and then click **Save**.
Creating an MS SQL Server DataServer broker

OpenEdge Management and OpenEdge Explorer provide a predefined default MS SQL Server DataServer configuration; however, you can create a new configuration by modifying the default settings.

To create an MS SQL Server DataServer broker:

1. From the drop-down for Resources on the management console menu, click New OpenEdge Resource → MSS DataServer. The MSS DataServer Configuration page appears.
2. Type the name of the new DataServer in the field provided.
3. Click Save. The MSS DataServer Configuration page appears, allowing you to establish the broker’s properties.

Deleting an MS SQL Server DataServer broker

You can delete an inactive DataServer broker for MS SQL Server from the management console.

If a broker is running, you cannot delete it. You must first stop the broker and then delete it.

To delete an MS SQL Server DataServer broker:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the MS SQL DataServer you want to delete.
3. From the list, click the MS SQL DataServer broker. The MSS DataServer details page appears.
4. Click Delete, and then click OK to confirm the deletion.
Starting or Stopping an MS SQL Server DataServer broker

You can start or stop an MS SQL Server DataServer broker from the management console.

To start or stop an MS SQL DataServer broker, refer to the Starting or Stopping OpenEdge resources section in OpenEdge Management: Servers, DataServers, Messengers, and Adapters.

Starting an MS SQL Server DataServer broker automatically

You can set a broker for MS SQL Server to start automatically when the AdminServer starts.

To start an MS SQL DataServer broker automatically:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the MS SQL DataServer you want to start automatically.
3. MS SQL DataServer broker. The MSS DataServer details page appears.
4. In the Command and control section of the page, click Configuration. The MSS DataServer Configuration page appears.
5. Click Edit.
6. Select the Auto start option by clicking in the check box.
7. Click Save.
Viewing the status of an MS SQL Server DataServer broker

You can view the status of any active MS SQL Server DataServer broker, server groups, and auxiliary processes through the management console. Most of the broker instances provide a status view, enabling you to monitor the status of multiple brokers simultaneously.

To see the status of a broker, the broker must be running or have been running.

To view the status of an MS SQL Server DataServer broker:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the MS SQL DataServer broker whose status you want to view.
3. From the list, click MS SQL DataServer broker. The MSS DataServer details page appears.
4. In the Operations views section of the page, click Status.

A Status summary and summary details appear and provide the information described in Table 26 and Table 27, respectively.

Table 26: MS SQL Server DataServer broker property status summary (1 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>The broker’s host machine.</td>
</tr>
<tr>
<td>Broker Name</td>
<td>The name of the broker whose status you are viewing.</td>
</tr>
<tr>
<td>Operating Mode</td>
<td>For DataServer brokers, the operating mode is state-aware.</td>
</tr>
<tr>
<td>Broker Status</td>
<td>The current state of the broker.</td>
</tr>
<tr>
<td>Broker Port</td>
<td>The TCP/IP port number that the broker listens to.</td>
</tr>
<tr>
<td>Broker PID</td>
<td>The process ID of the broker.</td>
</tr>
<tr>
<td>Active Servers</td>
<td>The number of active servers.</td>
</tr>
<tr>
<td>Busy Servers</td>
<td>N/A</td>
</tr>
<tr>
<td>Locked Servers</td>
<td>N/A</td>
</tr>
<tr>
<td>Available Servers</td>
<td>N/A</td>
</tr>
<tr>
<td>Active Clients (now, peak)</td>
<td>N/A</td>
</tr>
</tbody>
</table>
Table 26: MS SQL Server DataServer broker property status summary (2 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client Queue Depth (cur, max)</td>
<td>The number of clients waiting for agents to become available to service their request. The current value (cur) represents the number of waiting clients at the moment the status is displayed, and the maximum value (max) represents the largest number of clients waiting concurrently since the broker was started.</td>
</tr>
<tr>
<td>Total Requests</td>
<td>N/A</td>
</tr>
<tr>
<td>Rq Wait (max, avg)</td>
<td>N/A</td>
</tr>
<tr>
<td>Rq Duration (max, avg)</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Table 27: MS SQL Server DataServer broker status details

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PID</td>
<td>The process id of the server process</td>
</tr>
<tr>
<td>State</td>
<td>The current state of the server process</td>
</tr>
<tr>
<td>Port</td>
<td>The TCP/IP port number used by the server</td>
</tr>
<tr>
<td>nRq</td>
<td>N/A</td>
</tr>
<tr>
<td>nRcvd</td>
<td>N/A</td>
</tr>
<tr>
<td>nSent</td>
<td>N/A</td>
</tr>
<tr>
<td>Started</td>
<td>The time stamp indicating when the server process is started</td>
</tr>
<tr>
<td>Last Change</td>
<td>The time stamp indicating when the server process last changed state</td>
</tr>
</tbody>
</table>
Viewing the MS SQL Server DataServer log files

You can view the log files for an MS SQL Server DataServer by using the log file viewer. The log file viewer allows you to examine the MS SQL Server DataServer log files through an HTML interface.

To access the log file viewer:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the MS SQL DataServer broker whose log file viewer you want to access.
3. From the list, click the MS SQL DataServer broker. The MSS DataServer details page appears.
4. In the Command and control section of the page, Click Log File Viewer of Broker or Log File Viewer of Servers.

You can use the Log File Viewer in the following ways:

• Use the Show field to control how many database log file entries display at one time. The number entered into the Show field cannot 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 cannot be more 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.

• Click Reload after changing the values in either the Show field or the Overlap field. If you do not reload, the viewer continues to display the previous values.

• Click Go To to control which numbered entry in the log file the viewer begins its display with. For example, a value of 10 entered into the Go To field will begin the display from the tenth log file entry.

Note: You must click Go To after entering a value in the Go To field or the viewer will not update its display.

The default display of entries is in ascending order; choose Descending to change the display. Note that the Show field dictates the number of entries shown, regardless of whether they display in ascending or descending order.

• Click First to display the first x entries, where x is the value in the Show field.

• Click Previous 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 last x entries, where x is the value in the Show field.

- To view additional log file entries without changing your current starting log file entry, leave the Go To field blank, change the value in the Show field, and click Reload.

- If the contents of the log file have changed since you opened the viewer, the log file viewer indicates this in the Log file status field.

- OpenEdge Management and OpenEdge Explorer consider a viewer that has been inactive for more than four hours stale and at that point release ninety-five percent of any memory being held. If you try to use a stale viewer, OpenEdge Management and OpenEdge Explorer automatically reload the file. Because additional resource activity might have occurred during the viewer's inactivity, the reloaded log file view might not match the previous log file view of that resource.

- OpenEdge Management and OpenEdge Explorer consider a viewer that has been inactive for forty-eight hours dead. Once a viewer dies, OpenEdge Management and OpenEdge Explorer release all of its memory. To return to the log file displayed in a dead view, you must renavigate to it, even if you pinned up the view or saved a link to it before the viewer died.
Configuring NameServers

You can configure NameServer property settings and add new NameServer instances in OpenEdge Management and OpenEdge Explorer. You can also start, stop, and delete NameServer instances, as well as view their status and log files.

This chapter provides information about the following:

- NameServer configuration and administration
- Configuring a NameServer
- Creating a NameServer
- Configuring fault tolerance and load balancing
- Starting or Stopping a NameServer
- Viewing the status of a NameServer
- Deleting a NameServer
- Viewing the status of a NameServer
- Handling UDP and firewall issues
- Viewing the NameServer log file
NameServer configuration and administration

You can optionally use the NameServer to mediate client connections for instances of OpenEdge Unified Broker products that you configure with OpenEdge Management or OpenEdge Explorer. The Unified Broker products include all OpenEdge resources that are managed through the AdminServer. For more information on the Unified Broker products and the AdminServer, see OpenEdge Getting Started: Installation and Configuration.

When you choose to use the NameServer, each component instance registers with only one NameServer, which is known as the instance’s controlling NameServer. Each time an instance starts up, it registers with its controlling NameServer by sending its location and other configuration information. The NameServer uses this information to help resolve client connection requests. A particular NameServer can accept registrations from any number of instances, including instances of more than one type of component.

Depending on the way you configure and use the OpenEdge product, you can decide to specify a controlling NameServer to handle client connection requests. For example, your configuration might be simple and not require the location transparency the NameServer provides, or it might be more complex and, therefore, dependent on this and other NameServer features.

The OpenEdge Unified Broker products that you can optionally choose to register with the NameServer include:

- AppServers
- AppServer Internet Adapters
- DataServers for ODBC, Oracle, and MS SQL Server configured to use the NameServer
- SonicMQ Adapters
- WebSpeed Transaction Servers

NameServer features

You can use OpenEdge Management or OpenEdge Explorer to configure both local NameServer instances and remote NameServer instances. The NameServer installation provides one sample NameServer (NS1), and you can use the sample as a template for creating and configuring additional NameServers.

In its role as a connection arbitrator, the NameServer provides location transparency, connection-level fault tolerance, server-level fault tolerance, and optional load balancing. For details about these features, see the “Configuring fault tolerance and load balancing” section on page 128.
Working with the NameServer

To work with a NameServer instance, select the NameServer folder from the management console's list pane. There is one pre-defined default NameServer (NS1).

You can perform the following NameServer actions:

- Configuring a NameServer
- Editing a NameServer configuration
- Creating a NameServer
- Starting or Stopping a NameServer
- Deleting a NameServer
- Configuring fault tolerance and load balancing
- Viewing the status of a NameServer
- Handling UDP and firewall issues
- Viewing the NameServer log file

NameServer as an optional client connection request handler

Configuring a NameServer to work with the AppServer, a DataServer, a SonicMQ Adapter, or a WebSpeed Transaction Server is optional. Depending on the way you configure and use the OpenEdge product, you can decide if you want to specify a controlling NameServer. For example, your configuration might be simple and not require location transparency or load balancing, or it might be more complex and, therefore, dependent on those features.

Local and remote NameServers

Before configuring a NameServer, you must know the network location where it is to run. For any network host, you can configure two types of NameServer instances:

- Local — An instance that runs locally on the host where it is defined
- Remote — An instance that references a NameServer that is defined and runs on a machine that is remote from the host where the reference is defined

The purpose of defining a remote NameServer instance is to provide an easy way for having multiple Unified Broker products refer to a controlling NameServer that runs on another machine. You cannot start, stop, obtain status, or modify any properties of a remote NameServer instance other than the host name and port number that it references.
Configuring a NameServer

The OpenEdge® NameServer controls a pool of brokers that register (with the NameServer) the application services they provide. The NameServer can then direct client connection requests to a broker that supports a requested application service. The NameServer also provides location transparency, and, when the NameServer Enterprise Edition is installed, the ability to perform load balancing. Load balancing allows you to balance client workload among multiple brokers that service the same application (that is, the same set of procedures and resources).

For more information, see the “Configuring local instances” section on page 116 and the “Configuring remote instances” section on page 123.

Configuring local instances

You configure a local NameServer to run on the selected host where it is defined. Unified Broker products running on the same host as a local NameServer instance can use it as their controlling NameServer.

You can view or edit the configuration properties of a NameServer from the management console. You can also create a new NameServer, view a NameServer’s status, delete a NameServer you no longer need, or view the NameServer’s log file.

To configure a local NameServer instance:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the local NameServer broker instance you want to configure.

3. Click the NameServer broker instance. The NameServer details page appears.

4. In the Command and control section of the page, click Configuration. The NameServer Configuration page appears.

From this page, you can:

- View a read-only display of the NameServer properties as described in Table 28.

- Click Edit to modify the NameServer properties. For details, see the “Editing a NameServer configuration” section on page 124.
<table>
<thead>
<tr>
<th>Property category or property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Location</strong></td>
<td>A read-only property that indicates whether the NameServer is local or remote:</td>
</tr>
</tbody>
</table>
| Location                     | - **Local** — The NameServer runs locally on the selected host. You can configure all the properties of a local NameServer.  
- **Remote** — The NameServer runs remotely on a network machine that is separate from the selected host.  
You can set this property only when you first create the NameServer instance. To change the location of a NameServer, you must first delete and then recreate the NameServer with the new location setting. |
| Host name                    | The name of the host. If the NameServer is local, this is set to `localhost`. Otherwise, type the name of the remote host where the NameServer is to run. |
| Port number                  | The number of the UDP port that the NameServer uses to listen for client connection requests and registration messages from AppServers, DataServers, and WebSpeed Transaction Servers. |
| General                      | The NameServer working directory. Enter the pathname. |
Chapter 4: Configuring NameServers

Table 28: NameServer properties (2 of 6)

<table>
<thead>
<tr>
<th>Property category or property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Broker keep alive timeout</strong> (dynamic property)</td>
<td>A value, in seconds, that indicates how often the NameServer checks for Unified Broker instances that have timed out. When an OpenEdge Unified Broker instance registers with a NameServer, the instance indicates how often it sends &quot;keep-alive&quot; messages by setting a registration retry property value (a property setting in Advanced Features for the Unified Broker). Once a NameServer determines that it did not receive a &quot;keep-alive&quot; message from a Unified Broker instance within the broker's registration retry time, the NameServer automatically unregisters the instance. Note that you should use a <strong>Broker keep alive timeout</strong> value that is somewhat larger than the Unified Broker instance's registration retry value. The NameServer adjusts the specified value to allow for normal networking delays that can occur within your computer network. To keep the NameServer from using up computer resources unnecessarily, set the <strong>Broker keep alive timeout</strong> to a value that is at least 30% larger than the typical Unified Broker registration retry value. You can update this property dynamically. Any changes will affect all current and new NameServers.</td>
</tr>
<tr>
<td><strong>Auto start</strong></td>
<td>A check box that you can select if you want the NameServer to start automatically when the controlling AdminServer starts.</td>
</tr>
<tr>
<td><strong>jvmargs</strong></td>
<td>Any jvm arguments you set.</td>
</tr>
<tr>
<td><strong>Logging Setting</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Server log filename</strong></td>
<td>The NameServer log filename. Enter a valid path name. The installed default is @{WorkPath}\NS1.NS.log.</td>
</tr>
</tbody>
</table>
Table 28: NameServer properties (3 of 6)

<table>
<thead>
<tr>
<th>Property category or property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NameServer logging level</strong></td>
<td>A value that specifies the amount of information that is written to the NameServer log. Select from the following values in the drop-down list:</td>
</tr>
<tr>
<td>(dynamic property)</td>
<td>- <strong>None</strong> — Log no entries. This is equivalent to turning logging off.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Error Only</strong> — Log OpenEdge error messages. This includes all error messages and is unrelated to the entry types specified. Errors continue to be logged at all higher levels.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Basic</strong> — Logging entry types determine the logged information. Each entry type generates at least some output. This is the default.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Verbose</strong> — Logging entry types determine the logged information, but it is typically more information than <strong>Basic</strong>.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Extended</strong> — Logging entry types determine the logged information, but it is typically more information than <strong>Verbose</strong>.</td>
</tr>
<tr>
<td></td>
<td>The installed default is <strong>Basic</strong>.</td>
</tr>
<tr>
<td></td>
<td>You can update this property dynamically. Any changes will affect all current and new NameServers.</td>
</tr>
<tr>
<td><strong>Append to NameServer log file</strong></td>
<td>A check box that indicates whether a new server log file should be created when the NameServer is started.</td>
</tr>
<tr>
<td></td>
<td>To create a new NameServer log file each time the NameServer is started, even if the NameServer log file already exists, clear the check box.</td>
</tr>
<tr>
<td></td>
<td>To append log entries to the existing NameServer log file, select the check box.</td>
</tr>
<tr>
<td></td>
<td><strong>Append to NameServer log file</strong> is selected by default.</td>
</tr>
</tbody>
</table>
### Logging entry types (dynamic property)

The default and only supported entry is NSPlumbing. Specifying the NSPlumbing log entry type turns on logging for different NameServer actions based on logging level:

- **Basic** — Logs messages when a NameServer starts and stops, for load balancing status, and when registering and unregistering brokers. This is the default logging level.
- **Verbose** — Same as Basic, plus logs requests from clients and status information on the communication between client and NameServer and between NameServer and brokers.
- **Extended** — Same as Verbose, plus logs debugging messages on broker unregistering.

Log entries generated for this type contain the tag NSPlumbing.

You can update this property dynamically. Any changes will affect all current and new NameServers.

### Log file threshold size

A value of 0 or a value between 500,000 and 2,147,483,647, where 0 means there is no limit on the log file size other than what the operating system imposes.

### Maximum number of log files

The number of rolled-over log files to keep. The value can be 0 or a value between 2 and 999999, where 0 means there is no limit on the number of NameServer log files to keep. The specified number represents the maximum total number of log files to keep on disk at any time, including the current log file being written to.

When the file becomes equal to or greater than the Log file threshold size, the client process renames it and creates a new log file. The file is renamed as follows, where ###### is a number starting at 000001 and increasing to 999999, after which it rolls back over to 000001:

```
filename.######.extension
```
### Configuring a NameServer

#### Advanced Features

<table>
<thead>
<tr>
<th>Property category or property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neighboring NameServers</td>
<td>A list of selected NameServers to which this NameServer can forward connection requests for application services that are not registered with it (that is, the application service name is unknown). The list contains all NameServer instances defined in the management console, initially with none selected. To make a NameServer a neighboring NameServer, select the unselected NameServer to highlight it. You can select as many NameServer instances as you want to be neighboring NameServers. Each neighboring NameServer thus appears highlighted in the list. To remove a NameServer from the list of neighboring NameServers, select the already-selected NameServer to remove the highlight. The setting of this property is optional. When a NameServer receives a request for an application service name that is unknown and there are no neighboring NameServers specified, the NameServer sends a message to the requesting client indicating that the application service is unknown. When such a NameServer has neighboring NameServers specified, it forwards the request to each of the neighboring NameServers. The first neighboring NameServer that has the application service name, and responds to the connection request, provides the specified connection to the client.</td>
</tr>
</tbody>
</table>

#### TCP/IP Version

Internet Protocol for network communication:
- **IPv4** — Accepts only IPv4 connections
- **IPv6** — Allows IPv4 and IPv6 connections using mapped address (where supported)  

The default is IPv4. You must also set the `jvmArgs` property in the `ubroker.properties` file to override the default behavior of the JVM for IPv6 connections.

```
jvmArgs=-Djava.net.preferIPv4Stack=false -Djava.net.preferIPv6Addresses=true
```
Enable dynamic property updates

Allows dynamic changes to occur to several of the instance’s properties, without requiring you to shut down the instance.

This property is disabled by default.

**Note:** Any property that is dynamic is labeled as such in this document.

<table>
<thead>
<tr>
<th>Environment variables</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Environment</strong></td>
</tr>
<tr>
<td>Allows you to add and delete environment variables.</td>
</tr>
<tr>
<td>Environment variables that you add to the list are scoped to the process in which the NameServer executes.</td>
</tr>
</tbody>
</table>
Configuring remote instances

A remote NameServer instance is a reference to a NameServer that runs on some other host. By defining a remote instance, you allow a Unified Broker running on the local host to use the NameServer running on the remote host as its controlling NameServer. Thus, a remote NameServer configuration requires that you define:

- One local NameServer instance on the host where the NameServer runs
- One remote instance on each host where a Unified Broker uses the remote NameServer as its controlling NameServer

For example, consider that you are creating a NameServer instance on one system, which is named Sys1. The NameServer instance is actually physically running under an AdminServer on another system, which is named Sys2. The instance is remote for Sys1 and local for Sys2.

To create a remote instance, you need the host name and port number of Sys2 and its NameServer. You then use that information to create a remote NameServer on Sys1.

To configure a remote NameServer instance:

1. Configure at least one local NameServer instance on the remote host.
2. Using the management console, connect to the AdminServer on a host where you want to define the remote instance, and filter or search for the NameServer broker instances for that host.
3. Click the Edit icon for an existing remote NameServer instance. The NameServer details page appears.

   You can also create a new NameServer, specifying its name and indicating that its location is remote. For details, see the “Creating a NameServer” section on page 126.

4. In the Command and control section of the details page, click Configuration. The NameServer Configuration page appears.
5. Click Edit. The Location default is local. If you are editing an existing instance or creating a new instance, change the location to remote.
6. Set or change the host name to a machine where you have a NameServer defined as a local instance. Set the port number to the port on which the local instance you want to reference is listening.
7. Click Save.
Editing a NameServer configuration

You can edit a NameServer configuration from the management console.

To edit a NameServer configuration:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the NameServer broker instance whose configuration you want to edit.
3. Click the NameServer broker instance. The NameServer details page appears.
4. In the Command and control section of the page, click Configuration. The NameServer Configuration page appears.
5. Click Edit. (For details about the NameServer properties, see Table 28.)
6. Make the configuration changes, and then click Save. Any changes you make are saved in the ubroker.properties file.

Setting or deleting NameServer environment variables

You can set or delete environment variables for a NameServer.

To set NameServer environment variables:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the NameServer broker instance whose environment variables you want to set.
3. Click the NameServer broker instance. The NameServer details page appears.
4. In the Command and control section of the page, click Configuration. The NameServer Configuration page appears.
5. Click Edit.
6. Click Environment Variables.
7. Enter each variable name and then provide its value in the format name=value.
8. Click Save when you finish adding the variables and their values.
To delete NameServer environment variables:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the NameServer broker instance whose environment variables you want to delete.

3. Click the NameServer broker instance. The NameServer details page appears.

4. In the Command and control section of the page, click Configuration. The NameServer Configuration page appears.

5. Click Edit.

6. Click Environment Variables.

7. Delete the variable and value pair (or pairs).

8. Click Save when you finish.
Creating a NameServer

OpenEdge Management and OpenEdge Explorer provide a sample configuration NameServer instance (NS1) to use as a default.

You can create two types of NameServer instance:

- **Local** — Where the instance definition resides on, and the NameServer itself executes on, the selected host.
- **Remote** — Where the instance definition is a reference to a NameServer that resides and executes on a machine that is remote from the selected host.

**To create a NameServer:**

1. From the drop-down for **Resources** on the management console menu, click **New OpenEdge Resource** → **NameServer**. The **NameServer Configuration** page appears.

2. Type the name of the new NameServer in the field provided.

**Note:** The NameServer name is case sensitive and can include any character except a period (.) or square brackets ([ ]). The name must be unique among all configured NameServer names.

3. Click **Save**. The **NameServer Configuration** page appears, allowing you to configure the NameServer’s properties. (For details about the properties, see **Table 28**.)

Once you create a NameServer instance, you cannot change the location setting unless you delete and recreate the instance.

**Note:** If you are defining a remote NameServer instance, set its host name and port number properties to reference a NameServer that is defined as a local NameServer on some other machine in your network. For details, see the “Configuring remote instances” section on page 123.
Deleting a NameServer

You can delete any NameServer instance from the management console as long as neither of the following conditions is true:

- The NameServer instance is running.
- One or more Unified Broker instances reference this NameServer instance as their controlling NameServer.

To delete the NameServer, you must stop the NameServer instance and delete all Unified Broker instances that reference it.

To delete a NameServer instance:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the NameServer broker instance you want to delete.
3. Click the NameServer broker instance. The NameServer details page appears.
4. Click Delete, and then click OK to confirm the deletion.
Chapter 4: Configuring NameServers

Configuring fault tolerance and load balancing

With the NameServer, you have access to connection-level fault tolerance, server-level fault tolerance, and load balancing.

Connection-level fault tolerance

Multiple NameServers can cooperate to resolve a client connection. If one NameServer becomes unavailable, another can take its place, thus providing connection-level fault tolerance. When you configure and use NameServers with connection-level fault tolerance, each requesting client uses the Unified Broker connection provided by the first responding NameServer and ignores the rest.

You can configure two types of connection-level fault tolerance:

- NameServer replication
- NameServer neighbors

NameServer replication

To use NameServer replication, you first configure multiple NameServers on one subnet to listen on the same UDP port. You then have each Unified Broker instance register with its controlling NameServer and have each client contact its connecting NameServer using the subnet broadcast address and the same UDP port number. Thus, Unified Broker registrations are automatically replicated in each NameServer, and client connection requests are handled by each NameServer. The first replicated NameServer to respond to a client request provides the Unified Broker connection.

NameServer neighbors

To use NameServer neighbors, you configure multiple NameServers on one or more subnets. For each such NameServer, you specify the instance name of each of the other NameServers that you want to act as a NameServer neighbor. When a client makes a connection request to a specific NameServer, if this NameServer determines that it cannot satisfy the request (usually because it does not recognize the specified application service), it passes the request to all of its NameServer neighbors. The first NameServer neighbor to respond to a client request provides the Unified Broker connection.

You can configure and use NameServer replication and NameServer neighbors together to provide the maximum in connection-level fault tolerance.

Server-level fault tolerance

If you have the load-balancing option installed, you can configure multiple Unified Broker instances of a given type to support the same application service and share the same controlling NameServer. In this configuration, the controlling NameServer distributes client connection requests randomly across Unified Broker instances. If one Unified Broker instance becomes unavailable to handle client requests, another is available to take its place, thus providing server-level fault tolerance.
Load balancing

If you specify a priority weight factor for each Unified Broker instance, the NameServer distributes the connection requests proportionately, thus balancing the load. For example, suppose that an Inventory service is supported by WebSpeed brokers WS1, WS2, and WS3, which have weight factors of 20, 20, and 60, respectively. The selection algorithm used by the NameServer guarantees that WS1 and WS2 are selected 20% of the time and WS3 is selected 60% of the time.

Note: You do not assign application services or weight factors to NameServers, but rather to Unified Broker instances. Each Unified Broker instance passes its weight factor and list of application service names to its controlling NameServer when it registers with the NameServer.

Location transparency

When a Unified Broker instance is started, it registers with its controlling NameServer the application services that it supports. (An application service is an arbitrary designation for the business function that a Unified Broker instance provides.) The NameServer maintains a separate list of application services for each Unified Broker type. When a client of a certain type requests a connection to an application service, the NameServer redirects the connection request to a Unified Broker instance of the corresponding type that supports the requested application service. Because the client always connects indirectly to a Unified Broker through the NameServer, the client does not need to know the location of the Unified Broker instance. It needs to know only the location of the NameServer and the name of a valid application service that it wants.
Starting or Stopping a NameServer

From the management console, you can directly start or stop a local NameServer on the machine where it runs. You can also set a NameServer to start automatically when the AdminServer starts. For details, see the “Starting a NameServer automatically” section on page 130.

Note: To start or stop a remote NameServer, access the corresponding local NameServer instance selected in the container of the host on which it runs.

To start or stop a NameServer, refer to the Starting or Stopping OpenEdge resources section in OpenEdge Management: Servers, DataServers, Messengers, and Adapters

Starting a NameServer automatically

You can set a NameServer to start automatically when the AdminServer starts.

To start a NameServer automatically:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the NameServer broker instance you want to start automatically.
3. Click the NameServer broker instance. The NameServer details page appears.
4. In the Command and control section of the page, click Configuration. The NameServer Configuration page appears.
5. Click Edit.
6. Click the General tab.
7. Select the Auto start option by clicking in the check box.
8. Click Save.
Viewing the status of a NameServer

You can view the status of any running NameServer instance through the management console.

Each NameServer instance provides a separate status view. You can monitor the status of only those NameServer instances running locally on a selected host.

To view the status of a NameServer instance:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the NameServer broker instance whose status you want to view.

3. Click the NameServer broker instance. The **NameServer** details page appears.

4. In the **Operations views** section of the page, click **Status**.

A status summary and summary details appear.

The summary, as described in Table 29, provides collective property and statistics information for the NameServer and the OpenEdge Unified Broker instances that it coordinates.

The details, as described in Table 30, provide a table of status information for all Unified Broker instances that are registered with the NameServer. Each row in the table contains information for a single Unified Broker instance.

**Table 29: NameServer status summary properties**

<table>
<thead>
<tr>
<th>Property or statistic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>The name of the host machine</td>
</tr>
<tr>
<td>Port</td>
<td>The UDP port number that the NameServer listens to</td>
</tr>
<tr>
<td>Timeout</td>
<td>The Unified Broker keep alive timeout value for the NameServer</td>
</tr>
<tr>
<td>Start time</td>
<td>When the NameServer started</td>
</tr>
<tr>
<td>Number of AppServices</td>
<td>The number of application services that the NameServer is coordinating</td>
</tr>
<tr>
<td>Number of Brokers</td>
<td>The number of Unified Broker instances the NameServer is coordinating</td>
</tr>
<tr>
<td>Total client requests received</td>
<td>The number of client requests the NameServer has received</td>
</tr>
<tr>
<td>Total client requests rejected</td>
<td>The number of client requests the NameServer has rejected</td>
</tr>
</tbody>
</table>
Table 30: NameServer status detail properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AppService Name</td>
<td>The name of an application service registered with the NameServer.</td>
</tr>
<tr>
<td>UUID</td>
<td>The unique number for the Unified Broker instance.</td>
</tr>
<tr>
<td>Name</td>
<td>The Unified Broker instance name.</td>
</tr>
<tr>
<td>Host</td>
<td>The name of the host where the Unified Broker instance resides.</td>
</tr>
<tr>
<td>Port</td>
<td>The TCP/IP port number used by the Unified Broker instance to handle client connection requests.</td>
</tr>
<tr>
<td>Weight</td>
<td>The priority weight assigned to the Unified Broker instance for load balancing.</td>
</tr>
<tr>
<td>Timeout</td>
<td>The number of seconds that must pass between keep-alive messages sent by the Unified Broker instance before the NameServer assumes that the instance is no longer available. Once the timeout expires, the NameServer automatically unregisters the Unified Broker instance.</td>
</tr>
</tbody>
</table>
Handling UDP and firewall issues

Whenever an AppServer Internet Adapter (AIA) or WebSpeed Messenger instance communicates with a NameServer, the instance is assigned a UDP port. The NameServer uses this port to send a response message to the instance. By default this port is dynamically assigned by the operating system within the UDP port range of 1024 to 65535.

If there is a firewall between the instance and the NameServer, the default setting requires that all UDP ports in that range must be open from the network where the NameServer exists to the network where the AIA or Messenger instance exists. Otherwise, communication between the instance and the NameServer cannot occur.

However, the only port that must be open in the firewall is the controlling NameServer port for UDP traffic going from the network containing the AIA or Messenger instance to the network containing the NameServer. To increase security, you can limit the range of ports from which an AIA or Messenger instance can be assigned an UDP port. The NameServer Client Port Minimum (minNSClientPort) and the NameServer Client Port Maximum (maxNSClientPort) properties can be used to reduce the number of UDP ports that you must open in the firewall. When you set these properties, OpenEdge chooses the first port in that range that is not currently used by some other application for UDP broadcast communications. If no port in this range is available, the default behavior is to retry the ports for a period of time.
Chapter 4: Configuring NameServers

Viewing the NameServer log file

The NameServer log file viewer allows you to examine the contents of log files through an HTML interface.

For details, see the section about using the NameServer log file viewer in *OpenEdge Management: Servers, DataServers, Messengers, and Adapters.*
You can use OpenEdge Management or OpenEdge Explorer to configure the properties of OpenEdge resources that you are monitoring, such as AppServers or AppServer Internet Adapters. Any configuration changes that you make to either of these resources in OpenEdge Management or OpenEdge Explorer are automatically reflected in the configuration property file `ubroker.properties`.

This chapter provides the following information about working with AppServer and AppServer Internet Adapter configuration properties:

- **AppServer configuration and administration**
- **For additional AppServer information**
- **AppServer Internet Adapter configuration and administration**
AppServer configuration and administration

With the AppServer, you can make ABL applications available to multiple machines over a network.

You can use OpenEdge Management or OpenEdge Explorer to configure new AppServer instances and administer the configurations of existing instances. The AppServer installation provides one sample AppServer (asbroker1). You can use this sample component as a template for creating and configuring additional AppServers. Each AppServer is called an instance.

Understanding AppServer configuration

Each AppServer consists of an AppServer broker and one or more AppServer agents. An AppServer agent is a single ABL Virtual Machine (AVM) instance running on an AppServer. Each AppServer agent executes ABL procedures on behalf of AppServer clients, including ABL and Open Clients (Java, .NET, and Web services clients).

An AppServer can support clients using two different session models, which are dependent on its operating mode:

- **Session-managed** — Includes stateless, state-aware, and state-reset operating modes. These operating modes maintain a single session and connection between the AppServer and the client for all client requests.

- **Session-free** — Includes only the state-free operating mode. This operating mode maintains no session or connection between the AppServer and the client. In this operating mode, the AppServer treats any request from a client as independent from all other requests. As a result, some SESSION handle attributes in the ABL application have no valid meaning, or they provide different information.

The AppServer supports these session models for supported AppServer clients.

For more information, see *OpenEdge Application Server: Developing AppServer Applications*. 
AppServer features

The AppServer broker handles the following activities:

- Optionally registers with a NameServer the application services that it provides for access by clients
- Manages connections between clients and a pool of AppServer agents
- Maintains the status of each AppServer agent in its pool and dynamically scales the number of servers according to changing demand
- When configured for stateless or state-free operating mode, dispatches remote requests to AppServer agents
- When configured for state-reset or state-aware operating mode, connects each client to a dedicated AppServer agent that handles all remote requests for that client until the connection is terminated

Working with the AppServer

To work with an AppServer instance, double-click the AppServer folder in the management console’s list frame. There is one pre-defined default AppServer (asbroker1).

You can perform the following AppServer actions:

- Configuring an AppServer
- Creating an AppServer
- Starting and stopping an AppServer
- Adding or trimming AppServer agents
- Viewing the status of an AppServer
- Viewing the AppServer log files
Chapter 5: Configuring AppServers and AppServer Internet Adapters

AppServer configuration properties

An AppServer starts up by using the configuration information stored in the `ubroker.properties` file. The file contains the property settings for all server resources on the host where OpenEdge is installed.

The `ubroker.properties` file

The `ubroker.properties` file consists of a hierarchical structure of configuration entities, in which parent entities provide configuration information that you can override or extend in each child entity. Each configuration entity has a name that begins the entity definition, and the definition contains configuration settings for one or more product instances.

For example, property settings for an instance of an AppServer are determined hierarchically, as shown in Table 31.

<table>
<thead>
<tr>
<th>Configuration entity name</th>
<th>Configuration entity function</th>
</tr>
</thead>
<tbody>
<tr>
<td>[UBroker]</td>
<td>Defines default property settings for various OpenEdge brokers, including all AppServer brokers.</td>
</tr>
<tr>
<td>[UBroker.AS.product-instance-name]</td>
<td>Defines property settings for this instance of an AppServer. The <code>ubroker.properties</code> file can contain several of these entities, each with a unique <code>product-instance-name</code>.</td>
</tr>
</tbody>
</table>

Parent entities provide default values for all of their child entities. However, at any child level a redefinition of any value supersedes the default value of its parent. All children, from their definition level down, inherit this new value.

Like the `ubroker.properties` file, a similar file, `conmgr.properties`, stores all the properties for each instance of an OpenEdge database. For more information about OpenEdge Management and OpenEdge Explorer and the database configuration properties, see Chapter 2, "Configuring OpenEdge Databases."

The `ubroker.properties` file resides in the properties subdirectory of the OpenEdge install directory. The file is editable and any changes that you make to it are automatically reflected in the management console.

Note: Although making manual edits to the `ubroker.properties` file is possible, Progress Software recommends that you use the Mergeprop utility or the management console to make property changes. For more information about the Mergeprop utility, see OpenEdge Getting Started: Installation and Configuration.
Configuring an AppServer

From the management console, you can view or modify an AppServer broker’s default configuration properties in the following overall categories:

- Broker
- Agent
- SSL
- Messaging
- Environment variables

You can also create a new AppServer instance, view an instance’s status, delete an instance you no longer need, or view the instance’s log files.

Each of these tasks is described in the sections that follow.

Viewing or modifying AppServer broker properties

You can view or modify AppServer broker properties in the following categories: general properties, owner information, controlling NameServer information, AppService Name List information, logging setting details, and advanced features.

To view or modify AppServer broker properties:

1. From the OpenEdge Management console menu, click Resources. All resources managed by your console appear in the grid frame.

2. Filter or search for the AppServer broker instance whose properties you want to view or modify. For example, type the default AppServer broker name asbroker1 in the filter field or select AppServer as the selection in the Type drop-down.

3. From the list of AppServer broker’s, click the AppServer broker whose properties you want to view or modify. The AppServer details page appears.

4. In the Command and control section of the page, click Configuration. The AppServer Configuration page appears with the Broker properties in focus.

From this page, you can:

- View a read-only display of the AppServer broker properties as described in Table 32 through Table 37. All of these tables follow this procedure.

- Click Edit to modify the broker properties,. For details, see the “Editing an AppServer configuration” section on page 163.
### Table 32: AppServer Broker General properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto start</td>
<td>Indicates whether the server automatically starts when the controlling AdminServer starts.</td>
</tr>
<tr>
<td>Port number</td>
<td>The number of the TCP/IP port that the server broker listens on.</td>
</tr>
<tr>
<td>Working directory</td>
<td>Your working directory. The default is the working directory set during OpenEdge installation.</td>
</tr>
<tr>
<td>Operating mode</td>
<td>The operating mode for this server.</td>
</tr>
</tbody>
</table>

### Table 33: AppServer Broker Owner Information properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Username</td>
<td>The username of an account that has system-administrative rights.</td>
</tr>
<tr>
<td>Group name</td>
<td>(UNIX only) The name of the group.</td>
</tr>
<tr>
<td>Password</td>
<td>The username account password in Windows.</td>
</tr>
<tr>
<td>Password Confirm</td>
<td>Confirmation of the username account password. This field appears only if you are modifying the properties. Note that this field does not appear if the AdminServer is running on a non-Windows host machine.</td>
</tr>
</tbody>
</table>

### Table 34: AppServer Broker Controlling NameServer properties (1 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Register with NameServer</td>
<td>Indicates whether or not to register the AppServer with a controlling NameServer.</td>
</tr>
<tr>
<td>Controlling NameServer</td>
<td>The name of the NameServer with which this AppServer registered. If you did not select to register the broker with a NameServer, the field is disabled.</td>
</tr>
</tbody>
</table>
Registration mode

How the broker specifies its hostname if it is registering with a controlling NameServer. This hostname information is passed onto a client application when it attempts to connect to an application service that the AppServer broker supports. The choices are:

- **Register-IP** — (Default) This setting is the most efficient mechanism and can be used in most cases. It registers with the IP address of the machine where the broker is located.
- **Register-LocalHost** — The broker registers with the hostname of the machine that it runs on. Use this setting when the broker runs on a machine with a single hostname and more than one IP address.
- **Register-HostName** — The broker registers with the values specified in the HostName property. Use this setting when your clients need a fully qualified host name to connect to a broker in a different DNS domain.

Registration host name

The host name alias of the local system.

Table 35: **AppServer Broker AppService Name List properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application service names</td>
<td>A list to which you can add or delete application service names</td>
</tr>
<tr>
<td>Supports default service</td>
<td>Indicates whether or not the AppServer broker supports the default service</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>Broker log filename</td>
<td>The broker log filename.</td>
</tr>
</tbody>
</table>
| Broker logging level (dynamic property) | A value that specifies the amount of information that is written to the broker log. The following values are available:  
- **None** — Log no entries. This is equivalent to turning logging off.  
- **Error Only** — Log OpenEdge error messages. This includes all error messages and is unrelated to the entry types specified. Errors continue to be logged at all higher levels.  
- **Basic** — Broker logging entry types determine the logged information. Each entry type generates at least some output. This is the default.  
- **Verbose** — Logging entry types determine the logged information, but it is typically more information than **Basic**.  
- **Extended** — Logging entry types determine the logged information, but it is typically more information than **Verbose**.  
You can update this property dynamically. Any changes will affect all current and new brokers and/or agents. |
| Append to broker log file | Indicates whether or not a new broker log file should be created when the AppServer is started. |
**Broker logging entry types**  
(dynamic property)  

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| A comma-separated list of the following valid entry types for agent logging:  
  - **UBroker.AutoTrim** — Logs messages about automatic agent thread trimming by the broker based on the **Auto trim timeout** property. Information is logged at logging level **Basic** and higher.  
  - **UBroker.Basic** — Logs messages about broker and agent startup and shutdown at the following logging levels:  
    - **Basic** — Logs NameServer registration and connections from clients.  
    - **Verbose** — Logs information about all the property values read from the `ubroker.properties` file for that broker instance, and more information on the connection from clients.  
  - **UBroker.ThreadPool** — Logs messages about the pool of threads managed by the broker. These threads are used to control the client requests and the agent processes. This entry type logs messages when adding, removing, and communicating with these threads. Information is logged at logging level **Basic** and higher.  
  - **UBroker.ClientFSM** — Logs messages about the state of the client thread as it processes the requests, from the moment it receives a connection request to the end of the request. Information is logged at logging level **Basic** and higher.  
  - **UBroker.ServerFSM** — Logs messages about the state of the agent thread as it processes the requests, from the moment it receives a connection request to the end of the request. Information is logged at logging level **Basic** and higher.  
  - **UBroker.ClientMsgStream** — Logs messages about messages exchanged between the broker and client threads in binary format. Information is logged at logging level **Basic** and higher.  

You can update this property dynamically. Any changes will affect all current and new brokers and/or agents.  

Turn on the following logging entry types only at the request of Progress Software Corporation Technical Support:  
  - **UBroker.ClientFSM** — Logs messages about the state of the client thread as it processes the requests, from the moment it receives a connection request to the end of the request. Information is logged at logging level **Basic** and higher.  
  - **UBroker.ServerFSM** — Logs messages about the state of the agent thread as it processes the requests, from the moment it receives a connection request to the end of the request. Information is logged at logging level **Basic** and higher.  
  - **UBroker.ClientMsgStream** — Logs messages about messages exchanged between the broker and client threads in binary format. Information is logged at logging level **Basic** and higher.
Broker logging entry types

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UBroker.ServerMsgStream</td>
<td>Logs messages about messages exchanged between the broker and agent threads in binary format. Information is logged at logging level Basic and higher.</td>
</tr>
<tr>
<td>UBroker.ClientMsgQueue</td>
<td>Logs messages about the client thread as it processes queued messages. Information is logged at logging level Basic and higher.</td>
</tr>
<tr>
<td>UBroker.ServerMsgQueue</td>
<td>Logs messages about the agent thread as it processes queued messages. Information is logged at logging level Basic and higher.</td>
</tr>
<tr>
<td>UBroker.ClientMemTrace</td>
<td>Traces messages being processed by the client threads. Information is logged at logging level Basic and higher.</td>
</tr>
<tr>
<td>UBroker.ServerMemTrace</td>
<td>Traces messages being processed by the agent threads. Information is logged at logging level Basic and higher.</td>
</tr>
<tr>
<td>UBroker.Stats</td>
<td>Logs statistics about the number of requests, plus maximum, minimum, and average request wait and duration times. Information is logged at logging level Basic and higher.</td>
</tr>
<tr>
<td>UBroker.All</td>
<td>Logs all Unified Broker entry types.</td>
</tr>
</tbody>
</table>

You can specify a separate logging level for each log entry type and use a wildcard (*) to specify multiple entry types. For example:

UBroker.Basic:3, UBroker.ThreadPool:4, UBroker.Client*

For details, see the information about client logging in *OpenEdge Deployment: Startup Command and Parameter Reference*. 

---

OpenEdge® Management and OpenEdge Explorer: Configuration
<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Broker log file threshold size</td>
<td>A value of 0 or a value between 500,000 and 2,147,483,647, where 0 means there is no limit on the log file size other than what the operating system imposes.</td>
</tr>
<tr>
<td>Maximum number of broker log files</td>
<td>The number of rolled-over log files to keep. The value can be 0 or a value between 2 and 999999, where 0 means there is no limit on the number of broker log files to keep. The specified number represents the maximum number of log files to keep on disk at any time, including the current log file being written to. When the file becomes equal to or greater than the Broker log file threshold size, the client process renames it and creates a new log file. The file is renamed as follows, where ###### is a number starting at 000001 and increasing to 999999, after which it rolls back over to 000001: filename.#####.extension</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Maximum client instances</strong></td>
<td>The maximum number of client connections that the broker can support concurrently. The default is 512, which is high enough to ensure that the number of client connections is virtually unlimited. Realistically, however, the system-level resources needed to support an unlimited number of clients might be exhausted before this limit is reached. If you see broker log entries that contain Exception Messages for &quot;out of space&quot; or &quot;OutOfMemory&quot; you might need to reduce the maximum number of client connections to a more reasonable value. Note, however, that lowering this value can cause some client requests to be rejected (&quot;Exceeded Max Clients&quot;). Configuring and starting multiple brokers to handle higher client loads alleviates the problem.</td>
</tr>
<tr>
<td><strong>Priority weight (0-100)</strong></td>
<td>An integer value between 0 and 100 that influences the share of the workload that the selected AppServer receives. The larger the value, the heavier the load that is distributed to the server. The NameServer distributes client requests across all AppServers that have the same application service in proportion to the <strong>Priority weight</strong> value. The default is 0. You can update this property dynamically. Any changes will affect all current and new brokers.</td>
</tr>
<tr>
<td><strong>Registration retry (in seconds)</strong></td>
<td>After registering with its controlling NameServer, the AppServer periodically sends &quot;keep-alive&quot; messages to the NameServer to let the NameServer know that it is still active. The <strong>Registration retry</strong> value is the number of seconds that pass between &quot;keep-alive&quot; messages. The default is 30.</td>
</tr>
<tr>
<td><strong>Server startup timeout</strong></td>
<td>Specifies the amount of time, in seconds, that the broker waits for an active agent to become available before starting a new agent process. The default is 3. You can update this property dynamically. Any changes will affect only new agents that start after the property value has change; existing (running) brokers and agents are not updated.</td>
</tr>
</tbody>
</table>

Table 37: **AppServer Broker Advanced Features properties** (1 of 4)
<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request timeout</td>
<td>Specifies the amount of time, in seconds, that the broker will wait for an agent to become available for processing a request.</td>
</tr>
<tr>
<td>(dynamic property)</td>
<td>The broker waits for this period only if the <strong>Maximum servers</strong> setting has been reached. After the timeout has expired, the client receives a &quot;no servers available&quot; error message.</td>
</tr>
<tr>
<td></td>
<td>The default is 15.</td>
</tr>
<tr>
<td></td>
<td>You can update this property dynamically. Any changes will affect all current and new brokers and/or agents.</td>
</tr>
<tr>
<td>Auto trim timeout</td>
<td>The amount of time, in seconds, that the broker waits before automatically trimming the number of running agents.</td>
</tr>
<tr>
<td>(dynamic property)</td>
<td>The broker keeps track of the maximum number of agents that are simultaneously busy during the interval you specify. At the end of the interval, the broker attempts to trim the number of agents to match the <strong>Maximum servers</strong> for the interval, or the <strong>Minimum servers</strong>, whichever is greater.</td>
</tr>
<tr>
<td></td>
<td>You can disable this feature by setting the timeout to zero (0). The default <strong>Server Auto-trim Interval</strong> is 1800 seconds.</td>
</tr>
<tr>
<td>TCP/IP version</td>
<td>Internet Protocol for network communication:</td>
</tr>
<tr>
<td></td>
<td>• <strong>IPv4</strong> — Accepts only IPv4 connections</td>
</tr>
<tr>
<td></td>
<td>• <strong>IPv6</strong> — Allows IPv4 and IPv6 connections using mapped address (where supported)</td>
</tr>
<tr>
<td>TCP/IP version</td>
<td>The default is IPv4.</td>
</tr>
<tr>
<td></td>
<td>You must also set the jvmArgs property in the ubroker.properties file to override the default behavior of the JVM for IPv6 connections.</td>
</tr>
<tr>
<td>TCP/IP version</td>
<td><code>jvmArgs=-Djava.net.preferIPv4Stack=false -Djava.net.preferIPv6Addresses=true</code>.</td>
</tr>
<tr>
<td>jvmargs</td>
<td>Java System Properties.</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Enable dynamic property updates</td>
<td>Allows dynamic changes to occur to several of the instance’s properties, without requiring you to shut down the instance.</td>
</tr>
<tr>
<td></td>
<td>This property is disabled by default.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Any property that is dynamic is labeled as such in this document.</td>
</tr>
<tr>
<td>AppServer Keepalive</td>
<td>Allows the AppServer to recognize that a client bound to it is no longer connected.</td>
</tr>
<tr>
<td>serverASK Activity Timeout</td>
<td>Specifies the number of seconds with no activity that the AppServer waits before sending the client a message requesting that the connection be maintained.</td>
</tr>
<tr>
<td></td>
<td>The property is used only if the AppServer Keepalive property is set to <code>allowServerASK</code>.</td>
</tr>
<tr>
<td>serverASK Response Timeout</td>
<td>Specifies the number of seconds with no activity after the AppServer sends the client a message (requesting that the connection be maintained) before the client is disconnected.</td>
</tr>
<tr>
<td></td>
<td>This property is used only if the AppServer Keepalive property is set to <code>allowServerASK</code>.</td>
</tr>
<tr>
<td>Enable debugging through the broker</td>
<td>Specifies whether or not the debugger is enabled; select the check box to enable it.</td>
</tr>
<tr>
<td>(dynamic property)</td>
<td>If you disable this property while the debugger is connected, the debugger connection terminates.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> The Broker debugger port number property must also be set to a valid value for the debugger to be properly enabled.</td>
</tr>
<tr>
<td></td>
<td>This property is disabled by default.</td>
</tr>
<tr>
<td>Broker debugger port number</td>
<td>Specifies the port number that the broker debugger will use when it starts. You can choose any valid value; if the port you specify is already in use, the debugger service will fail to start.</td>
</tr>
<tr>
<td>(dynamic property)</td>
<td>If you change the port number while the debugger service is active, the debugger service will be shut down and restarted using the new port number.</td>
</tr>
<tr>
<td></td>
<td>The default port number is 3099.</td>
</tr>
</tbody>
</table>
Table 37: AppServer Broker Advanced Features properties  

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| Enable SSL for broker debugger                               | Specifies whether the broker debugger should accept only SSL/TLS-encrypted connections. If you enable this property, the Progress Developer Studio debugger will be required to establish an SSL/TLS session.  
If you enable or disable the property while the debugger service is active, the debugger will be disconnected. The debugger service will be shut down and then restarted using the new selection.  
This option is disabled by default. |
| Use the broker's private key/digital certificate for debugger  | When enabled, indicates that, when using SSL, the broker debugger service should use the same private key as used by the AppServer agents.  
If this property is enabled, then the Debugger private key/digital certificate alias name and Password to access the debugger’s key/certificate property settings are ignored.  
This property is enabled by default. |
| Debugger private key/digital certificate alias name           | The name of the private key that should be used for the debugger service when the agent's private key is not being used.  
This property must be defined if Enable SSL for broker debugger is enabled. |
| Password to access the debugger’s key/certificate             | The password required to access the private key specified by Debugger private key/digital certificate alias name.  
The default for this property is blank. |
| Broker debugger passphrase                                    | An optional property that represents the passphrase required by the debugger to connect to the debugger service in the broker.  
If the debugger is connected to the debugger service when the passphrase is changed, the debugger remains connected. The next time the debugger attempts to connect to the debugger service, the new password is required.  
The default for this property is blank. |
| Broker debugger passphrase Confirm                             | Confirms the passphrase provided in Broker debugger passphrase. |
| Publish directory                                             | The name of the local directory where you want r-code to be placed when you are publishing from Progress Developer Studio.  
If you are publishing remotely from Progress Developer Studio, you must set this property. |
Chapter 5: Configuring AppServers and AppServer Internet Adapters

Viewing or modifying AppServer Agent properties

You can modify the following Agent properties for the AppServer Broker: general properties, logging setting details, pool range, and advanced features.

To view or modify AppServer Agent properties:

1. From the OpenEdge Management console menu, click Resources. All resources managed by your console appear in the grid frame.

2. Filter or search for the AppServer broker instance whose properties you want to view or modify. For example, for the default AppSever broker name type asbroker1 in the filter field or select AppSever from the Type drop-down.

3. From the list of AppServer broker instances, click the AppServer broker whose properties you want to view or modify. The AppServer details page appears.

4. In the Command and control section of the page, click Configuration. The AppServer Configuration page appears with the Broker properties in focus.

5. Click the Agent tab.

From this page, you can do the following:

- View a read-only display of the AppServer Agent properties as described in Table 38 through Table 41. All of these tables follow this procedure.

- Click Edit to modify the broker properties. For details, see the "Editing an AppServer configuration" section on page 163.
Table 38: AppServer Agent General properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Server executable file</strong></td>
<td>Either the default agent executable pathname or a different agent executable file of your own choosing. You generally need to specify a nondefault value only if you generated a new executable using the OEBUILD utility. You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated.</td>
</tr>
<tr>
<td><strong>Server startup parameters</strong></td>
<td>The OpenEdge startup parameters that you want to specify to start each agent. You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated. <strong>Note:</strong> It is recommended to set the Stack size startup parameter value as <code>-s 128</code> in the Server startup parameters field, when working with databases having large schemas.</td>
</tr>
<tr>
<td><strong>PROPATH</strong></td>
<td>The search path that agents use to locate ABL procedures that they execute. Specify a list of directory pathnames in the form of an OpenEdge PROPATH. Make sure that you copy all of your AppServer procedures to one or more of the directories in the PROPATH. You can enter the pathname directly.</td>
</tr>
<tr>
<td><strong>Minimum port number</strong></td>
<td>The minimum TCP/IP port number from a range that each agent process can listen on. When each agent starts, it allocates a port that is not being used from the specified range up to the Maximum port number setting. You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated.</td>
</tr>
</tbody>
</table>
Chapter 5: Configuring AppServers and AppServer Internet Adapters

Table 38:  AppServer Agent General properties  (2 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| Maximum port number                    | The maximum TCP/IP port number from a range that each agent can listen on. When each agent starts, it allocates a port that is not being used from the specified range down to the **Minimum port number** setting.  
You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated. |
| Flush statistical data                  | How often the data is flushed by the agent to the broker. This is expressed as the number of remote procedure calls between flushes.  
You can update this property dynamically. Any changes will affect all current and new brokers and/or agents. |

Table 39:  AppServer Agent Logging Setting properties  (1 of 4)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server log filename</td>
<td>The log file name.</td>
</tr>
</tbody>
</table>
| Server logging level                   | A value that specifies the amount of information to be written to the agent log. Each logging level name has the indicated numeric value. You can select from the following values in the drop-down list:  
- **None** — Log no entries. This is equivalent to turning logging off.  
- **Error Only** — Log OpenEdge error messages. This includes all error messages and is unrelated to the entry types specified. Errors continue to be logged at all higher levels.  
- **Basic** — Server logging entry types determine the logged information. Each entry type generates at least some output. This is the default.  
- **Verbose** — Logging entry types determine the logged information, but it is typically more information than **Basic**.  
- **Extended** — Logging entry types determine the logged information, but it is typically more information than **Verbose**.  
You can update this property dynamically. Any changes will affect all current and new brokers and/or agents. |
### Server logging entry types (dynamic property)

A comma-separated list of valid entry types for agent logging:

- **4GLTrace** — Turns on the logging for the execution of the following ABL statements: RUN, FUNCTION (user-defined functions), PUBLISH, and SUBSCRIBE. Logs information at Basic or higher.
- **ASDefault** — Combines the ASPlumbing and DB.Connects log entry types. This is the default value for AppServer agents.
- **ASPlumbing** — Turns on logging for different actions depending on the logging level specified:
  - **Basic** — Logs messages when an AppServer agent starts and stops, and when client processes connect and disconnect from the AppServer agent.
  - **Verbose** — Same as Basic, plus logging of the execution of the top level procedure (that is, the procedure that the client process asks the AppServer agent to execute on its behalf, but not any procedure that the top level procedure executes).
  - **Extended** — Same as Verbose, plus messages about agent state, and status messages about read and write socket operations. Do not turn this on unless directed to do so by Progress Software Technical Support.
- **DB.Connects** — Turns on logging of database connections and disconnections. The messages include database name and user ID number. Logs information at logging level Basic or higher.
- **DynObjects.DB** — Turns on logging for dynamic database-related objects (for example, TEMP-TABLE or DATASET) as they are created and destroyed. Logs information at Basic or Verbose.

### Append to server log file

Indicates whether or not a new agent log file should be created when the AppServer is started.

---

**Table 39: AppServer Agent Logging Setting properties (2 of 4)**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server logging entry types</td>
<td>A comma-separated list of valid entry types for agent logging:</td>
</tr>
<tr>
<td></td>
<td>- 4GLTrace — Turns on the logging for the execution of the following ABL statements: RUN, FUNCTION (user-defined functions), PUBLISH, and SUBSCRIBE. Logs information at Basic or higher.</td>
</tr>
<tr>
<td></td>
<td>- ASDefault — Combines the ASPlumbing and DB.Connects log entry types. This is the default value for AppServer agents.</td>
</tr>
<tr>
<td></td>
<td>- ASPlumbing — Turns on logging for different actions depending on the logging level specified:</td>
</tr>
<tr>
<td></td>
<td>- Basic — Logs messages when an AppServer agent starts and stops, and when client processes connect and disconnect from the AppServer agent.</td>
</tr>
<tr>
<td></td>
<td>- Verbose — Same as Basic, plus logging of the execution of the top level procedure (that is, the procedure that the client process asks the AppServer agent to execute on its behalf, but not any procedure that the top level procedure executes).</td>
</tr>
<tr>
<td></td>
<td>- Extended — Same as Verbose, plus messages about agent state, and status messages about read and write socket operations. Do not turn this on unless directed to do so by Progress Software Technical Support.</td>
</tr>
<tr>
<td></td>
<td>- DB.Connects — Turns on logging of database connections and disconnections. The messages include database name and user ID number. Logs information at logging level Basic or higher.</td>
</tr>
<tr>
<td></td>
<td>- DynObjects.DB — Turns on logging for dynamic database-related objects (for example, TEMP-TABLE or DATASET) as they are created and destroyed. Logs information at Basic or Verbose.</td>
</tr>
<tr>
<td>Append to server log file</td>
<td>Indicates whether or not a new agent log file should be created when the AppServer is started.</td>
</tr>
</tbody>
</table>
Table 39: AppServer Agent Logging Setting properties (3 of 4)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| Server logging entry types              | • DynObjects.XML — Turns on logging for dynamic XML-related objects (for example, x-document or x-noderef) as they are created and destroyed. Logs information at level Basic or Verbose. You can update this property dynamically. Any changes will affect all current and new brokers and/or agents.  
• DynObjects.Other — Turns on logging for dynamic objects that do not match the other dynamic object categories (for example, procedure or server socket) as they are created and destroyed. Logs information at Basic or Verbose.  
• ProEvents.UI.Char — Turns on logging of keystroke events for printable characters. Logs information at Basic or higher.  
• ProEvents.UI.Command — Turns on logging of keystroke events for nonprintable characters at the following logging levels:  
  – Basic — Logs nonprintable keystrokes and a subset of Windows GUI events  
  – Verbose — Logs nonprintable keystrokes and all Windows GUI events  
• ProEvents.Other — Turns on logging of COM, asynchronous, and server socket events. Logs information at Basic or higher, beginning with the events for which you have written triggers.  
• QryInfo — Turns on database query information logging at the following logging levels, with the same amount of information logged for each level:  
  – Basic — Logs static queries and initial dynamic queries.  
  – Verbose — Logs static queries and multiple instances of dynamic queries. |

(continued)
Table 39: AppServer Agent Logging Setting properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Server logging entry types</strong> (continued)</td>
<td>• <strong>SAX</strong> — Turns on logging for various stages of execution using the ABL SAX parser.</td>
</tr>
<tr>
<td></td>
<td>You can specify a separate logging level for each log entry type and use a wildcard (*) to specify multiple entry types. For example:</td>
</tr>
<tr>
<td></td>
<td>4GLTrace:3, ASPlumbing:2,DynObjects.*</td>
</tr>
<tr>
<td></td>
<td>For details, see the information about client logging in OpenEdge Deployment: Startup Command and Parameter Reference.</td>
</tr>
<tr>
<td><strong>Server log file threshold size</strong></td>
<td>A value of 0 or a value between 500,000 and 2,147,483,647, where 0 means there is no limit on the agent log file size other than what the operating system imposes.</td>
</tr>
<tr>
<td><strong>Maximum number of server log files</strong></td>
<td>The number of rolled-over log files to keep. The value can be 0 or a value between 2 and 999999, where 0 means there is no limit on the number of agent log files to keep. The specified number represents the maximum total number of log files to keep on disk at any time, including the current log file being written to.</td>
</tr>
<tr>
<td></td>
<td>When the file becomes equal to or greater than the <strong>Server log file threshold size</strong>, the client process renames it and creates a new log file. The file is renamed as follows, where ###### is a number starting at 000001 and increasing to 999999, after which it rolls back over to 000001:</td>
</tr>
<tr>
<td></td>
<td><code>filename.######.extension</code></td>
</tr>
<tr>
<td><strong>Server watch dog interval</strong></td>
<td>The interval (in seconds) for the server log file watchdog thread. If the <strong>Server log file threshold size</strong> property is specified, the broker will roll over the server log file when its size is larger than the logThreshold value.</td>
</tr>
</tbody>
</table>
Table 40: **AppServer Agent Pool Range properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial number of servers to start</td>
<td>The number of agents you want the broker to start initially.</td>
</tr>
<tr>
<td>Minimum servers</td>
<td>The minimum number of agents before the broker starts additional agents. If you trim the number of agents below this value, the server starts any additional agents needed to maintain the specified minimum when the next client connects.</td>
</tr>
<tr>
<td>Maximum servers</td>
<td>The maximum number of agents that this server can have running at the same time.</td>
</tr>
</tbody>
</table>

Table 41: **AppServer Agent Advanced Features properties** (1 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>4GL debugger enabled</td>
<td>Indicates if the AppServer remote debugging facility is enabled. If the facility is enabled, ABL client applications have the ability to step into remote AppServer procedures and WebSpeed procedures have the ability to step into remote AppServer procedures.</td>
</tr>
<tr>
<td>Activate procedure (dynamic property)</td>
<td>The name of an ABL procedure that executes before the AppServer executes a remote procedure request for a client connection that is unbound. <strong>Activate procedure</strong> is available only for an AppServer running in stateless operating mode. You can update this property dynamically. Any changes will affect all current and new agents.</td>
</tr>
<tr>
<td>Deactivate procedure (dynamic property)</td>
<td>The name of an ABL procedure that executes after the AppServer executes a remote procedure request for a client connection that is unbound. <strong>Deactivate procedure</strong> property is available only for an AppServer running in stateless operating mode. You can update this property dynamically. Any changes will affect all current and new agents.</td>
</tr>
<tr>
<td>Connect procedure (dynamic property)</td>
<td>The name of an ABL procedure that executes when a client attempts to connect to this AppServer. You can update this property dynamically. Any changes will affect all current and new agents.</td>
</tr>
<tr>
<td>Disconnect procedure (dynamic property)</td>
<td>The name of an ABL procedure that executes when a client disconnects from this AppServer. You can update this property dynamically. Any changes will affect all current and new agents.</td>
</tr>
</tbody>
</table>
### Table 41: AppServer Agent Advanced Features properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Startup procedure</strong></td>
<td>The name of an ABL procedure that executes when an AppServer agent starts up for this AppServer. The <em>Startup procedure</em> property is not available for an AppServer running in state-reset operating mode. You can update this property dynamically. Any changes will affect only new agents; existing (running) agents are not updated.</td>
</tr>
<tr>
<td>(dynamic property)</td>
<td></td>
</tr>
<tr>
<td><strong>Shutdown procedure</strong></td>
<td>The name of an ABL procedure that executes in an AppServer agent when the AppServer agent shuts down. AppServer agents shut down when you use the trim feature and also when you shut down the AppServer. The <em>Shutdown procedure</em> property is not available for an AppServer running in state-reset operating mode. You can update this property dynamically. Any changes will affect all current and new agents.</td>
</tr>
<tr>
<td>(dynamic property)</td>
<td></td>
</tr>
<tr>
<td><strong>Startup procedure parameters</strong></td>
<td>The parameters to pass the startup procedure when it executes. The <em>Startup procedure parameters</em> property takes a character string as an input parameter. For example: <strong>DEFINE INPUT PARAMETER startup-data AS CHARACTER NO-UNDO.</strong> You can set this parameter to any arbitrary value. If you do not specify a parameter in this field, the parameter is set to the unknown value (?) when the AppServer agent executes the startup procedure. You can update this property dynamically. Any changes will affect only new agents; existing (running) agents are not updated.</td>
</tr>
<tr>
<td>(dynamic property)</td>
<td></td>
</tr>
<tr>
<td><strong>Execution Time Limit</strong></td>
<td>Sets the maximum time in seconds that a remote procedure can execute on a given AppServer. The time limit applies to all remote procedures that execute on the AppServer. The default value is zero, which indicates that no time limit exists.</td>
</tr>
</tbody>
</table>

*OpenEdge*® Management and OpenEdge Explorer: Configuration 157
Viewing or modifying AppServer SSL properties

You can view or modify the following SSL properties for an AppServer Broker: general properties and advanced features.

To view or modify AppServer SSL properties:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the AppServer broker instance whose SSL properties you want to view or modify.

3. Click the Edit icon. The AppServer details page appears.

4. In the Command and control section of the page, click Configuration. The AppServer Configuration page appears with the Broker properties in focus.

5. Click the SSL tab.

From this page, you can:

- View a read-only display of the AppServer SSL properties as described in Table 42 and Table 43. These tables follow this procedure.

- Click Edit to modify the SSL properties. For details, see the “Editing an AppServer configuration” section on page 163.

Table 42: AppServer SSL General properties (1 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable SSL client connections</td>
<td>If the check box is selected, specifies that all connections to this AppServer must use SSL tunneling.</td>
</tr>
<tr>
<td>Private key/Digital certificate alias name</td>
<td>The alias name within the OpenEdge keystore of the private key and digital certificate entry to use to authenticate all connections to this AppServer. If not otherwise selected, the AppServer uses the default_server server certificate alias.</td>
</tr>
</tbody>
</table>
### Table 42: AppServer SSL General properties (2 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Password to access the key/certificate** | The password to use for accessing the private key and digital certificate.  
*Note:* OpenEdge Explorer encrypts your password in the `ubroker.properties` file. If you set the password in `ubroker.properties` manually, you must specify the password as an encrypted value. You can obtain this value using the `genpassword` utility located in the bin directory of your OpenEdge installation.  
If you use the default_server server certificate, it also has a default password that you do not need to specify. |
| **Password to access the key/certificate Confirm** | Confirm the password. This field appears only if you are editing the properties. |

### Table 43: AppServer SSL Advanced Features properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Disable SSL session caching</strong></td>
<td>If the check box is selected, caching for the SSL client session is disabled.</td>
</tr>
</tbody>
</table>
| **SSL session cache timeout** | The length of time, in seconds, that an SSL client session is held in the session cache, during which an SSL client can resume its session.  
The default is 180. |
Viewing or modifying AppServer Messaging properties

You can view or modify several AppServer Messaging properties.

To view or modify AppServer Messaging properties:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the AppServer broker instance whose messaging properties you want to view or modify.

3. Click the Edit icon. The **AppServer** details page appears.

4. In the **Command and control** section of the page, click **Configuration**. The **AppServer Configuration** page appears with the **Broker** properties in focus.

5. Click the **Messaging** tab.

   From this page, you can:

   - View a read-only display of the AppServer Messaging properties as described in **Table 44**.
   - Click **Edit** to modify the AppServer Messaging properties. For details, see the “Editing an AppServer configuration” section on page 163.
Table 44 lists the AppServer Messaging properties.

Table 44: AppServer Messaging properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SonicMQ ServerConnect enabled</td>
<td>Allows the asbroker or wsbroker agent to start a personal SonicMQ Adapter instead of going to a separate one running elsewhere.</td>
</tr>
<tr>
<td>SonicMQ ServerConnect broker log filename</td>
<td>A valid path and file name.</td>
</tr>
<tr>
<td>Broker logging level (dynamic property)</td>
<td>A value that specifies the amount of information to be written to the SonicMQ ServerConnect broker log. You can select from the following values in the drop-down list:</td>
</tr>
<tr>
<td></td>
<td>• None — Log no entries. This is equivalent to turning logging off.</td>
</tr>
<tr>
<td></td>
<td>• Error Only — Log error messages. This includes all error messages and is unrelated to the entry types specified. Errors continue to be logged at all higher levels.</td>
</tr>
<tr>
<td></td>
<td>• Basic — Broker logging entry types determine the logged information. Each entry type generates at least some output. This is the default.</td>
</tr>
<tr>
<td></td>
<td>• Verbose — Logging entry types determine the logged information, but it is typically more information than Basic.</td>
</tr>
<tr>
<td></td>
<td>• Extended — Logging entry types determine the logged information, but it is typically more information than Verbose.</td>
</tr>
<tr>
<td></td>
<td>You can update this property dynamically. Any changes will affect all current and new brokers and/or agents.</td>
</tr>
<tr>
<td>Append to broker log file</td>
<td>If the check box is selected, Indicates that a new broker log file should be created, even if the broker log file already exists, when the SonicMQ ServerConnect broker is started.</td>
</tr>
<tr>
<td>SonicMQ ServerConnect server log filename</td>
<td>A valid path and file name.</td>
</tr>
</tbody>
</table>
Setting or deleting AppServer broker environment variables

You can set or delete environment variables. For the AppServer, environment variables that appear in the list are scoped to:

- The process in which the AppServer broker executes
- Every AppServer agent that the AppServer broker starts

To set or delete AppServer broker environment variables:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the AppServer broker instance whose environment variables you want to set.

3. Click the Edit icon. The AppServer details page appears.
4. In the Command and control section of the page, click Configuration. The AppServer Configuration page appears.

5. Click Edit.

6. Click Environment Variables.

7. Enter each variable name and then provide its value in the format name=value. Alternatively, you can remove an existing variable by deleting the name and value pair.

8. Click Save when you finish setting (or deleting) the variables and their values.

Editing an AppServer configuration

You can edit the configuration properties for an AppServer resource from the management console.

To edit AppServer configuration properties:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the AppServer broker instance whose configuration properties you want to edit.

3. Click the Edit icon. The AppServer details page appears.

4. In the Command and control section of the page, click Configuration. The AppServer Configuration page appears.

5. Click Edit, and make the changes you want to the properties. (For details about the properties, see Table 32 through Table 44.)

6. Click Save. Any changes you make are also reflected in the ubroker.properties file.
Creating an AppServer

OpenEdge Management and OpenEdge Explorer provide a sample AppServer instance (asbroker1) with a default configuration. Each new instance you create uses the same default configuration, which you can edit.

To create an AppServer:

1. From the drop-down for Resources on the management console menu, click **New OpenEdge Resource** → **AppServer**.

   The **AppServer Configuration** page appears.

2. Type the name of the new AppServer in the field provided.

   **Note:** The AppServer name is case sensitive and can include any character except a period (.) or square brackets ([ ]). The name must be unique among all configured AppServer names.

3. Click **Save**. The **AppServer Configuration** page appears, allowing you to configure the AppServer’s properties.

   Each new AppServer instance you create uses the default configuration. However, the port number must be unique for each AppServer instance in order for the instance to operate properly. Once you create a new instance, you are prompted to change the port number.

Starting and stopping an AppServer

From the management console, you can start or stop either a local or a remote AppServer. You must first connect to the AdminServer that manages the AppServer you want to start.

You can also choose to start an AppServer automatically when the AdminServer starts.

**Note:** Prior to starting an AppServer using SonicMQ ServerConnect, you must first configure the AppServer as SonicMQ ServerConnect enabled. To stop a remote AppServer, access the corresponding local AppServer instance selected in the container of the host where it runs.

To start or stop an AppServer, refer to the **Starting or Stopping OpenEdge resources** section in *OpenEdge Management: Servers, DataServers, Messengers, and Adapters*.
Starting an AppServer automatically

You can set an AppServer to start automatically when the AdminServer starts.

To start an AppServer automatically:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the AppServer broker instance you want to start automatically.

3. Click the Edit icon. The AppServer details page appears.

4. In the Command and control section of the page, click Configuration. The AppServer Configuration page appears.

5. Click Edit.

6. Select the Auto start option by clicking in the check box.

7. Click Save.
Chapter 5: Configuring AppServers and AppServer Internet Adapters

Adding or trimming AppServer agents

At run time, you can increase the number of AppServer agents running and available to service AppServer client requests by having the AppServer add a specified number of AppServer agents to its process pool.

You can also trim running AppServer agents, down to the Minimum Servers property setting, from the management console.

More information about trimming AppServer agents

To better understand the use of the Agent Auto-trim feature, consider a scenario involving a broker serving a client/server application running in stateless mode. Assume that the following property values are set:

- Minimum Agent/Server Instances: 10
- Initial Agent/Server Instances to Start: 10
- Maximum Agent/Server Instances: 50
- Agent/Server Auto-trim Interval: 1800

When the broker starts, it starts 10 agents. This also marks the start of the first Agent/Server Auto-trim Interval. Assume that the broker is idle for 30 minutes (it is just before people arrive for work, for instance). At the end of the 30-minute interval, the highest number of agents that were busy at the same time would be zero. (Since there were no requests made of any of the agents, none of the agents was busy.) However, since zero is less than the Minimum Agent/Servers Instances value of 10, no automatic trimming takes place. This marks the start of the second interval.

As the second interval starts, employees begin their workday. There is a sudden demand for agents; in fact, the broker receives so many requests that it must start 20 additional agents. Assume that the high volume of workload, and demand on the agents, continues through the second interval. At the interval's end, the maximum number of busy agents is 30, which is the number the broker scaled up to in response to high demand. Thirty agents are running, and the maximum busy level is 30. No trimming occurs.

During the next interval, the request load diminishes somewhat. At the end of the interval, only 20 agents are concurrently busy. The maximum number of busy agents for the interval remains at 30, since 30 were busy at the start of the interval. Thirty agents are running, and the maximum busy level is 30. No trimming occurs.

Assume that all employees have to attend a company-wide meeting that begins midway through the interval. The highest number of agents busy at that same time was 20. At the end of the interval, there are 30 agents running, with zero agents currently busy. Since the maximum busy level is 20 (at the start of the interval), and the number of running agents is 30 (all are available since everyone is at the company meeting), the broker attempts to trim back the number of agents to 20.

Assume that the company meeting ran for another half-hour, and the entire next interval elapsed without any requests. The maximum busy agents would be zero, with 20 currently running. Since the Minimum Agent/Server Instances property value is set to 10, the broker attempts to trim back to 10 agents.
Keep in mind that the lower you set the Agent/Server Auto-trim Interval value, the more quickly the broker reacts to a drop in the workload. Under a reasonably steady work load, the broker should trim the number of running agents to the number needed. However, if the workload fluctuates a great deal and the Agent/Server Auto-trim Interval is set too low, the broker may trim agents that it must then restart. As stated earlier, repetitive trimming and restarting causes the system to waste resources on unnecessary process management.

To add or trim AppServer agents:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the AppServer broker instance to which you want to add or trim agents.

3. Click the Edit icon. The AppServer details page appears.

4. If the AppServer is not started, click Broker Control. Then click Start AppServer.

5. When the AppServer is running, click the backlink for the AppServer broker instance on the breadcrumb trail to return to the details page.

6. In the Command and control section of the page, click Server Pool Control.

7. Review the Server pool initial configuration, the Servers state, and the Server pool summary information. (You can view detailed information about an agent by clicking on its PID.)

8. To add or trim servers, choose the corresponding option in the Add/Trim dropdown menu, and type the number of servers in the box provided.

9. Click Submit. A confirmation message appears.

Viewing the status of an AppServer

You can view the status of any running AppServer instance through the management console. Status and/or statistical details appear in the following different views:

- The AppServer instance’s Details page
- The Broker Control link on the AppServer instance’s Details page
- The Broker PID link within the instance’s AppServer Control page
- The Server Pool Control link on the AppServer instance’s Details page

You can also add and trim AppServer agents from the Server Pool Control page.
To view the status of an AppServer instance from the Details page:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the AppServer broker instance whose status you want to view.

3. Click the Edit icon. The **AppServer** details page appears and displays the following information:
   - **Host** — The name of the host machine
   - **Broker** — Whether the broker is active or not running
   - **Operating mode** — One of four possible modes can be reported: **Stateless**, **State-reset**, **State-aware**, and **State-free**
   - **Broker statistics available** — Whether broker statistics are being collected
   - **Servers available** — The number of AppServers running and available to fulfill a connection request from a client to an AppServer through this broker when the broker's status is **ACTIVE**
   - **Should register with NameServer** — The status of **True** or **False** to indicate whether or not the broker resource is registered with a NameServer

To view the status of an AppServer instance from the Broker Control link:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the AppServer broker instance whose status you want to view.

3. Click the Edit icon. The **AppServer** details page appears.

4. In the **Command and control** section of the page, click **Broker Control**. The **AppServer Control** page for that instance appears and displays the following information:
   - **Broker name** — The name of the selected AppServer broker.
   - **Host** — The name of the host machine.
   - **Port** — The port number.
   - **Broker PID** — The broker's process identification number (PID).
   - **Status** — The running status of the broker. Possible values are:
     - **ACTIVE** — The broker is currently running.
     - **Not Running** — The broker is not currently running.

The broker can also report **Starting** and **Shutting Down** values; however, depending on the speed of the machine on which your management console is running, you may not see these intermediary states.
• **Operating mode** — One of four possible modes can be reported: **Stateless**, **State-free**, **State-aware**, or **State-reset**.

• **Broker statistics available** — Whether collected broker statistics are available.

• **Enabled** — Whether the broker is enabled.

• **Collect Statistics** — Whether statistics are being collected.

To view the status of an AppServer instance from the Server Pool Control link:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the AppServer broker instance whose status you want to view.

3. Click the Edit icon. The **AppServer** details page appears.

4. In the **Command and control** section of the page, click **Server Pool Control**. The **Server Pool Control** page for that instance appears and displays the following information:

   - **Initial number of servers to start** — The value referenced when the AppServer broker starts AppServers.

   - **Minimum servers** — The minimum number of AppServers that must be simultaneously running before the AppServer broker starts additional servers. The broker strives to maintain this specified minimum. If at any time the number of servers falls below the specified minimum, the broker automatically starts the additional servers needed to maintain the minimum.

     If you set a trim value that requires trimming the number of servers to below the number specified for this field, a message appears.

   - **Maximum servers** — The maximum number of AppServer processes that can be running simultaneously. Add requests you initiate that will exceed the specified maximum are not fulfilled.

   - **Active servers** — The number of AppServers currently running.

   - **Busy servers** — The number of AppServers handling ABL client requests.

   - **Locked servers** — The number of AppServers servicing a bound connection. (This state applies to a stateless AppServer.)

   - **Available servers** — The number of AppServers available to handle broker requests.

A server pool summary provides details about the following:

- **PID** — The process identifier for this AppServer. Click the PID number to display a detail page that provides specific information about this server process and, as necessary, to kill the process. See the section about killing an AppServer process in OpenEdge Management: Servers, DataServers, Messengers, and Adapters for more information.
- **State** — The current execution state of the agent.
- **Port** — The TCP/IP port number used by the agent.
- **nRq** — The number of messages sent to the agent.
- **nRcvd** — The number of messages received by the agent.
- **nSent** — The number of requests sent by the agent.
- **CPU Use** — The percentage of CPU user and system time consumed by a process.
- **Memory Use** — The amount of virtual memory (in Kbytes) consumed by a process.
- **Started** — The time stamp indicating when the agent started.
- **Last Change** — The time stamp indicating when the agent last changed execution state.
Viewing AppServer process details

You can access real-time details and statistics that provide you with snapshot information about a running AppServer process at the point you access this information from the Control page.

To view process details:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the AppServer broker instance to which you want to add or trim agents.

3. Click the Edit icon. The AppServer details page appears.

4. In the Command and control section of the page, click Server Pool Control.

5. Click a PID link. The following sections appear:

   • The Process summary section identifies the Process name and Process start time. User id and Group id values appear when UNIX-based data is shown. The Parent pid provides the identifier number associated with the process that spawned this current process.

   • The Process statistics section presents details about the database’s real-time operational status. Values presented without parentheses identify that the processing time determined since the last scheduled polling interval, as noted, has occurred. Values presented within parentheses have been calculated based on information obtained since the start of the process.
Table 45 identifies and describes the fields of information presented in the Process statistics section.

**Table 45: Process statistics data**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resident size</td>
<td>The physical size of the process as defined by the host system.</td>
</tr>
<tr>
<td>Virtual size</td>
<td>The virtual size of the process as defined by the host system.</td>
</tr>
<tr>
<td>CPU</td>
<td>The percentage of time spent using the CPU in either the user or kernel mode since the last scheduled poll.</td>
</tr>
<tr>
<td>Weighted CPU</td>
<td>The percentage of time spent using the CPU in either the user or kernel mode since the last scheduled poll divided by the number of CPU processors on the system. This value appears only when there is more than one CPU process on the system where the process is running.</td>
</tr>
<tr>
<td>User time</td>
<td>The amount of CPU time spent in the user mode since the last scheduled poll.</td>
</tr>
<tr>
<td>Kernel time</td>
<td>The amount of CPU time spent in the kernel mode since the last scheduled poll.</td>
</tr>
<tr>
<td>Process time</td>
<td>The sum of the values that appear in the User time and Kernel time fields.</td>
</tr>
</tbody>
</table>

**Stopping an AppServer agent**

When you stop an AppServer agent, the related process is completed and then stopped.

To stop an AppServer agent:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the AppServer broker instance which you want to stop.
3. Click the required instance icon. The AppServer details page appears.
4. In the Command and control section of the page, click Server Pool Control.
5. Click a PID link.
6. Click Stop. Click OK when you get a confirmation message.
Killing an AppServer agent

When you kill an AppServer agent, the agent is stopped immediately without waiting for any related processes to complete.

To kill an AppServer agent:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the AppServer broker instance which you want to kill.
3. Click the required instance icon. The AppServer details page appears.
4. In the Command and control section of the page, click Server Pool Control.
5. Click a PID link.
6. Click Kill. Click OK when you get a confirmation message.

Deleting an AppServer

You can delete any AppServer instance listed in the management console as long as the instance is not running. (The Delete button is not available if the instance is running.) If the instance is running, first stop it and then delete it.

To delete an AppServer instance:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the AppServer broker instance you want to delete.
3. Click the Edit icon. The AppServer details page appears.
4. Click Delete, and then click OK to confirm the deletion.

Viewing the AppServer log files

The AppServer log file viewer allows you to examine the contents of a log file through an HTML interface.

For details, see the section about using the AppServer log file viewer in OpenEdge Management: Servers, DataServers, Messengers, and Adapters.
For additional AppServer information

You can find additional information about using OpenEdge Management with the AppServer in *OpenEdge Management: Servers, DataServers, Messengers, and Adapters*. See that document for details, including specifics about the following:

- Working with the AppServer log file monitor and viewer
- Killing an AppServer process
- Listing AppServer Client connections
AppServer Internet Adapter configuration and administration

With the AppServer Internet Adapter (AIA) you can make AppServer or SonicMQ Adapter application services available over the Web to ABL applications, and you can make AppServer application services available over the Web to .NET and Java Open Clients. The AIA is a Java Servlet that is invoked by a Java Servlet Engine (JSE).

When you use the URL syntax for one of the following connection protocols to connect standard ABL clients (including AppServer and WebSpeed agents, and WebClients) or .NET and Java Open Clients to an AppServer, or to connect standard ABL clients to a SonicMQ Adapter:

- **HTTP** — Directs the client to connect through an AIA instance using an unsecured Internet connection to the Web server that hosts the AIA
- **HTTPS** — Directs the client to connect through an AIA instance using a Secure Sockets Layer (SSL) Internet connection to the Web server that hosts the AIA

**Notes:** You can also use the AppServer protocol to allow a client to connect directly to an AppServer or a SonicMQ Adapter using the TCP/IP protocol with or without SSL tunneling. This is the preferred option for client access over an intranet.

However, if you want to use a secure Internet connection to the Web server and also secure the connection between the AIA and the AppServer (or SonicMQ Adapter), you can configure the AIA to use SSL for its own connection to the AppServer (or SonicMQ Adapter).

Working with the AppServer Internet Adapter

To work with an AIA instance, double-click the **AppServer Internet Adapter** folder from the management console's list pane. There is one pre-defined default AppServer Internet Adapter (Aia1).

The following sections describe these AppServer Internet Adapter actions:

- Configuring an AppServer Internet Adapter
- Editing an AppServer Internet Adapter configuration
- Creating an AppServer Internet Adapter
- Deleting an AppServer Internet Adapter broker
- Viewing the AppServer Internet Adapter log file
Configuring an AppServer Internet Adapter

You can view or modify the following configuration properties of an AppServer Internet Adapter broker from the management console:

- General
- Controlling NameServer
- Logging Setting
- SSL
- Advanced Features

You can also create a new AppServer Internet Adapter instance, delete an instance you no longer need, or view its log file.

To view or modify AppServer Internet Adapter configuration properties:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the AppServer Internet Adapter broker instance whose configuration properties you want to view or modify.

3. Click the Edit icon. The **AppServer Internet Adapter** details page appears.

4. In the **Command and control** section of the page, click **Configuration**. The **AppServer Internet Adapter Configuration** page appears with the **General** properties in focus.

From this page, you can:

- View a read-only display of the AppServer Internet Adapter instance’s properties, as described in **Table 46** through **Table 50**. All of these tables follow this procedure.

- Click **Edit** to modify the properties. For details, see the “Editing an AppServer Internet Adapter configuration” section on page 182.
### Table 46: AIA General properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Idle connection timeout</td>
<td>Timeout value, in seconds, between an AppServer client and an AIA instance. If a connection is idle for more than the specified timeout value, then the AIA instance automatically disconnects from the AppServer. The default is 3600.</td>
</tr>
<tr>
<td>Secure port</td>
<td>Identifies the port on which the Web server listens for requests that specify the HTTPS protocol. This value is used only when the HTTPS enabled property check box is selected. If you do not specify a value for this property, and the HTTPS enabled property check box is selected, the default value for Secure port is 443.</td>
</tr>
<tr>
<td>HTTPS enabled</td>
<td>Determines how an AppServer connection using the AIA connects to the AIA's Web server host. If you select this option, clients must use HTTPS to connect and send encrypted data across the Internet to the Web server. If you do not select this option, clients can use HTTP to connect and send data across the Internet without encryption. When the HTTPS enabled check box is selected, if an AppServer client uses HTTP to connect to an AppServer, the AIA returns an HTTPS redirection URL to the client. The client automatically retries the connection using the HTTPS URL. The default setting for this check box is enabled.</td>
</tr>
</tbody>
</table>

### Table 47: AIA Controlling NameServer properties (1 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connect using NameServer</td>
<td>A check box that indicates whether to connect to an AppServer using a Controlling NameServer.</td>
</tr>
<tr>
<td>Controlling NameServer</td>
<td>Identifies the NameServer that the AIA uses to find a broker that supports the correct application services.</td>
</tr>
</tbody>
</table>
Chapter 5: Configuring AppServers and AppServer Internet Adapters

Table 47: AIA Controlling NameServer properties (2 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum NameServer client port</td>
<td>Properties used to limit the range of ports of an AIA communicating with the NameServer via UPD. This is important when the AIA and NameServer are separated by a firewall. The value for these two properties must be a number between 1024 and 65535 inclusive (or 0), and Minimum NameServer client port must be less than or equal to Maximum NameServer client port. If the minimum and maximum values are both set to zero, then a random port number within the implicit fixed range will be dynamically assigned. If both minimum and maximum values are set to the same port number, then this UDP port number will be used when communicating with the NameServer. The default value for both minimum and maximum is 0.</td>
</tr>
<tr>
<td>Maximum NameServer client port</td>
<td></td>
</tr>
<tr>
<td>AppServer host name</td>
<td>If you did not select the Connect using NameServer check box and the AIA will be directly connected to an AppServer or SonicMQ Adapter, you must specify the AppServer's or SonicMQ Adapter's host location.</td>
</tr>
<tr>
<td>AppServer port</td>
<td>If you did not select the Connect using NameServer check box and the AIA will be directly connected to an AppServer or SonicMQ Adapter, you must specify the AppServer's or SonicMQ Adapter's port.</td>
</tr>
</tbody>
</table>
Table 48: AIA Logging Setting properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log filename</td>
<td>The AIA log filename. Enter a valid path name.</td>
</tr>
<tr>
<td></td>
<td>The installed default is @WorkPath\Aia1\aia.log</td>
</tr>
<tr>
<td>Logging level</td>
<td>A value that specifies the amount of information to be written to the AIA log. Select from the following values in the drop-down list:</td>
</tr>
<tr>
<td></td>
<td>• None — Log no entries. This is equivalent to turning logging off.</td>
</tr>
<tr>
<td></td>
<td>• Error Only — Log OpenEdge error messages. This includes all error messages and is unrelated to the entry types specified. Errors continue to be logged at all higher levels.</td>
</tr>
<tr>
<td></td>
<td>• Basic — Logging entry types determine the logged information. Each entry type generates at least some output. This is the default.</td>
</tr>
<tr>
<td></td>
<td>• Verbose — Logging entry types determine the logged information, but it is typically more information than Basic.</td>
</tr>
<tr>
<td></td>
<td>• Extended — Logging entry types determine the logged information, but it is typically more information than Verbose.</td>
</tr>
<tr>
<td></td>
<td>The default is Basic.</td>
</tr>
<tr>
<td>Append to log file</td>
<td>A check box that indicates whether or not a new server log file should be created when the AIA is started.</td>
</tr>
<tr>
<td></td>
<td>To create a new AIA log file each time the AIA is started, even if the AIA log file already exists, clear the Append to log file check box.</td>
</tr>
<tr>
<td></td>
<td>To append log entries to the existing AIA log file, select the Append to log file check box.</td>
</tr>
<tr>
<td></td>
<td>The Append to log file check box is selected by default.</td>
</tr>
</tbody>
</table>
The AIA log entry type turns on logging for different AIA actions based on the logging level. The AIA log entry types provide the following information:

- **AiaDefault** — Combines AiaMgmt and AiaProp. This is the default entry type for the AIA component.
- **AiaMgmt** — Logs messages about regular AIA operations, such as AIA version information, instantiation, informational details, and information on the AIA connection pool according to the following logging level settings:
  - **Basic** — Logs instantiation and informational messages
  - **Verbose** — Logs the connection pool information
  - **Extended** — Logs the same information as **Verbose**
- **AiaProp** — Logs the properties set for the specific AIA instance. These properties are set in the ubroker.properties file. Information is logged at **Basic** and higher.
- **AiaRqst** — Logs information on the GET and POST commands sent by the client and the response back to the client. Information is logged at **Basic** and higher.
- **AiaUBroker** — Logs information on the communication between the AIA and the Unified Broker (AppServer or SonicMQ Adapter) according to the following Logging level settings:
  - **Basic** — Provides most of the information
  - **Verbose** — Logs a binary dump of messages exchanged between the client and the broker/server process

### Table 48: AIA Logging Setting properties (2 of 3)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logging entry types</td>
<td>The AIA log entry type turns on logging for different AIA actions based on</td>
</tr>
<tr>
<td></td>
<td>the logging level. The AIA log entry types provide the following information:</td>
</tr>
<tr>
<td></td>
<td>- <strong>AiaDefault</strong> — Combines AiaMgmt and AiaProp. This is the default entry</td>
</tr>
<tr>
<td></td>
<td>type for the AIA component.</td>
</tr>
<tr>
<td></td>
<td>- <strong>AiaMgmt</strong> — Logs messages about regular AIA operations, such as AIA</td>
</tr>
<tr>
<td></td>
<td>version information, instantiation, informational details, and information</td>
</tr>
<tr>
<td></td>
<td>on the AIA connection pool according to the following logging level settings:</td>
</tr>
<tr>
<td></td>
<td>- <strong>Basic</strong> — Logs instantiation and informational messages</td>
</tr>
<tr>
<td></td>
<td>- <strong>Verbose</strong> — Logs the connection pool information</td>
</tr>
<tr>
<td></td>
<td>- <strong>Extended</strong> — Logs the same information as <strong>Verbose</strong></td>
</tr>
<tr>
<td></td>
<td>- <strong>AiaProp</strong> — Logs the properties set for the specific AIA instance. These</td>
</tr>
<tr>
<td></td>
<td>properties are set in the ubroker.properties file. Information is logged at</td>
</tr>
<tr>
<td></td>
<td><strong>Basic</strong> and higher.</td>
</tr>
<tr>
<td></td>
<td>- <strong>AiaRqst</strong> — Logs information on the GET and POST commands sent by the</td>
</tr>
<tr>
<td></td>
<td>client and the response back to the client. Information is logged at **Basic</td>
</tr>
<tr>
<td></td>
<td>and higher.</td>
</tr>
<tr>
<td></td>
<td>- <strong>AiaUBroker</strong> — Logs information on the communication between the AIA and</td>
</tr>
<tr>
<td></td>
<td>the Unified Broker (AppServer or SonicMQ Adapter) according to the following</td>
</tr>
<tr>
<td></td>
<td>Logging level settings:</td>
</tr>
<tr>
<td></td>
<td>- <strong>Basic</strong> — Provides most of the information</td>
</tr>
<tr>
<td></td>
<td>- <strong>Verbose</strong> — Logs a binary dump of messages exchanged between the client</td>
</tr>
<tr>
<td></td>
<td>and the broker/server process</td>
</tr>
</tbody>
</table>
### Table 48: AIA Logging Setting properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Log file threshold size</strong></td>
<td>A value of 0 or a value between 500,000 and 2,147,483,647, where 0 means there is no limit on the log file size other than what the operating system imposes.</td>
</tr>
<tr>
<td><strong>Maximum number of log files</strong></td>
<td>The number of rolled-over log files to keep. The value can be 0 or a value between 2 and 999999, where 0 means there is no limit on the number of AIA log files to keep. The specified number represents the maximum total number of log files to keep on disk at any time, including the current log file being written to. When the file becomes equal to or greater than the <strong>Log file threshold size</strong>, the client process renames it and creates a new log file. The file is renamed as follows, where ###### is a number starting at 000001 and increasing to 999999, after which it rolls back over to 000001: filename.######.extension</td>
</tr>
</tbody>
</table>

### Table 49: AIA SSL properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Enable SSL client connections</strong></td>
<td>If the check box is selected, this property directs that the connection to the AppServer or SonicMQ Adapter use SSL tunneling.</td>
</tr>
<tr>
<td><strong>Disable client verification of SSL host name</strong></td>
<td>If the check box is selected, this property turns off host verification for an SSL connection. If the check box is cleared, the client compares the host name for the AppServer or SonicMQ Adapter specified by AppServer Location Host Name (or as returned by any Controlling NameServer) with the Common Name specified in the server digital certificate, and raises an error if they do not match. With this parameter specified, the client never raises the error.</td>
</tr>
<tr>
<td><strong>Disable SSL session reuse</strong></td>
<td>If the check box is selected, the connection does not reuse the SSL session ID when reconnecting to the same AppServer or SonicMQ Adapter.</td>
</tr>
</tbody>
</table>
To edit an AppServer Internet Adapter configuration:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the AppServer Internet Adapter broker instance whose configuration you want to edit.
3. Click the AppServer Internet Adapter broker instance. The AppServer Internet Adapter details page appears.
4. In the Command and control section of the page, click Configuration. The AppServer Internet Adapter Configuration page appears.
5. Click Edit. (For details about the properties, see Table 46 through Table 50.)
6. Make the configuration changes, and then click Save.

### Table 50: AIA Advanced Features properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal administrative command</td>
<td>Provides permission to see whether or not the AIA is running successfully.</td>
</tr>
<tr>
<td></td>
<td>The check box is cleared by default.</td>
</tr>
<tr>
<td>Authorized IP list</td>
<td>If the Internal administrative command check box is selected, this box shows</td>
</tr>
<tr>
<td></td>
<td>a comma-separated list of IP addresses that are allowed to access the AIA</td>
</tr>
<tr>
<td></td>
<td>Adapter's internal administrative commands.</td>
</tr>
<tr>
<td></td>
<td>To add or delete an IP address, click Edit and type or remove an IP address</td>
</tr>
<tr>
<td></td>
<td>in the box.</td>
</tr>
<tr>
<td></td>
<td>To allow any IP address to access the AIA Adapter's internal commands,</td>
</tr>
<tr>
<td></td>
<td>leave the box blank. This is the default.</td>
</tr>
<tr>
<td>AppServer Keepalive</td>
<td>Allows the AppServer to recognize that a client bound to it is no longer</td>
</tr>
<tr>
<td></td>
<td>connected.</td>
</tr>
<tr>
<td>Enable dynamic property updates</td>
<td>If the check box is selected, dynamic property updates are allowed.</td>
</tr>
</tbody>
</table>
Creating an AppServer Internet Adapter

You can create an AppServer Internet Adapter instance.

To create an AppServer Internet Adapter instance:

1. From the drop-down for Resources on the management console menu, click New OpenEdge Resource → AppServer Internet Adapter.

   The AppServer Internet Adapter Configuration page appears.

2. Type the name of the new instance in the field provided.

   Note: The AppServer Internet Adapter name is case sensitive and can include any character except a period (.) or square brackets ([ ]). The name must be unique among all configured AppServer Internet Adapter names.

3. Click Save. The AppServer Internet Adapter Configuration page appears, showing the configuration details and allowing you to configure the instance’s properties.

Viewing the AppServer Internet Adapter log file

You can view the log file for an AppServer Internet Adapter log file by using the log file viewer. The log file viewer allows you to examine the log file through an HTML interface.

To access the log file viewer:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the AppServer Internet Adapter broker instance whose log file viewer you want to access.

3. Click the Edit icon. The AppServer Internet Adapter details page appears.

4. In the Command and control section of the page, click Log File Viewer.

   You can work with the Log File Viewer in the following ways:

   - Use the Show field to control how many AIA log file entries display at one time. The number entered into the Show field cannot 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 cannot be more 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.
Chapter 5: Configuring AppServers and AppServer Internet Adapters

- Click **Reload** after changing the values in either the **Show** field or the **Overlap** field. If you do not reload, the viewer continues to display the previous values.

- Click **Go To** to control which numbered entry in the log file the viewer begins its display with. For example, a value of 10 entered into the **Go To** field will begin the display at the tenth log file entry.

| Note: You must click **Go To** after entering a value in the **Go To** field or the viewer will not update its display. |

The default display of entries is in ascending order; choose **Descending** to change the display. Note that the **Show** field dictates the number of entries shown, regardless of whether they display in ascending or descending order.

- Click **First** to display the first $x$ entries, where $x$ is the value in the **Show** field.

- Click **Previous** 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 last $x$ entries, where $x$ is the value in the **Show** field.

- To view additional log file entries without changing your current starting log file entry, leave the **Go To** field blank, change the value in the **Show** field, and click **Reload**.

- If the contents of the log file have changed since you opened the viewer, the log file viewer indicates this in the **Log file status** field.

- OpenEdge Management and OpenEdge Explorer consider a viewer that has been inactive for more than four hours stale and at that point release ninety-five percent of any memory being held. If you try to use a stale viewer, OpenEdge Management and OpenEdge Explorer automatically reload the file. Because additional resource activity might have occurred during the viewer's inactivity, the reloaded log file view might not match the previous log file view of that resource.

- OpenEdge Management and OpenEdge Explorer consider a viewer that has been inactive for forty-eight hours dead. Once a viewer dies, OpenEdge Management and OpenEdge Explorer release all of its memory. To return to the log file displayed in a dead view, you must renavigate to it, even if you pinned up the view or saved a link to it before the viewer died.
Enabling or disabling an AppServer Internet Adapter broker

You can enable or disable an AppServer Internet Adapter broker.

To enable or disable an AppServer Internet Adapter broker:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the AppServer Internet Adapter broker instance you want to enable or disable.

3. Click the Edit icon. The AppServer Internet Adapter details page appears.

4. In the Command and control section of the page, click Control. The broker’s Control page appears.

5. Click Edit, and either select the Enabled check box to enable the broker or clear the check box to disable the broker.

6. Click Save.

Deleting an AppServer Internet Adapter broker

You can delete an AppServer Internet Adapter (AIA) instance using the management console. The instance is also deleted from the ubroker.properties file.

If you delete an AIA instance that is running, the instance and all associated properties are deleted from the ubroker.properties file. However, the AIA instance continues to run until you use the Java servlet engine to shut down the AIA.

To delete an AppServer Internet Adapter broker:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the AppServer Internet Adapter broker instance you want to delete.

3. Click the Edit icon. The AppServer Internet Adapter details page appears.

4. Click Delete, and then click OK to confirm the deletion.
Configuring OpenEdge SOAP Web Services

You can work with Web Services Adapter instances (for OpenEdge SOAP Web Services) in OpenEdge Management and OpenEdge Explorer, as described in the following sections:

- Configuring and managing SOAP Web services
- Managing Web Services Adapter (for OpenEdge SOAP Web Services) instances
- Working with SOAP Web services
Configuring and managing SOAP Web services

The Web Services Adapter (WSA) (for OpenEdge SOAP Web Services) is a Java servlet that exposes OpenEdge AppServer applications as SOAP Web services. The WSA is installed and runs in the context of a Java servlet engine (JSE) that runs in the context of a Web server.

To expose AppServer applications as SOAP Web services, the WSA serves a dual role:

- As a gateway between the SOAP request messages, which SOAP Web services and SOAP Web service clients exchange, and ABL applications on the AppServer, which execute SOAP Web service requests
- As an application server that hosts, manages, and provides communications and run-time support for multiple deployed SOAP Web service applications

SOA-AM gateway

For any given SOAP Web service, the WSA is a gateway between SOAP and the ABL Virtual Machine (AVM) running on the AppServer. As such, the WSA receives SOAP Web service method calls and parameters sent as SOAP messages from SOAP Web service clients and converts them into ABL function and procedure calls that the WSA invokes on a specified AppServer. In response, the WSA also receives the return and output parameter values sent from the called ABL functions and procedures on the AppServer and converts them into SOAP messages that it sends back to the calling SOAP Web service clients.

Web Service Application Server

The WSA is where you deploy a SOAP Web service that you want to make available to clients. Once deployed, it also allows you to manage the SOAP Web service individually and as part of a group of SOAP Web services. This includes such options as making SOAP Web service definitions available to potential clients as downloadable Web Services Description Language (WSDL) files and making one or more SOAP Web services available as applications for access by clients.

To initially create SOAP Web service definitions, you use ProxyGen to define an existing set of AppServer procedures and functions in terms of SOAP Web service objects. ProxyGen generates a Web Service Mapping (WSM) file. This WSM file stores the SOAP Web service object definitions in a form similar to a WSDL file, but without the final deployment information targeted to a specific WSA installation.

Thus, when you deploy a SOAP Web service to a WSA, you specify the information including the WSM file into a WSDL file that a specific WSA can use to provide the SOAP Web service definition to Web service clients. This same process also creates a Web Service Application Descriptor (WSAD) file that provides the SOAP Web service definition to a specific WSA in a form that allows it to host the deployed SOAP Web service and manage its run-time behavior.
Managing SOAP Web services with the WSA

The WSA is installed in a JSE as a single Web application that you can instantiate as one or more servlets (WSA instances). Each WSA instance provides a single point of management for one or more deployed SOAP Web services. Thus, to begin working with the WSA in OpenEdge Management or OpenEdge Explorer:

1. Create a new WSA instance if none exists or you need another, and configure it for operation. (The WSA comes installed with one local WSA instance, wsa1.)

2. For a WSA instance where you want to deploy a new SOAP Web service, optionally review and update the default property settings that govern the run-time behavior of new SOAP Web service deployments. You can also adjust these settings for each SOAP Web service individually after you deploy it.

3. If you want to import an existing SOAP Web service, first export the SOAP Web service from the context of the WSA instance where it is currently hosted.

4. Deploy the new SOAP Web service or import the existing SOAP Web service into the context of a WSA instance that you have configured.

5. As you client-test the SOAP Web service, review the WSA instance status and statistics, and update any WSA instance properties as needed. Such changes affect all SOAP Web services deployed to the same WSA instance.

Note: To apply changes to most WSA properties, you must restart the JSE where the WSA is installed. You can make temporary changes to selected WSA properties at run time without restarting the JSE. However, you must update the WSA instance configuration to make these changes permanent.

6. As you client-test the SOAP Web service, review the SOAP Web service status and statistics, and update any SOAP Web service properties as needed. Such changes affect only the selected SOAP Web service. For most of these changes, you must disable the SOAP Web service before you make them, then enable the SOAP Web service afterward to evaluate the effect.

SOAP Web service session models and the WSA

SOAP Web service applications in OpenEdge are designed to run in one of two AppServer session models:

- Session-Managed
- Session-Free

The session model of a SOAP Web service application significantly affects how the WSA manages the SOAP Web service at run time. In any case, the session model specified for a SOAP Web service (when defined using ProxyGen) must match the session model of the AppServer to which it is bound. Otherwise, the WSA returns an error when a client attempts to access the SOAP Web service.
Session-managed SOAP Web services

The WSA handles session-managed SOAP Web services in exactly the same way that other OpenEdge clients access an AppServer running in state-aware, state-reset, or stateless operating mode. That is, session-managed SOAP Web services are single-threaded. Thus, the WSA maintains a single network session to connect the SOAP Web service and all its objects to a single AppServer on behalf of a single SOAP Web service client. All requests from the client to the same SOAP Web service objects use the same AppServer connection. This session model provides full support on the AppServer side for context management, and makes available the capabilities of the AppServer to support transaction-oriented interactions across multiple client requests.

Session-free SOAP Web services

The WSA handles session-free SOAP Web services in exactly the same way that other OpenEdge clients access a traditional AppServer running in state-free operating mode. For session-free SOAP Web services, the WSA maintains a pool of AppServer TCP/IP connections (connection pool) for use by all clients of a single SOAP Web service. This allows multiple clients to access the same SOAP Web service concurrently; that is, session-free SOAP Web services are multi-threaded. The WSA can do this because a session-free AppServer maintains no client connections, but accepts and responds to client requests one at a time regardless of origin and with no context maintained from one request to another. This session model thus supports the industry standard for SOAP Web service interactions that focuses on small, atomic, and complete transactions in a single client request.

The connection pool maintains a list of available connections to session-free AppServers that support the same application service on behalf of all clients using the corresponding SOAP Web service. The WSA can then service as many client requests at one time as AppServer connections in the connection pool. It also queues any additional client requests for the first available AppServer that completes a client request.

Note: Session-free SOAP Web services can support SOAP Web service objects that require dedicated AppServer connections. However, typical session-free SOAP Web services avoid the use of these types of SOAP Web service objects because they undercut the benefits of connection-less interactions.
Managing Web Services Adapter (for OpenEdge SOAP Web Services) instances

You can manage Web Services Adapters (for OpenEdge SOAP Web Services) by:

- Creating a new Web Services Adapter (for OpenEdge SOAP Web Services) instance
- Configuring a Web Services Adapter (for OpenEdge SOAP Web Services) instance
- Editing a Web Services Adapter (for OpenEdge SOAP Web Services) instance configuration
- Starting Web Services Adapter (for OpenEdge SOAP Web Services) instances
- Setting SOAP Web service defaults
- Deploying a SOAP Web service to a Web Services Adapter (for OpenEdge SOAP Web Services) instance
- Importing a SOAP Web service to a Web Services Adapter (for OpenEdge SOAP Web Services) instance
- Listing the deployed SOAP Web services for a Web Services Adapter (for OpenEdge SOAP Web Services) instance
- Viewing the status of a Web Services Adapter (for OpenEdge SOAP Web Services) instance
- Viewing SOAP Web service statistics
- Changing Web Services Adapter (for OpenEdge SOAP Web Services) instance run-time properties
- Stopping Web Services Adapter (for OpenEdge SOAP Web Services) instances
- Deleting a Web Services Adapter (for OpenEdge SOAP Web Services) instance

Each of these tasks is described in the following sections.
Creating a new Web Services Adapter (for OpenEdge SOAP Web Services) instance

OpenEdge Management and OpenEdge Explorer provide a sample Web Services Adapter (WSA) instance (wsa1). You can edit the wsa1 sample configuration, and you can create new WSA instances.

**Note:** To complete the creation of a WSA instance after creating it in the management console and before you can deploy SOAP Web services to it, you must update your JSE configuration and edit the WSA's `web.xml` file. Add a servlet and servlet-mapping element for the instance, then restart your JSE. This creates the root sub-path for the new instance, initializes and runs it. For more information, see *OpenEdge Development: Web Services* and your JSE documentation.

To create a new WSA instance:

1. From the drop-down for Resources on the management console menu, click New OpenEdge Resource → WebServices Adapter. The WebServices Adapter Configuration page appears.

2. Type the name of the new Web Services Adapter (for OpenEdge SOAP Web Services) in the New Web Services name field.

**Note:** The Web Services Adapter (for OpenEdge SOAP Web Services) name is case sensitive and can include any character except a period (.) or square brackets ([ ]). The name must be unique among all configured NameServer names.

3. Type the URL in the Url field.

   This is a URL that directly addresses the WSA instance on the Internet and serves as the root for other URLs required to access SOAP Web services and other functions of the WSA.

   Form the root URL according to the following format:

   `http://host[:port] [/jse-context]/webapp-context/servlet-instance`

   Where:

   - **host** — The name of the machine where the WSA is installed (Default: localhost).
   - **port** — The HTTP port number on the WSA machine (Default: 80).
   - **jse-context** — The JSE context path name is present only when you have a Web server between your Java Servlet Engine (JSE) and the client application. It specifies the path name that you define in your Web server configuration to send client requests to your JSE.
   - **webapp-context** — The Web application context path name always exists and specifies the path name that you define in your JSE configuration to identify the WSA as a servlet application.
Managing Web Services Adapter (for OpenEdge SOAP Web Services) instances

- **servlet-instance** — The servlet-mapping name for the WSA instance (WSA Web application) specified in
  OpenEdge-install-dir\servlets\wsa\WEB-INF\web.xml. This is also the name by which the WSA instance appears in the Web Services Adapter folder in the management console.

4. Select the WSA installation location for this instance from these choices:
   - **local** — The WSA is installed on the machine where the selected AdminServer is running.
   - **remote** — The WSA is installed on a network machine that is remote from the selected AdminServer.

5. Click **Save**. The new WSA instance appears under the **Web Services Adapter** folder.

You can now configure the properties of the new instance. Note that when you create a new instance, it has a copy of the default configuration. Initially, the instance is enabled only for administration, allowing you to modify its properties and deploy SOAP Web services to it without allowing premature user access to an incomplete configuration.

### Configuring a Web Services Adapter (for OpenEdge SOAP Web Services) instance

Before deploying SOAP Web services, you must configure a Web Services Adapter (for OpenEdge SOAP Web Services) instance to host the SOAP Web services that you want to deploy or import. To configure a WSA instance, you set properties for it similar to any other Unified Broker product.

After the WSA is configured, you can import or deploy a SOAP Web service and configure additional properties that affect execution of the SOAP Web service. The SOAP Web service properties that are available for configuration depend on the session model of the SOAP Web service (session managed or session free). To verify the session model of a SOAP Web service, check its status. For details on checking SOAP Web service status, see the “Viewing the status of a SOAP Web service” section on page 217. For details about configuring SOAP Web service properties, see the “Updating a deployed SOAP Web service” section on page 219. For details about the properties available on a SOAP Web service, see OpenEdge Application Server: Administration.

You must disable a SOAP Web service before you can configure most properties on it. You can verify if the SOAP Web service is disabled by checking its status. The only properties that you can configure for an enabled SOAP Web service include:

- **serviceFaultLevel**
- **serviceLoggingLevel**

If you set other properties while the SOAP Web service is enabled, the property value changes take effect only after you disable and then enable the SOAP Web service again. For details about enabling or disabling a SOAP Web service, see the “Enabling or disabling a SOAP Web service for client access” section on page 216.
To configure a Web Services Adapter (for OpenEdge SOAP Web Services) instance, you can view or edit properties in the following categories:

- Location
- Proxy Server Setting
- WSDL
- Logging Setting
- Security
- Advanced Features

You can also create a new Web Services Adapter (for OpenEdge SOAP Web Services) instance, view its status, or delete an instance you no longer need.

Each of these tasks is described in the following sections.

**Setting Web Services Adapter (for OpenEdge SOAP Web Services) instance properties**

You can set location, proxy server setting, WSDL, logging setting, security, and advanced features properties for a Web Services Adapter (for OpenEdge SOAP Web Services) instance.

To set WebServices Adapter instance properties:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the WebServices Adapter broker instance whose properties you want to set.

3. Click the WebServices Adapter broker instance. The **WebServices Adapter** details page appears.

4. If your Web server requires that you log in, click **Login** in the Command and control section of the page. Type your user name and your password, and then click **Submit**.

5. In the Command and control section of the page, click **Configuration**. The **WebServices Adapter Configuration** page appears.

From this page, you can:

- View a read-only display of the Web Services Adapter (for OpenEdge SOAP Web Services) properties as described in **Table 51** through **Table 56**. All of these tables follow this procedure.

- Click **Edit** to modify the properties. For details, see the “Editing a Web Services Adapter (for OpenEdge SOAP Web Services) instance configuration” section on page 204.
Location properties

You set the properties, shown in Table 51, in this category when you first create the WSA instance. For more information see the “Creating a new Web Services Adapter (for OpenEdge SOAP Web Services) instance” section on page 192.

You cannot change the location properties. Instead, you must delete and then recreate the WSA instance with the new values. For more information, see the “Deleting a Web Services Adapter (for OpenEdge SOAP Web Services) instance” section on page 215.

Table 51: Web Services Adapter (for OpenEdge SOAP Web Services) Location properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location</td>
<td>A read-only property that indicates where the WSA is installed:</td>
</tr>
<tr>
<td></td>
<td>• <strong>local</strong> — The WSA is installed on the machine where the selected AdminServer is running. You can configure all the properties of a local WSA.</td>
</tr>
<tr>
<td></td>
<td>• <strong>remote</strong> — The WSA is installed on a network machine that is remote from the selected AdminServer.</td>
</tr>
<tr>
<td>URL</td>
<td>A read-only property set to the base URL for the WSA.</td>
</tr>
<tr>
<td>Web Service Protocol Value</td>
<td>The protocol for all SOAP Web service URLs inserted into WSDL documents that are generated for this WSA instance during SOAP Web service deployment. The host, port, and other path information specified for the WSA URL when the instance was originally created remains unchanged. You can select from the following values in the drop-down list:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Duplicate WSA Url value</strong> — Uses the original protocol specified for the WSA URL</td>
</tr>
<tr>
<td></td>
<td>• <strong>Use HTTP protocol</strong> — Replaces the original WSA URL protocol with HTTP</td>
</tr>
<tr>
<td></td>
<td>• <strong>Use HTTPS protocol</strong> — Replaces the original WSA URL protocol with HTTPS</td>
</tr>
</tbody>
</table>

Installed Default: **Duplicate WSA Url value**
Proxy Server Setting properties

The properties in this category, as shown in Table 52, apply to both local and remote WSA instances. They specify the information required to access any proxy server for the network on which the WSA resides.

Table 52: Web Services Adapter (for OpenEdge SOAP Web Services) Proxy Server Setting properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proxy host</td>
<td>The TCP host name for the proxy server</td>
</tr>
<tr>
<td></td>
<td>Installed Default: None</td>
</tr>
<tr>
<td>Proxy port</td>
<td>The TCP port number of the proxy service</td>
</tr>
<tr>
<td></td>
<td>Installed Default: 0</td>
</tr>
<tr>
<td>Proxy username</td>
<td>A valid username required to access the proxy server</td>
</tr>
<tr>
<td></td>
<td>Installed Default: None</td>
</tr>
<tr>
<td>Proxy password</td>
<td>The password for the specified username</td>
</tr>
<tr>
<td></td>
<td>Installed Default: None</td>
</tr>
<tr>
<td>Proxy password Confirm</td>
<td>The password retyped for confirmation</td>
</tr>
</tbody>
</table>

WSDL

The properties in this category, as shown in Table 53, control access to WSDL files and information.

Table 53: Web Services Adapter (for OpenEdge SOAP Web Services) WSDL properties (1 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable WSDL retrieval</td>
<td>If selected, this check box setting allows network users to access WSDL files deployed to this WSA instance. Otherwise, users cannot access any WSDL files deployed to this WSA instance.</td>
</tr>
<tr>
<td></td>
<td>Installed Default: Activated</td>
</tr>
<tr>
<td>WSDL inaccessible page</td>
<td>The relative path and filename of the Web page that you want the WSA instance to return when users attempt to access WSDL files deployed to this instance. The Enable WSDL Retrieval check box is cleared to prevent that access. The path name is relative to your Web application (WSA) servlet root directory.</td>
</tr>
<tr>
<td></td>
<td>Installed Default: None</td>
</tr>
</tbody>
</table>
Managing Web Services Adapter (for OpenEdge SOAP Web Services) instances

Logging Setting

The properties in this category, as shown in Table 54, control the location and content of the WSA log file.

Table 53: Web Services Adapter (for OpenEdge SOAP Web Services) WSDL properties (2 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable WSDL listing retrieval</td>
<td>If selected, this check box setting allows network users to access a Web page that lists the WSDL files deployed to this WSA instance. Otherwise, users cannot access the list of WSDL files deployed to this WSA instance. Installed Default: Activated</td>
</tr>
<tr>
<td>WSDL listing page</td>
<td>The path and filename of the Web page that you want the WSA instance to return when users request a listing of the WSDL files deployed to this instance. The WSA returns this page only when the <strong>Enable WSDL listing retrieval</strong> check box is selected. The path name is relative to your Web application (WSA) servlet root directory. Installed Default: <strong>WSDLListing.html</strong></td>
</tr>
</tbody>
</table>

Table 54: Web Services Adapter (for OpenEdge SOAP Web Services) Logging Setting properties (1 of 3)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Working directory</td>
<td>The directory where the WSA writes temporary files when required. Enter a valid path name.</td>
</tr>
<tr>
<td>Log filename</td>
<td>The WSA log filename. Enter a valid path name.</td>
</tr>
</tbody>
</table>
Chapter 6: Configuring OpenEdge SOAP Web Services

Table 54: Web Services Adapter (for OpenEdge SOAP Web Services) Logging Setting properties (2 of 3)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logging level</td>
<td>A value that specifies the amount of information that is written to the WSA log. Select from the following values in the drop-down list:</td>
</tr>
<tr>
<td></td>
<td>• None — Log no entries. This is equivalent to turning logging off.</td>
</tr>
<tr>
<td></td>
<td>• Error Only — Log OpenEdge error messages. This includes all error messages and is unrelated to the entry types specified. Errors continue to be logged at all higher levels.</td>
</tr>
<tr>
<td></td>
<td>• Basic — Logging Entry Types determine the logged information. Each entry type generates at least some output. This is the default.</td>
</tr>
<tr>
<td></td>
<td>• Verbose — Logging entry types determine the logged information, but it is typically more information than Basic.</td>
</tr>
<tr>
<td></td>
<td>• Extended — Logging entry types determine the logged information, but it is typically more information than Verbose.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> A value of Other in this field indicates an option manually specified in the ubroker.properties file that provides additional diagnostic logging features. You can reset this by choosing from the values in the drop-down list.</td>
</tr>
<tr>
<td></td>
<td>Installed Default: <strong>Basic</strong></td>
</tr>
<tr>
<td>Append to log file</td>
<td>A check box that indicates whether or not a new server log file should be created when the WSA is started.</td>
</tr>
<tr>
<td></td>
<td>To create a new WSA log file each time the WSA is started, even if the WSA log file already exists, clear the <strong>Append to log file</strong> check box.</td>
</tr>
<tr>
<td></td>
<td>To append log entries to the existing WSA log file, select the <strong>Append to log file</strong> check box.</td>
</tr>
<tr>
<td></td>
<td>Installed Default: <strong>Selected</strong></td>
</tr>
<tr>
<td>Log message threshold size</td>
<td>An integer value that limits the amount of SOAP message detail written to the log file for clients specified using the Debug Clients property in the Advanced Features category. A value of -1 indicates that all SOAP message detail is written to the file. A positive value indicates the number of characters written to the file for each logged SOAP message. Practical values range from 100 to 10000, but any positive value is valid.</td>
</tr>
<tr>
<td></td>
<td>Installed Default: <strong>-1</strong></td>
</tr>
</tbody>
</table>
### Table 54: Web Services Adapter (for OpenEdge SOAP Web Services) Logging Setting properties (3 of 3)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logging entry types</td>
<td>A comma-separated list of valid entry types for WSA logging. The only standard entry is WSADefault. Set additional entry types only if instructed to do so by Progress Software Corporation Technical Support. You can specify a separate logging level for each log entry type, and you can use a wildcard (*) to specify multiple entry types. For example: <code>UBroker.Basic:3,UBroker.ThreadPool:4, UBroker.Client*</code></td>
</tr>
<tr>
<td>Log file threshold size</td>
<td>A value of 0 or a value between 500,000 and 2,147,483,647, where 0 means there is no limit on the log file size other than what the operating system imposes.</td>
</tr>
<tr>
<td>Maximum number of log files</td>
<td>The number of rolled-over log files to keep. The value can be 0 or a value between 2 and 999999, where 0 means there is no limit on the number of WSA log files to keep. The specified number represents the maximum number of log files to keep on disk at any time, including the current log file being written to. The default value is 3. When the file becomes equal to or greater than the Log file threshold size, the client process renames it and creates a new log file. The file is renamed as follows, where ##### is a number starting at 000001 and increasing to 999999, after which it rolls back over to 000001: <code>filename.#####.extension</code></td>
</tr>
</tbody>
</table>

---

*OpenEdge® Management and OpenEdge Explorer: Configuration* 199
Security

The properties in this category, as shown in Table 55, control options for authorizing and authenticating access to WSA functions and SOAP Web services.

Table 55: Web Services Adapter (for OpenEdge SOAP Web Services) Security properties (1 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable WSA administration</td>
<td>If the check box is selected, indicates that administration functions are available for the selected WSA instance.</td>
</tr>
<tr>
<td></td>
<td>Installed Default: Selected</td>
</tr>
<tr>
<td>WSA administration authentication</td>
<td>A drop-down list that indicates the criteria used to authenticate users for access to WSA administrative functions. The choices include:</td>
</tr>
<tr>
<td></td>
<td>• No user authentication — Any user can perform WSA administration functions.</td>
</tr>
<tr>
<td></td>
<td>• Use Admin Server user authentication — The AdminServer forwards the same user credentials supplied at connection time to the Web server (or JSE) where the WSA is running in order to perform administration functions.</td>
</tr>
<tr>
<td></td>
<td>• Prompt user for Web Server username and password — Only users authorized to manage the Web server (or JSE) where the WSA is running can perform WSA administration functions.</td>
</tr>
<tr>
<td>Require WSA administration authorization</td>
<td>If the check box is selected, indicates that each user should be validated against selected Administration Role definitions when the user attempts to perform an administrative function on the WSA instance or on any of its deployed SOAP Web services.</td>
</tr>
<tr>
<td></td>
<td>For details, see the “Viewing or modifying the Admin or Operator role” section on page 201.</td>
</tr>
<tr>
<td>To enable administration roles, select from the following list</td>
<td>Predefined Roles: PSCAdmin, PSCOper</td>
</tr>
<tr>
<td></td>
<td>For details, see the “Viewing or modifying the Admin or Operator role” section on page 201.</td>
</tr>
<tr>
<td>Enable Web Services</td>
<td>If the check box is selected, indicates that users can access the SOAP Web services deployed to the selected WSA instance.</td>
</tr>
<tr>
<td></td>
<td>Installed Default: Cleared</td>
</tr>
</tbody>
</table>
You can define what level of access is allowed for the Web Services Adapter (for OpenEdge SOAP Web Services) instance: Administrator or Operator. Under **Admin Role Links**, click either **AdminRole.PSCAdmin** or **AdminRole.PSCOpr** to see the current read, write, delete, and execute settings for the following:

- **Application defaults** — Governs which operations the role may perform on the default SOAP Web service properties that are used to initialize a newly deployed SOAP Web service
- **Application enablement** — Governs which operations the role may perform on the enable state of SOAP Web service applications
- **Application properties** — Governs which operations the role may perform on a SOAP Web service’s properties
- **Application statistics** — Governs which operations the role may perform on a SOAP Web service’s statistical counters
- **Servlet properties** — Governs which operations the role may perform on the WSA’s run-time changeable properties
- **Servlet services** — Governs which operations the role may perform on the registry of deployed SOAP Web service applications
- **Servlet statistics** — Governs which operations the role may perform on the WSA’s statistical counters

To edit the settings, click **Edit**. To create a new AdminRole name, click **Create**.
Advanced Features properties

The properties in this category, shown in Table 56, support less commonly-used or special-purpose features, some of which might relate to the properties of other categories.

Table 56: Web Services Adapter (for OpenEdge SOAP Web Services)
Advanced Features properties (1 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>WSA administration SOAP action</td>
<td>A string placed in the SOAPAction HTTP header when the AdminServer forwards administration operations to the WSA. The SOAPAction HTTP header is required for all HTTP messages that carry SOAP messages and is used by intervening security servers (such as firewalls) to determine if each HTTP message is allowed to pass through to its destination. The installed default is a Universal Resource Number URN identifying the WSA administration SOAP Web service, but you can also specify any string required by intervening security servers on the network. Installed Default: urn:services-progress-com:wsa-admin:01</td>
</tr>
</tbody>
</table>
### Table 56: Web Services Adapter (for OpenEdge SOAP Web Services) Advanced Features properties (2 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP error page</td>
<td>The path and filename of a static Web page that you want returned to a SOAP Web service user if the request results in an HTTP error, such as a service unavailable or an authorization error. The path name is relative to your web application (WSA) servlet root directory.</td>
</tr>
<tr>
<td></td>
<td>Installed Default: <code>httperror.html</code></td>
</tr>
<tr>
<td>Debug clients</td>
<td>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. The WSA dumps each SOAP message up to the limit specified by the Log message threshold property setting in the Logging Setting category.</td>
</tr>
<tr>
<td></td>
<td>To add a client machine to the list, enter its dot-formatted IP address in the Client Host field, then choose the <strong>Add</strong> button to add it to the list. You can also enter multiple, comma-separated addresses in the field at one time to dump messages for multiple clients, or enter a single asterisk (*) to dump the messages for all clients.</td>
</tr>
<tr>
<td></td>
<td>To remove whole lines of addresses from the list, select one or more lines in the list and click <strong>Delete</strong>.</td>
</tr>
<tr>
<td></td>
<td>To edit a single line of addresses, select the line in the list, and after modifying the line click <strong>Add</strong> to add it to the list. The original line remains. Select the original line and click <strong>Delete</strong> to remove it from the list and leave the edited line in its place.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> The list automatically adds new lines in increasing order of collation (numerical, for valid IP addresses). If you add an invalid IP address to the list, the WSA never matches it with a message to dump in the log file, and thus effectively ignores it.</td>
</tr>
<tr>
<td></td>
<td>Installed Default: None</td>
</tr>
</tbody>
</table>
**Editing a Web Services Adapter (for OpenEdge SOAP Web Services) instance configuration**

You can edit the configuration properties of a Web Services Adapter (WSA) (for OpenEdge SOAP Web Services) instance from the management console.

**To edit Web Services Adapter (for OpenEdge SOAP Web Services) properties:**

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the WebServices Adapter broker instance whose properties you want to edit.
3. WebServices Adapter broker instance. The **WebServices Adapter** details page appears.
4. In the **Command and control** section of the page, click **Configuration**. The **WebServices Adapter Configuration** page appears.
5. Click **Edit**, and make the changes you want to the properties. (For details about the properties, see Table 51 through Table 56.)
6. Click **Save**. Any changes you make are also reflected in the `ubroker.properties` file.

You can also make run-time changes in the management console to the values of the following WSA properties:

- **debugClients** — Controls whether to dump HTTP and SOAP protocol message information to the WSA log file, and specifies the clients (if any) whose SOAP messages are to be dumped
- **enableWsdl** — Controls the WSA's ability to respond to inquiries about WSDL documents for any of the SOAP Web service applications it hosts
- **enableWsdlListings** — Controls the WSA's ability to respond to a request for a list of WSDL documents it has available
- **loggingLevel** — Specifies the amount of information to be written to the broker log
- **logMsgThreshold** — Controls the amount of SOAP message information that can be written to the WSA log file
- **webAppEnabled** — Controls the WSA's ability to accept and process SOAP requests to any of its hosted SOAP Web service applications

These changes last only for the duration of the current Java servlet engine (JSE) session. To make these changes permanent across sessions, you must make the changes in the property settings.
To edit the WSA instance configuration:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the WebServices Adapter broker instance whose configuration you want to edit.

3. Click the WebServices Adapter broker instance. The **WebServices Adapter** details page appears.

4. In the **Command and control** section of the page, click **Configuration**. The **WebServices Adapter Configuration** page appears.

5. Click **Edit**.

6. Click **Save**.

Starting Web Services Adapter (for OpenEdge SOAP Web Services) instances

The Web Services Adapter (WSA) (for OpenEdge SOAP Web Services) and its instances start up automatically when you start your Web server or Java servlet engine (JSE). For more information, see the documentation for your Web server or JSE.
Setting SOAP Web service defaults

You can view and set the default values used to initialize properties for SOAP Web services that you deploy. Any changes to these defaults affect only SOAP Web services that you deploy after confirming the change. To set the values of properties for deployed SOAP Web services, set the property values individually for each SOAP Web service.

Notes: Unlike WSA instance properties, the defaults for SOAP Web service properties are not stored in the ubroker.properties file. They are stored in the default.props XML file located in the Web application servlet directory for each WSA instance.

To view and set SOAP Web service defaults:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the WebServices Adapter broker instance whose default values you want to view.
3. Click the WebServices Adapter broker instance. The WebServices Adapter details page appears.
4. In the Command and control section of the page, click Defaults. The WebServices Adapter Defaults page appears.
5. Click Edit.
6. Select the property whose default value you want to change. For details on the available properties, see OpenEdge Application Server: Administration.
7. Type the new value in the available field. For logical values, typing any value other than true results in a setting of false.
8. Click Submit to change the default value.

To return to the default values, you can click Reset from the Defaults page.
Deploying a SOAP Web service to a Web Services Adapter (for OpenEdge SOAP Web Services) instance

You can deploy a SOAP Web service to the context of any available WSA instance using the Web Service Mapping (WSM) file generated using ProxyGen. For more information on generating a WSM file, see OpenEdge Development: Web Services.

A deployed SOAP Web service receives its initial property values from the default.props file for the WSA instance. You can either change these default values before you deploy the SOAP Web service, or you can change the deployed SOAP Web service property values after you deploy the SOAP Web service. A SOAP Web service always deploys with a disabled status to prevent premature or unintended client access.

To deploy a SOAP Web service:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the WebServices Adapter broker instance where you want to deploy the SOAP Web service.

3. Click the WebServices Adapter broker instance. The WebServices Adapter details page appears.

4. In the Command and control section of the page, click Deploy.

5. Type the path and name of the WSM file for the SOAP Web service that you want to deploy.

6. Click Submit. Verify that the deployment information shown (and which is based on information in the WSM file) is correct. You can modify information in the following fields:

   - **WSM Filename** — The path and name of the WSM file.

   - **Name** — The name used to identify the SOAP Web service in the management console and to name the SOAP Web service files that are deployed to the WSA instance.

   - **Web Service Namespace** — Any value you choose to uniquely qualify the names for operations and parameters used to define the SOAP Web service.

   - **SOAP Action** — A string that the client application is required to place in the SOAPAction HTTP header when accessing a SOAP Web service hosted by the WSA instance. The SOAPAction HTTP header is required for all HTTP messages that carry SOAP messages and is used by intervening security servers (such as firewalls) to determine if each HTTP message is allowed to pass through to its destination. The default is a blank string, but it can be any string required by the intervening security servers on the network.

   - **Append to SOAP Action** — Indicates whether to append the specified SOAP Action value to any other SOAP Action values generated for the SOAP Web service messages by Web service clients.
Chapter 6: Configuring OpenEdge SOAP Web Services

- **WSDL Style/Use** — Specifies the SOAP format to use when sending or receiving SOAP messages for this SOAP Web service. The value that you choose is dependent on what your anticipated SOAP Web service clients can support and can be any of the following: **RPC/Encoded, RPC/Literal, and Document/Literal**.

7. Click **Submit** to deploy the SOAP Web service with these settings. A confirmation message appears, and the WSM file name is identified in the **Deployed Web Services** section of the WSA instance’s **Details** page.

You might need to document additional information about these settings, especially **WSDL Style/Use**, for use by SOAP Web service client developers. For more information, see the chapter on client requirements in *OpenEdge Development: Web Services*.

Deployment generates the following files in the WSA instance directory:

- **WebServiceFriendlyName.props** — An XML file containing the current SOAP Web service property settings (initially set from **default.props**)
- **WebServiceFriendlyName.wsad** — The Web Service Adapter Descriptor (WSAD) XML file that defines the SOAP Web service to the WSA instance
- **WebServiceFriendlyName.wsdl** — The Web Services Description Language (WSDL) XML file that defines the SOAP Web service to potential SOAP Web service clients

For more information about these files, see *OpenEdge Development: Web Services*.

**Caution:** Once you deploy a SOAP Web service during development, you can change its definition and deployment information using a SOAP Web service update. However, once you deploy and enable a SOAP Web service for client access under production conditions, avoid making any changes to this information, as client implementations depend on its stability.

After production deployment, to make the same SOAP Web service available using different information (for example, to add a new operation or use a different **WSDL Style/Use**), deploy a new SOAP Web service with the new information, using a different SOAP Web service name in the same WSA context. You could also deploy the SOAP Web service to a different WSA context.

However, you can change the run-time properties of a deployed SOAP Web service at any time.
Exporting a SOAP Web service

You can export an existing SOAP Web service definition to save its deployment information, properties, and other settings. This creates a Web Service Definition (WSD) XML file with a name in the form filename.wsdl, with a filename that you specify. You can use this file to copy (import) the SOAP Web service to another WSA instance or to save it as a backup for the SOAP Web service definition.

To export a SOAP Web service:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the WebServices Adapter broker instance where the SOAP Web service you want to export is deployed.

3. Click the WebServices Adapter broker instance. The WebServices Adapter details page appears.

4. In the Deployed Web Services section of the page, click Export.

5. Type the path and file name, ending in .wsdl, of the file for the SOAP Web service you want to export.

6. Click Submit. A confirmation message appears.
Importing a SOAP Web service to a Web Services Adapter (for OpenEdge SOAP Web Services) instance

You can import a SOAP Web service to any available WSA instance using the Web Service Description (WSD) file saved during a previous export of the SOAP Web service. The import function generates the appropriate SOAP Web service properties ([WebServiceFriendlyName].props), WSAD, and WSDL files to duplicate the original SOAP Web service in the context of the selected WSA instance. However, a SOAP Web service always imports with a disabled status to prevent premature or unintended client access.

To import a SOAP Web service:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the WebServices Adapter broker instance where you want to import the SOAP Web service.
3. Click the WebServices Adapter broker instance. The WebServices Adapter details page appears.
4. In the Command and control section of the page, click Import.
5. Type the path name of the .wsd file for the SOAP Web service you want to import.
6. Choose Submit. Verify that the information shown (and which is based on information in the WSD file) is correct. You can modify information in the following fields:
   - Name — The name used to identify the SOAP Web service and to name the SOAP Web service files that are deployed to the WSA instance.
   - Web Service Namespace — Any value you choose to uniquely qualify the names for operations and parameters used to define the SOAP Web service.
   - SOAP Action — A string that the client application is required to place in the SOAPAction HTTP header when accessing a SOAP Web service hosted by the WSA instance. The SOAP Action HTTP header is required for all HTTP messages that carry SOAP messages and is used by intervening security servers (such as firewalls) to determine if each HTTP message is allowed to pass through to its destination. The default is a blank string, but can be any string required by the intervening security servers on the network.
   - Append to SOAP Action — Indicates whether to append the specified SOAP Action value to any other SOAP Action values generated for the SOAP Web service messages by SOAP Web service clients.
   - WSDL Style/Use — Specifies the SOAP format to use when sending or receiving SOAP messages for this SOAP Web service. Supported formats are RPC/Encoded, RPC/Literal, and Document/Literal.
7. Click Submit. A confirmation message appears.
Listing the deployed SOAP Web services for a Web Services Adapter (for OpenEdge SOAP Web Services) instance

You can list all SOAP Web services in the management console that are deployed to a selected WSA instance.

To list all SOAP Web services deployed to a WSA instance:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the WebServices Adapter broker instance whose deployed SOAP Web services you want to see.
3. Click the WebServices Adapter broker instance. The WebServices Adapter details page appears.
4. Under the Command and control section, click List. The list of all SOAP Web services deployed to the WSA instance appears.

Viewing the status of a Web Services Adapter (for OpenEdge SOAP Web Services) instance

You can view the execution status of a selected WSA instance to see if the instance is running. If the instance is running, the status indicates if administration, SOAP Web service application, or WSDL document access is enabled.

To view the status of a WSA instance:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the WebServices Adapter broker instance whose status you want to view.
3. Click the WebServices Adapter broker instance. The WebServices Adapter details page appears.
4. Under Operations views, click Status. The following status information appears:
   - Whether the WSA instance is running
   - Whether access to administrative functions, SOAP Web service applications (by clients), and WSDL document retrieval is enabled
Viewing Web Services Adapter (for OpenEdge SOAP Web Services) instance statistics

You can view (and reset) run-time statistics for a selected WSA instance. Each WSA instance creates and begins accumulating these statistics (Start Time) when it is first started by the Java Servlet Engine (Creation Time). You can also reset the statistics at any point during WSA execution, so that the statistics begin accumulating again from a new Start Time.

To view the statistics of a WSA instance:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the WebServices Adapter broker instance whose statistics you want to view.
3. Click the WebServices Adapter broker instance. The WebServices Adapter details page appears.
4. Under Operations views, click Statistics. The following information appears:
   - **Number of SOAP Requests** — Requests for SOAP Web service applications
   - **Number of Active Requests** — Requests queued and being acted upon
   - **Number of HTTP Requests** — Total requests received from the HTTP listener, including administrative, WSDL, and SOAP Web service requests
   - **Number of WSDL Requests** — Requests for WSDL documents
   - **Number of SOAP Faults** — Error returns from SOAP Web service applications
   - **Number of Services Disabled** — SOAP Web services deployed to this WSA that are disabled from client access
   - **Number of Errors** — Total errors returned by the WSA, with error counts broken out at the bottom of the list for each of several error categories when total errors are greater than zero (0)
5. To reset the statistics and start accumulating all statistics from zero, click Reset and confirm that you want to restart the statistics.
Viewing statistics for deployed SOAP Web services

You can view statistical details for deployed SOAP Web services.

To view statistics for deployed SOAP Web services:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the WebServices Adapter broker instance whose runtime statistics you want to view.

3. Click the WebServices Adapter broker instance. The WebServices Adapter details page appears.


You can reset the statistical display by clicking Reset on the adapter instance’s Statistics page.
Changing Web Services Adapter (for OpenEdge SOAP Web Services) instance run-time properties

You can temporarily change some Web Services Adapter (WSA) (for OpenEdge SOAP Web Services) instance properties at run time without restarting your Java servlet engine (JSE). This is most useful for testing and debugging. The next time you restart your JSE, these settings revert to the current configuration settings for these properties in the ubroker.properties file.

To change WSA instance runtime properties:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the WebServices Adapter broker instance whose runtime properties you want to change.

3. Click the WebServices Adapter broker instance. The WebServices Adapter details page appears.


5. Click Edit to modify any of the following properties:

   - adminEnabled — Specifies whether administrator privileges are in effect, either true or false.

   - debugClients — Controls whether to dump HTTP and SOAP protocol message information to the WSA log file, and specifies the clients (if any) whose SOAP messages are to be dumped.

   - enableWsdl — Controls the WSA's ability to respond to inquiries about WSDL documents for any of the SOAP Web service applications it hosts.

   - enableWsdlListings — Controls the WSA's ability to respond to a request for a list of WSDL documents it has available.

   - loggingLevel — Specifies the amount of information to be written to the broker log.

   - logMsgThreshold — Controls the amount of SOAP message information that can be written to the WSA log file.

   - webAppEnabled — Controls the WSA's ability to accept and process SOAP requests to any of its hosted SOAP Web service applications.

6. Click Submit.

Stopping Web Services Adapter (for OpenEdge SOAP Web Services) instances

The Web Services Adapter (for OpenEdge SOAP Web Services) (WSA) and its instances stop execution automatically when you shut down your Web server or Java servlet engine (JSE). For more information, see the documentation for your Web server or JSE.
Deleting a Web Services Adapter (for OpenEdge SOAP Web Services) instance

You can delete any Web Services Adapter (WSA) (for OpenEdge SOAP Web Services) instance listed in the management console as long as it has no SOAP Web services deployed to it. Undeploy all of its deployed SOAP Web services before you attempt to delete the instance.

For more information, see *OpenEdge Development: Web Services* and your JSE documentation.

**Note:** To completely remove a WSA instance after deleting it in the management console, you must also remove its corresponding servlet and servlet-mapping elements from the WSA's `web.xml` file and restart your Web server or Java Servlet Engine (JSE).

To delete a Web Services Adapter (for OpenEdge SOAP Web Services) instance:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the WebServices Adapter broker instance you want to delete.
3. Click the WebServices Adapter broker instance. The **WebServices Adapter** details page appears.
4. Click **Delete**, and then click **OK** to confirm the deletion.
Working with SOAP Web services

You can perform the following SOAP Web services administration tasks from the management console:

- Enabling or disabling a SOAP Web service for client access
- Viewing the status of a SOAP Web service
- Viewing SOAP Web service statistics
- Updating a deployed SOAP Web service
- Exporting a SOAP Web service
- Undeploying a SOAP Web service

Enabling or disabling a SOAP Web service for client access

You can enable any deployed SOAP Web service, making it accessible or inaccessible to user requests on the network.

To enable or disable a SOAP Web service for client access:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the WebServices Adapter broker instance where the SOAP Web service you want to enable or disable is deployed.
3. Click the WebServices Adapter broker instance. The WebServices Adapter details page appears.
4. In the Deployed Web Services section, click Status Enablement. The list of all SOAP Web services deployed to the WSA instance appears.
5. Click Enable or Disable.
6. Click OK. A confirmation message appears.
Viewing the status of a SOAP Web service

You can view status information on a deployed SOAP Web service.

To view the status of a SOAP Web service:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the WebServices Adapter broker instance where the SOAP Web service whose status you want to view is deployed.

3. Click the WebServices Adapter broker instance. The **WebServices Adapter** details page appears.

4. Under **Deployed Web Services**, click **Status Enablement**.

This read-only status presents the following information in a list of label-value pairs separated by a colon:

- **Target NameSpace** — The target name space identifier, typically a Universal Resource Name (URN)
- **Status** — Indicates if the SOAP Web service is enabled or disabled for client access
- **AppServerInfo** — The AppServer connection information, expressed as a URL
- **Session Model** — Indicates if the SOAP Web service requires an AppServer configured with session-managed or session-free AppServer operating modes
- **Style/Use** — SOAP message style (RPC or Document)

In addition, the status includes the a list of current application properties.
Viewing SOAP Web service statistics

You can view (and reset) run-time statistics for a selected SOAP Web service. Each SOAP Web service creates and begins accumulating these statistics (Start Time) from the time that its parent WSA instance starts up (Creation Time). You can reset the SOAP Web service statistics at any point while the WSA instance is running. A reset clears and begins accumulating the statistics again with a new Start Time.

To view the run-time statistics for a SOAP Web service:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the WebServices Adapter broker instance where the SOAP Web service whose runtime statistics you want to view is deployed.

3. Click the WebServices Adapter broker instance. The WebServices Adapter details page appears.

4. Under Deployed Web Services, click Statistics. A read-only statistics presentation with the following information appears:
   - **Number of Requests** — Total requests handled by the SOAP Web service
   - **Number of Active Requests** — Requests currently being serviced by the SOAP Web service
   - **Number of SOAP Faults** — Errors returned by the SOAP Web service application
   - **Number of Objects Not Found** — Calls to unidentified SOAP Web service methods
   - **Number of Errors** — Total errors returned by the SOAP Web service, with error counts broken out at the bottom of the list for each of several error categories when the total errors in each category are greater than 0
   - **Number of AppServer Connections** — Total AppServer connections in the connection pool for this SOAP Web service
   - **Number of AppObjects** — Total AppObjects registered for all clients of this SOAP Web service (only one AppObject is shared by all clients of a session-free SOAP Web service; no AppObjects are ever shared for a session-managed SOAP Web service)
   - **Number of SubAppObjects** — Total SubAppObjects registered for all clients of this SOAP Web service (SubAppObjects never shared)
   - **Number of ProcObjects** — Total ProcObjects registered for all clients of this SOAP Web service (ProcObjects never shared)

5. To reset the statistics and start accumulating from zero, click Reset and confirm that you want to restart the statistics.

6. Click OK on the confirmation message.
Updating a deployed SOAP Web service

You can update the definition and deployment information for a deployed SOAP Web service at any time.

Caution: You can freely update the definition and deployment information for a deployed SOAP Web service during development. However, once you deploy and enable a SOAP Web service for client access under production conditions, avoid making any changes to this information, as client implementations depend on its stability.

After production deployment, to make the same SOAP Web service available using different information (for example, to add a new operation or use a different SOAP Message Style), deploy a new SOAP Web service with the new information, using a different SOAP Web service name in the same WSA context. You can also deploy the SOAP Web service to a different WSA context.

Updating the definition and deployment information

You must disable a SOAP Web service before you can update it. You can verify if the SOAP Web service is disabled by checking its status.

Once you enable the SOAP Web service for client access, avoid making any changes to this deployment information, as client implementations depend on its stability. To make the same SOAP Web service available using different information (for example, to add a new operation or use a different WSDL Style/Use), deploy a new SOAP Web service for the application with a different name or in a different WSA context, and with the new information you want to use.

To update a SOAP Web service:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the WebServices Adapter broker instance where the SOAP Web service you want to update is currently deployed.

3. Click the WebServices Adapter broker instance. The WebServices Adapter details page appears.

4. Under Deployed SOAP Web services, click Update.

5. Type the path and name of the WSM file for the SOAP Web service that you want to update.

6. Click Submit to update deployment information. The SOAP Web service name for an update is read-only and preset from the initial deployment. Initial values for other fields are set from existing information in the new WSM file, and you can change them as necessary. The fields that you can update include:

   • SOAP Web service Namespace — Any value you choose to uniquely qualify the names for operations and parameters used to define the SOAP Web service.
• **SOAP Action** — A string that the client application is required to place in the SOAPAction HTTP header when accessing a SOAP Web service hosted by the WSA instance. The SOAPAction HTTP header is required for all HTTP messages that carry SOAP messages and is used by intervening security servers (such as firewalls) to determine if each HTTP message is allowed to pass through to its destination. The default is a blank string, but can be any string required by the intervening security servers on the network.

• **Append to SOAP Action** — Indicates whether or not to append the specified SOAP Action value to any other SOAP Action values generated for the SOAP Web service SOAP messages by SOAP Web service clients.

• **WSDL Style/Use** — Specifies the SOAP format to use when sending or receiving SOAP messages for this SOAP Web service. The value that you choose is dependent on what your anticipated SOAP Web service clients can support.

7. Click **Submit**. A confirmation message appears.

You might need to document additional information about the updated SOAP Web service for use by SOAP Web service client developers. For more information, see the chapter on client requirements in *OpenEdge Development: Web Services*.

**Setting SOAP Web service application properties**

You can view and set the value of each property for a deployed SOAP Web service, or you can reset the values of all properties to the current SOAP Web service defaults (stored in the default.props file for the WSA instance where the SOAP Web service is deployed).

You must disable a SOAP Web service before you can set most properties for it. You can verify if the SOAP Web service is disabled by checking its status. The only properties that you can set for an enabled SOAP Web service are:

• **serviceFaultLevel**

• **serviceLoggingLevel**

If you set other properties while the SOAP Web service is enabled, the property value changes take effect only after you disable and then enable the SOAP Web service again.

**Note**: Unlike WSA instance properties, the values for SOAP Web service properties are not stored in the ubroker.properties file. They are stored in the WebServiceFriendlyName.props XML file located in the Web application servlet directory for the WSA instance where the SOAP Web service is deployed.
To view and set SOAP Web service properties:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the WebServices Adapter broker instance where the SOAP Web service whose properties you want to view and set is deployed.

3. Click the WebServices Adapter broker instance. The **WebServices Adapter** details page appears.

4. Under **Deployed Web Services**, click **Application Properties**.

   For details on the properties that can appear in this list, see *OpenEdge Application Server: Administration*.

   **Note:** The actual list of properties displayed is a subset of all the properties, and changes depending on whether the session model of the SOAP Web service is session managed or session free. To verify the session model of a SOAP Web service, check its status. For more information, see the “Viewing the status of a SOAP Web service” section on page 217.

5. Click **Edit**.

6. Type the new value in the available field. For logical values, entering any value other than true results in a setting of false.

7. Click **Submit** when you finish. The changes are then written to the `WebServiceFriendlyName.props` file for the selected SOAP Web service.
Undeploying a SOAP Web service

You can undeploy a deployed SOAP Web service, which removes its deployment information, its properties, and other settings from the WSA instance context.

You must disable a SOAP Web service before you can undeploy it. You can verify if the SOAP Web service is disabled by checking its status.

To undeploy a SOAP Web service:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the WebServices Adapter broker instance where the SOAP Web service you want to undeploy is deployed.

3. Click the WebServices Adapter broker instance. The WebServices Adapter details page appears.

4. Under Deployed Web Services, click Undeploy.

If the SOAP Web service is already disabled, it is now undeployed from the WSA instance context and removed from the management console. If the SOAP Web service is still enabled, an error appears indicating that you must disable the SOAP Web service before undeploying it.

The changes are then written to the WebServiceFriendlyName.props file for the selected SOAP Web service.
You can configure property settings for WebSpeed Messengers and examine their log files in OpenEdge Management and OpenEdge Explorer.

You can configure WebSpeed Transaction Server property settings and add new WebSpeed instances in OpenEdge Management and OpenEdge Explorer. You can also start, stop, and delete WebSpeed instances, as well as view their status and log files.

This chapter provides information about WebSpeed Messenger and WebSpeed Transaction Server configuration, as described in the following topics:

- WebSpeed Messenger administration
- WebSpeed Transaction Server administration
WebSpeed Messenger administration

The WebSpeed Messenger resides on your Web server machine. It picks up incoming application service requests from WebSpeed clients and directs them to a WebSpeed broker that supports that application service. The Messenger is either a CGI program, or an ISAPI or NSAPI process.

There are four different WebSpeed Messengers:

- **CGIIP Messenger** — Runs on almost all Web servers, but tends to have the slowest response times.
- **WSASP Messenger** — Is used to call WebSpeed applications from a Microsoft Active Server Page. It cannot coexist with any other Messenger on your Web server.
- **WSISA Messenger** — Runs on Microsoft IIS Web servers.
- **WSNSA Messenger** — Runs on Netscape Web servers.

You cannot create or delete WebSpeed Messengers from the management console. You can, however, edit the Messenger's properties, enable or disable the Messenger, and examine the Messenger's log file from the console.

**Configuring a WebSpeed Messenger**

You can view or edit properties in the following categories for a WebSpeed Messenger in the management console:

- General
- Controlling NameServer
- Logging Setting
- SSL
- Advanced

You cannot create or delete WebSpeed Messengers from the management console. You can, however, edit the Messenger's properties, enable or disable the Messenger, and examine the Messenger's log file from the console.

Procedures for viewing or editing the configuration are described in the sections that follow.
WebSpeed Messenger administration

Viewing or modifying WebSpeed Messenger properties

You can view or modify General, Controlling NameServer, Logging Setting, SSL and Advanced properties for each WebSpeed Messenger.

To view or modify WebSpeed Messenger properties:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the WebSpeed Messenger whose properties you want to view or modify.

3. Click the WebSpeed Messenger. The Messenger details page appears.

4. Under Command and control, click Configuration.

From this page, you can do the following:

- View a read-only display of the Messenger properties as described in Table 57 through Table 62. All of these tables follow this procedure.

- Click Edit to modify the broker properties. For details, see the "Editing a WebSpeed Messenger configuration" section on page 229.

<table>
<thead>
<tr>
<th>Table 57: WebSpeed Messenger General properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Property</td>
</tr>
<tr>
<td>----------</td>
</tr>
<tr>
<td>Working directory</td>
</tr>
<tr>
<td>Messenger executable filename</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 58: WebSpeed Messenger Controlling NameServer properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1 of 2) Property</td>
</tr>
<tr>
<td>--------------------</td>
</tr>
<tr>
<td>Register NameServer</td>
</tr>
<tr>
<td>Register with NameServer</td>
</tr>
<tr>
<td>NameServer Information</td>
</tr>
<tr>
<td>Controlling NameServer</td>
</tr>
</tbody>
</table>
Minimum NameServer client port and Maximum NameServer client port

These properties are used to specify a range of UDP ports for the messenger instance. From this range, a messenger instance can select which UDP port to use when communicating with the NameServer. To limit the range, network administrators can specify a Minimum and a Maximum value.

Specifying a range using Minimum and Maximum allows network administrators to limit the number of UDP ports needed to be open in the firewall when the NameServer client (in this case, the messenger) and the NameServer are separated by a firewall. The value for Minimum and Maximum can be zero (0) or a number between 1024 and 65535 inclusive.

The Minimum value must be less than or equal to the Maximum value. If both the Minimum and Maximum values are set to zero (0), then a random port number within the implicit fixed range is dynamically assigned. How the port is assigned depends on the platform on which the messenger instance is running. If both Minimum and Maximum values are set to the same valid port number, then this UDP port number is used when communicating with the NameServer.

The default value for both Minimum and Maximum is 0.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum NameServer client port and Maximum NameServer client port</td>
<td>These properties are used to specify a range of UDP ports for the messenger instance. From this range, a messenger instance can select which UDP port to use when communicating with the NameServer. To limit the range, network administrators can specify a Minimum and a Maximum value. Specifying a range using Minimum and Maximum allows network administrators to limit the number of UDP ports needed to be open in the firewall when the NameServer client (in this case, the messenger) and the NameServer are separated by a firewall. The value for Minimum and Maximum can be zero (0) or a number between 1024 and 65535 inclusive. The Minimum value must be less than or equal to the Maximum value. If both the Minimum and Maximum values are set to zero (0), then a random port number within the implicit fixed range is dynamically assigned. How the port is assigned depends on the platform on which the messenger instance is running. If both Minimum and Maximum values are set to the same valid port number, then this UDP port number is used when communicating with the NameServer. The default value for both Minimum and Maximum is 0.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>WebSpeed Transaction Server Location</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>WebSpeed transaction server host</td>
<td>The WebSpeed Transaction Server host location.</td>
</tr>
<tr>
<td>WebSpeed transaction server port</td>
<td>The WebSpeed Transaction Server port location.</td>
</tr>
</tbody>
</table>
Logging setting properties

For complete information about these logging settings, see *OpenEdge Development: Debugging and Troubleshooting*.

Table 59: WebSpeed Messenger Logging Setting properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log filename</td>
<td>Location and name of the log file.</td>
</tr>
<tr>
<td>Logging level</td>
<td>A value that specifies the amount of information to be written to the Messenger log. Select from the following values in the drop-down list:</td>
</tr>
<tr>
<td></td>
<td>• <strong>None</strong> — Log no entries. This is equivalent to turning logging off.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Error Only</strong> — Log OpenEdge error messages. This includes all error messages and is unrelated to the entry types specified. Errors continue to be logged at all higher levels.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Basic</strong> — Logging entry types determine the logged information. Each entry type generates at least some output. This is the default.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Verbose</strong> — Logging entry types determine the logged information, but it is typically more information than <strong>Basic</strong>.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Extended</strong> — Logging entry types determine the logged information, but it is typically more information than <strong>Verbose</strong>.</td>
</tr>
<tr>
<td>Logging entry types</td>
<td>The one valid entry type for WebSpeed Messenger logging is MsgrTrace, which turns on logging for WebSpeed Messengers. The information logged depends on which Messenger is running and the logging level specified. For example:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Basic</strong> — For CGIIP, logs the start and end of the request, and the values of the CONTENT_LENGTH, PATH_INFO and QUERY_STRING variables. For other Messengers, logs the environment variables.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Verbose</strong> — Same as <strong>Basic</strong> but also includes the CGIIP environment variables and the environment variables in binary format for other Messengers.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Extended</strong> — Same as <strong>Verbose</strong>, but also includes the environment variables in binary format for CGIIP.</td>
</tr>
</tbody>
</table>

You can specify a logging level on the entry type. For example: **MsgrTrace:3**
Log file threshold size

A value of 0 or a value between 500,000 and 2,147,483,647, where 0 means there is no limit on the log file size other than what the operating system imposes.

Maximum number of log files

The number of rolled-over log files to keep. The value can be 0 or a value between 2 and 999999, where 0 means there is no limit on the number of server log files to keep. The specified number represents the maximum total number of log files to keep on disk at any time, including the current log file being written to.

When the file becomes equal to or greater than the log file threshold size, the client process renames it and creates a new log file. The file is renamed as follows, where ###### is a number starting at 000001 and increasing to 999999, after which it rolls back over to 000001:

filename.#####.extension

Table 60: WebSpeed Messenger SSL properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable SSL client connections</td>
<td>If the check box is selected, this property directs that the connection to the WebSpeed Transaction Server use SSL tunneling.</td>
</tr>
<tr>
<td>Disable client verification of SSL host name</td>
<td>If the check box is selected, this property turns off host verification for an SSL connection. If cleared, the client compares the host name for the WebSpeed Transaction Server specified by Host Name (or as returned by any Controlling NameServer) with the Common Name specified in the server digital certificate, and raises an error if they do not match. With this parameter specified, the client never raises the error.</td>
</tr>
<tr>
<td>Disable SSL session reuse</td>
<td>If the check box is selected, the connection does not reuse the SSL session ID when reconnecting to the same WebSpeed Transaction Server.</td>
</tr>
<tr>
<td>Disable SSL session caching</td>
<td>Disables session caching. (Session caching allows a client to reuse a previously established session if it reconnects prior to the session cache timeout expiring.)</td>
</tr>
<tr>
<td>SSL session cache timeout</td>
<td>Specifies in seconds the length of time an SSL session will be held in the session cache. The default is 180 seconds.</td>
</tr>
</tbody>
</table>
Table 61: WebSpeed Messenger Advanced properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal administration command</td>
<td>If the check box is selected, the Messenger allows access to the WSMAdmin Web page. The WSMAdmin Web page provides links to customize error messages, enable or disable session logging, and validate the traffic flow between the Messenger and the WebSpeed broker/agents.</td>
</tr>
<tr>
<td>Session connection ID</td>
<td>If the check box is selected, WebSpeed Session Connection IDs are enabled.</td>
</tr>
<tr>
<td>Authorized IP list</td>
<td>The list of all IP addresses that are allowed to access the WSMAdmin Web page, if it has been enabled. If you add any addresses to this list, then only requests coming from addresses that are on the list will reach the WSMAdmin Web page. This gives you the ability to tailor access to the WSMAdmin Web page, so that only users connecting from certain sites can access the administration functions available on that page.</td>
</tr>
</tbody>
</table>

Editing a WebSpeed Messenger configuration

You can edit the configuration of a WebSpeed Messenger.

To edit a WebSpeed Messenger configuration:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the WebSpeed Messenger whose configuration you want to edit.
3. From the list, click the WebSpeed Messenger. The Messenger details page appears.
4. Under Command and control, click Configuration. The messenger’s Configuration page appears.
5. Click Edit, and make the changes to the properties. (For details about the properties, see Table 57, Table 58, Table 59, Table 60, and Table 61.)
6. Click Save.
Viewing the WebSpeed Messenger log file

You can view the log file for a WebSpeed Messenger by using the log file viewer. The log file viewer allows you to examine the log file through an HTML interface.

To access the log file viewer:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the WebSpeed Messenger whose log file viewer you want to access.
3. From the list, click the WebSpeed Messenger. The Messenger details page appears.
4. Under Command and control section, click Log File Viewer.

You can use the Log File Viewer in the following ways:

- Use the Show field to control how many database log file entries display at one time. The number entered into the Show field cannot 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 cannot be more 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.

- Click Reload after changing the values in either the Show field or the Overlap field. If you do not reload, the viewer continues to display the previous values.
- Click Go To to control which numbered entry in the log file the viewer begins its display with. For example, a value of 10 entered into the Go To field will begin the display at the tenth log file entry.

Note: You must click Go To after entering a value in the Go To field or the viewer will not update its display.

The default display of entries is in ascending order; choose Descending to change the display. Note that the Show field dictates the number of entries shown, regardless of whether they display in ascending or descending order.

- Click First to display the first $x$ entries, where $x$ is the value in the Show field.
- Click Previous 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 last $x$ entries, where $x$ is the value in the Show field.
To view additional log file entries without changing your current starting log file entry, leave the Go To field blank, change the value in the Show field, and click Reload.

If the contents of the log file have changed since you opened the viewer, the log file viewer indicates this in the Log file status field.

OpenEdge Management and OpenEdge Explorer consider a viewer that has been inactive for more than four hours stale and at that point release ninety-five percent of any memory being held. If you try to use a stale viewer, OpenEdge Management and OpenEdge Explorer automatically reload the file. Because additional resource activity might have occurred during the viewer's inactivity, the reloaded log file view might not match the previous log file view of that resource.

OpenEdge Management and OpenEdge Explorer consider a viewer that has been inactive for forty-eight hours dead. Once a viewer dies, OpenEdge Management and OpenEdge Explorer release all of its memory. To return to the log file displayed in a dead view, you must renavigate to it, even if you pinned up the view or saved a link to it before the viewer died.
WebSpeed Transaction Server administration

With the WebSpeed Transaction Server, you can make WebSpeed applications available over the Web, an Intranet, or an Extranet.

You can use OpenEdge Management or OpenEdge Explorer to configure new WebSpeed Transaction Server instances and administer the configurations of existing instances. The WebSpeed Transaction Server installation provides one sample WebSpeed Transaction Server (wsbroker1). You can use this sample component as a template for creating and configuring additional WebSpeed Transaction Servers. Each WebSpeed Transaction Server is called an instance.

Working with the WebSpeed Transaction Server

Each WebSpeed Transaction Server consists of a WebSpeed broker and one or more WebSpeed agents. A WebSpeed agent is a single ABL Virtual Machine (AVM) instance running on a WebSpeed Transaction Server. Each WebSpeed agent executes ABL procedures on behalf of WebSpeed clients.

To work with a WebSpeed Transaction Server instance, double-click the WebSpeed folder from the management console's list frame. There is one pre-defined default WebSpeed Transaction Server (wsbroker1).

The following sections describe these WebSpeed Transaction Server actions:

• Configuring a WebSpeed Transaction Server
• Editing a WebSpeed Transaction Server configuration
• Creating a WebSpeed broker
• Starting and stopping a WebSpeed broker
• Adding or trimming WebSpeed agents
• Deleting a WebSpeed broker

Configuring a WebSpeed Transaction Server

You can view or edit the following configuration properties of a WebSpeed Transaction Server broker from the management console:

• Broker
• Agent
• SSL
• Messaging
• Environment variables

You can also create a new WebSpeed Transaction Server broker, delete a broker you no longer need, or view the broker's log files.

Each of these tasks is described in the sections that follow.
Viewing or modifying WebSpeed Transaction Server broker properties

You can view or modify the following WebSpeed Transaction Server Broker properties: general properties, owner information, controlling NameServer information, AppService Name List information, logging setting details, and advanced features.

To view or modify WebSpeed Transaction Server broker properties:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the WebSpeed Transaction Server broker whose properties you want to view or modify.
3. Click the WebSpeed Transaction Server broker. The **WebSpeed** details page appears.
4. In the **Command and control** section of the page, click **Configuration**. The **WebSpeed Configuration** page opens with that Broker’s properties in focus.

From this page, you can:

- View a read-only display of the WebSpeed broker configuration properties as described in Table 62 through Table 67. All these tables follow this procedure.
- Click **Edit** to modify the broker properties. For details, see the “Editing a WebSpeed Transaction Server configuration” section on page 256.

**Table 62: WebSpeed Broker General properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto start</td>
<td>Indicates whether or not the server automatically starts when the controlling AdminServer starts.</td>
</tr>
<tr>
<td>Port number</td>
<td>The number of the TCP/IP port that the server broker listens on.</td>
</tr>
<tr>
<td>Working directory</td>
<td>Your working directory. The default is the working directory set during OpenEdge installation.</td>
</tr>
<tr>
<td>Operating Mode</td>
<td>The operating mode for this server.</td>
</tr>
</tbody>
</table>

**Table 63: WebSpeed Broker Owner Information properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group name</td>
<td>(UNIX only) The group name on UNIX.</td>
</tr>
<tr>
<td>Username</td>
<td>The username of an account that has system-administrative rights.</td>
</tr>
</tbody>
</table>
Table 63: WebSpeed Broker Owner Information properties (2 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Password</td>
<td>The username account password.</td>
</tr>
<tr>
<td>Password Confirm</td>
<td>Confirmation of the username account password. This field appears only if you are editing the configuration. Note that this field does not appear if the AdminServer is running on a non-Windows host machine.</td>
</tr>
</tbody>
</table>

Table 64: WebSpeed Broker Controlling NameServer properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Register with NameServer</td>
<td>Indicates whether or not to register the broker with a controlling NameServer.</td>
</tr>
<tr>
<td>Controlling NameServer</td>
<td>The name of the NameServer with which this broker registers. (If you did not select to register the broker with a NameServer, the field is unavailable.)</td>
</tr>
<tr>
<td>Registration mode</td>
<td>How the broker specifies its hostname if it is registering with a controlling NameServer. This hostname information is passed onto a client application when it attempts to connect to an application service that the broker supports. The choices are:</td>
</tr>
<tr>
<td></td>
<td>• Register-IP — (Default) This setting is the most efficient mechanism, and can be used in most cases. It registers with the IP address of the machine where the broker is located.</td>
</tr>
<tr>
<td></td>
<td>• Register-LocalHost — The broker registers with the hostname of the machine that it runs on. Use this setting when the broker runs on a machine with a single hostname and more than one IP address.</td>
</tr>
<tr>
<td></td>
<td>• Register-HostName — The broker registers with the values specified in the HostName property. Use this setting when your clients need a fully qualified host name to connect to a broker in a different DNS domain.</td>
</tr>
<tr>
<td>Registration host name</td>
<td>The name of the non-local host.</td>
</tr>
</tbody>
</table>
Table 65: WebSpeed Broker AppService Name List properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application service names</td>
<td>A list to which you can add or delete application service names.</td>
</tr>
<tr>
<td>Supports default service</td>
<td>Indicates whether the AppServer broker supports the default service.</td>
</tr>
</tbody>
</table>

Table 66: WebSpeed Broker Logging Setting properties (1 of 4)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Broker log filename</td>
<td>The broker log filename.</td>
</tr>
<tr>
<td>Broker logging level (dynamic property)</td>
<td>A value that specifies the amount of information that is written to the broker log. The following values are available:</td>
</tr>
<tr>
<td></td>
<td>• <em>None</em> — Log no entries. This is equivalent to turning logging off.</td>
</tr>
<tr>
<td></td>
<td>• <em>Error Only</em> — Log OpenEdge error messages. This includes all error messages and is unrelated to the entry types specified. Errors continue to be logged at all higher levels.</td>
</tr>
<tr>
<td></td>
<td>• <em>Basic</em> — Broker logging entry types determine the logged information. Each entry type generates at least some output. This is the default.</td>
</tr>
<tr>
<td></td>
<td>• <em>Verbose</em> — Logging entry types determine the logged information, but it is typically more information than <em>Basic</em>.</td>
</tr>
<tr>
<td></td>
<td>• <em>Extended</em> — Logging entry types determine the logged information, but it is typically more information than <em>Verbose</em>.</td>
</tr>
<tr>
<td></td>
<td>You can update this property dynamically. Any changes will affect all current and new brokers and/or agents.</td>
</tr>
<tr>
<td>Append to broker log file</td>
<td>Indicates whether or not a new broker log file should be created when the broker is started.</td>
</tr>
</tbody>
</table>
### Broker logging entry types

A comma-separated list of the following valid entry types for agent logging:

- **UBroker.AutoTrim** — Logs messages about automatic agent thread trimming by the broker based on the Auto trim timeout property. Information is logged at **Basic** and higher.

- **UBroker.Basic** — Logs messages about broker and agent startup and shutdown at the following logging levels:
  - **Basic** — Logs NameServer registration and connections from clients.
  - **Verbose** — Logs information about all the property values read from the ubroker.properties file for that broker instance, and more information on the connection from clients.

- **UBroker.ThreadPool** — Logs messages about the pool of threads managed by the broker. These threads are used to control the client requests and the agent processes. This entry type logs messages when adding, removing, and communicating with these threads. Information is logged at **Basic** and higher.

You can update this property dynamically. Any changes will affect all current and new brokers and/or agents.

Turn on the following logging entry types only at the request of Progress Software Corporation Technical Support:

- **UBroker.ClientFSM** — Logs messages about the state of the client thread as it processes the requests, from the moment it receives a connection request to the end of the request. Information is logged at **Basic** and higher.

- **UBroker.ServerFSM** — Logs messages about the state of the agent thread as it processes the requests, from the moment it receives a connection request to the end of the request. Information is logged at **Basic** and higher.

- **UBroker.ClientMsgStream** — Logs messages about messages exchanged between the broker and client threads in binary format. Information is logged at **Basic** and higher.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Broker logging entry types</td>
<td>A comma-separated list of the following valid entry types for agent logging: <strong>UBroker.AutoTrim</strong> logs messages about automatic agent thread trimming by the broker based on the Auto trim timeout property. Information is logged at <strong>Basic</strong> and higher. <strong>UBroker.Basic</strong> logs messages about broker and agent startup and shutdown at the following logging levels: <strong>Basic</strong> logs NameServer registration and connections from clients. <strong>Verbose</strong> logs information about all the property values read from the ubroker.properties file for that broker instance, and more information on the connection from clients. <strong>UBroker.ThreadPool</strong> logs messages about the pool of threads managed by the broker. These threads are used to control the client requests and the agent processes. This entry type logs messages when adding, removing, and communicating with these threads. Information is logged at <strong>Basic</strong> and higher. You can update this property dynamically. Any changes will affect all current and new brokers and/or agents. Turn on the following logging entry types only at the request of Progress Software Corporation Technical Support: <strong>UBroker.ClientFSM</strong> logs messages about the state of the client thread as it processes the requests, from the moment it receives a connection request to the end of the request. Information is logged at <strong>Basic</strong> and higher. <strong>UBroker.ServerFSM</strong> logs messages about the state of the agent thread as it processes the requests, from the moment it receives a connection request to the end of the request. Information is logged at <strong>Basic</strong> and higher. <strong>UBroker.ClientMsgStream</strong> logs messages about messages exchanged between the broker and client threads in binary format. Information is logged at <strong>Basic</strong> and higher.</td>
</tr>
</tbody>
</table>

---

Table 66: WebSpeed Broker Logging Setting properties (2 of 4)
Broker logging entry types
(continued)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UBroker.ServerMsgStream</td>
<td>Logs messages about messages exchanged between the broker and agent threads in binary format. Information is logged at Basic and higher.</td>
</tr>
<tr>
<td>UBroker.ClientMsgQueue</td>
<td>Logs messages about the client thread as it processes queued messages. Information is logged at Basic and higher.</td>
</tr>
<tr>
<td>UBroker.ServerMsgQueue</td>
<td>Logs messages about the agent thread as it processes queued message. Information is logged at Basic and higher.</td>
</tr>
<tr>
<td>UBroker.ClientMemTrace</td>
<td>Traces messages being processed by the client threads. Information is logged at Basic and higher.</td>
</tr>
<tr>
<td>UBroker.ServerMemTrace</td>
<td>Traces messages being processed by the agent threads. Information is logged at Basic and higher.</td>
</tr>
<tr>
<td>UBroker.Stats</td>
<td>Logs statistics about the number of requests, plus maximum, minimum, and average request wait and duration times. Information is logged at Basic and higher.</td>
</tr>
<tr>
<td>UBroker.All</td>
<td>Logs all UBroker entry types.</td>
</tr>
</tbody>
</table>

You can specify a separate logging level for each log entry type and use a wildcard (*) to specify multiple entry types. For example:

```
UBroker.Basic:3,UBroker.ThreadPool:4,UBroker.Client*
```
<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Broker log file threshold size</td>
<td>A value of 0 or a value between 500,000 and 2,147,483,647, where 0 means there is no limit on the log file size other than what the operating system imposes.</td>
</tr>
<tr>
<td>Maximum number of broker log files</td>
<td>The number of rolled-over log files to keep. The value can be 0 or a value between 2 and 999999, where 0 means there is no limit on the number of broker log files to keep. The specified number represents the maximum total number of log files to keep on disk at any time, including the current log file being written to. When the file becomes equal to or greater than the Broker log file threshold size, the client process renames it and creates a new log file. The file is renamed as follows, where ###### is a number starting at 000001 and increasing to 999999, after which it rolls back over to 000001: filename.#####.extension</td>
</tr>
</tbody>
</table>

Table 66: WebSpeed Broker Logging Setting properties (4 of 4)
Table 67: WebSpeed Broker Advanced Features properties  

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum client instances</td>
<td>The maximum number of client connections that the broker can support concurrently. The default is 512, which is high enough to ensure that the number of client connections is virtually unlimited. Realistically, however, the system-level resources required to support an unlimited number of clients might be exhausted before this limit is reached. If you see broker log entries that contain Exception Messages for &quot;out of space&quot; or &quot;OutOfMemory&quot; you might need to reduce the maximum number of client connections to a more reasonable value. Note, however, that lowering this value can cause some client requests to be rejected (&quot;Exceeded Max Clients&quot;). Configuring and starting multiple brokers to handle higher client loads will alleviate the problem.</td>
</tr>
<tr>
<td>Priority weight (0-100)</td>
<td>An integer value between 0 and 100 that influences the share of the workload that the selected AppServer receives. The larger the value, the heavier the load that is distributed to the server. The NameServer distributes client requests across all WebSpeed Transaction Servers that have the same application service in proportion to the Priority weight value. The default is 0. You can update this property dynamically. Any changes will affect all current and new brokers.</td>
</tr>
<tr>
<td>Registration retry (in seconds)</td>
<td>After registering with its controlling NameServer, the WebSpeed Transaction Server periodically sends &quot;keep-alive&quot; messages to the NameServer to let the NameServer know that it is still active. The Registration retry value is the number of seconds that pass between &quot;keep-alive&quot; messages. The default is 30.</td>
</tr>
<tr>
<td>Server startup timeout</td>
<td>Specifies the amount of time, in seconds, that the broker waits for an active agent to become available before starting a new agent process. The default is 3. You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated.</td>
</tr>
</tbody>
</table>
### Table 67: WebSpeed Broker Advanced Features properties (2 of 4)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Request timeout**       | Specifies the amount of time, in seconds, that the broker waits for an agent to become available for processing a request.  
                          | The broker waits for this period only if the **Maximum servers** setting has been reached.  
                          | After the timeout has expired, the client receives a "no servers available" error message.  
                          | The default is 15.  
                          | You can update this property dynamically. Any changes will affect all current and new brokers and/or agents. |
| **Auto trim timeout**     | The amount of time, in seconds, that the broker waits before automatically trimming the number of running agents.  
                          | The broker keeps track of the maximum number of agents that are simultaneously busy during the interval you specify. At the end of the interval, the broker attempts to trim the number of agents to match either the **Maximum servers** for the interval, or the **Minimum servers**, whichever is greater.  
                          | You can disable this feature by setting the timeout to zero (0). The default **Server Auto-trim interval** is 1800 seconds.  
                          | Progress Software recommends that you use this default setting to avoid unnecessary process management that might result from frequent starting and trimming of agents.  
                          | You can update this property dynamically. Any changes will affect all current and new brokers and/or agents. |
| **TCP/IP version**        | Internet Protocol for network communication:  
                          |  • **IPv4** — Accepts only IPv4 connections  
                          |  • **IPv6** — Allows IPv4 and IPv6 connections using mapped address (where supported)  
                          | The default is IPv4.  
                          | You must also set the **jvmArgs** property in the ubroker.properties file to override the default behavior of the JVM for IPv6 connections. |
| **jvmargs**               | Java System Properties.                                                                                                                          |

`jvmArgs=-Djava.net.preferIPv4Stack=false -Djava.net.preferIPv6Addresses=true`
### Table 67: WebSpeed Broker Advanced Features properties (3 of 4)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Enable dynamic property updates</strong></td>
<td>Allows dynamic changes to occur to several of the instance’s properties, without requiring you to shut down the instance.</td>
</tr>
<tr>
<td></td>
<td>This property is disabled by default.</td>
</tr>
<tr>
<td><strong>Enable debugging through the broker</strong></td>
<td>Specifies whether or not the debugger is enabled; select the check box to enable it.</td>
</tr>
<tr>
<td>(dynamic property)</td>
<td>If you disable this property while the debugger is connected, the debugger connection terminates.</td>
</tr>
<tr>
<td><strong>Broker debugger port number</strong></td>
<td>Specifies the port number that the broker debugger will use when it starts. You can choose any valid value; if the port you specify is already in use, the debugger service will fail to start.</td>
</tr>
<tr>
<td>(dynamic property)</td>
<td>If you change the port number while the debugger service is active, the debugger service will be shut down and restarted using the new port number.</td>
</tr>
<tr>
<td></td>
<td>The default port number is 3099.</td>
</tr>
<tr>
<td><strong>Enable SSL for broker debugger</strong></td>
<td>Specifies whether the broker debugger should accept only SSL/TLS-encrypted connections. If you enable this property, the Progress Developer Studio debugger will be required to establish an SSL/TLS session.</td>
</tr>
<tr>
<td>(dynamic property)</td>
<td>If you enable or disable the property while the debugger service is active, the debugger will be disconnected. The debugger service will be shut down and then restarted using the new selection.</td>
</tr>
<tr>
<td></td>
<td>This option is disabled by default.</td>
</tr>
<tr>
<td><strong>Use the broker’s private key/digital certificate for debugger</strong></td>
<td>When enabled, indicates that, when using SSL, the broker debugger service should use the same private key as used by the AppServer agents.</td>
</tr>
<tr>
<td>(dynamic property)</td>
<td>If this property is enabled, then the Debugger private key/digital certificate alias name and Password to access the debugger's key/certificate property settings are ignored.</td>
</tr>
<tr>
<td></td>
<td>This property is enabled by default.</td>
</tr>
</tbody>
</table>
Table 67: WebSpeed Broker Advanced Features properties  

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Debugger private key/digital certificate alias name (dynamic property)</td>
<td>The name of the private key that should be used for the debugger service when the agent's private key is not being used. This property must be defined if <strong>Enable SSL for broker debugger</strong> is enabled.</td>
</tr>
<tr>
<td>Password to access the debugger's key/certificate (dynamic property)</td>
<td>The password required to access the private key specified by <strong>Debugger private key/digital certificate alias name</strong>. The default for this property is blank.</td>
</tr>
<tr>
<td>Broker debugger passphrase (dynamic property)</td>
<td>An optional property that represents the passphrase required by the debugger to connect to the debugger service in the broker. If the debugger is connected to the debugger service when the passphrase is changed, the debugger remains connected. The next time the debugger attempts to connect to the debugger service, the new password is required. The default for this property is blank.</td>
</tr>
<tr>
<td>Broker debugger passphrase Confirm</td>
<td>Confirms the passphrase provided in <strong>Broker debugger passphrase</strong>.</td>
</tr>
<tr>
<td>Publish directory</td>
<td>The name of the local directory where you want r-code to be placed when you are publishing from Progress Developer Studio. If you are publishing remotely from Progress Developer Studio, you must set this property.</td>
</tr>
</tbody>
</table>
Viewing or modifying WebSpeed Agent properties

You can set the following agent properties for a WebSpeed Broker: general properties, logging setting details, pool range, and advanced features.

To view or modify WebSpeed Agent properties:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the WebSpeed Transaction Server broker whose agent properties you want to view or modify.
3. Click the WebSpeed Transaction Server broker. The WebSpeed details page appears.
4. In the Command and control section of the page, click Configuration. The WebSpeed Configuration page opens with the Broker properties in focus.
5. Click the Agent tab.

From this page, you can do the following:

- View a read-only display of the WebSpeed Agent properties as described in Table 68 through Table 71. All of these tables follow this procedure.
- Click Edit to modify the broker properties. For details, see the “Editing a WebSpeed Transaction Server configuration” section on page 256.

### Table 68: WebSpeed Agent General properties (1 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server executable file</td>
<td>Either the default agent executable pathname or a different agent executable file of your choosing. You generally only need to specify a nondefault value if you have generated a new executable using the OEBUILD utility.</td>
</tr>
<tr>
<td>(dynamic property)</td>
<td>You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated.</td>
</tr>
</tbody>
</table>

| Server startup parameters          | The OpenEdge startup parameters that you want to specify to start each agent.                                                                 |
| (dynamic property)                 | You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated. |
### Table 68: WebSpeed Agent General properties (2 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROPATH</td>
<td>The search path that agents use to locate ABL procedures that they execute. Specify a list of directory pathnames in the form of an OpenEdge PROPATH. Make sure that you have copied all of your WebSpeed Transaction Server procedures to one or more of the directories in the PROPATH.</td>
</tr>
<tr>
<td>Minimum port number (dynamic property)</td>
<td>The minimum TCP/IP port number from a range that each agent process can listen on. When each agent starts, it allocates a port that is not being used from the specified range, up to the <strong>Maximum port number</strong> setting. You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated.</td>
</tr>
<tr>
<td>Maximum port number (dynamic property)</td>
<td>The maximum TCP/IP port number from a range that each agent can listen on. When each agent starts, it allocates a port that is not being used from the specified range, down to the <strong>Minimum port number</strong> setting. You can update this property dynamically. Any changes will affect only new agents that start after the property value has changed; existing (running) brokers and agents are not updated.</td>
</tr>
<tr>
<td>Server application mode</td>
<td>Development or Production.</td>
</tr>
<tr>
<td>Flush statistical data (dynamic property)</td>
<td>How often the data is flushed by the agent to the broker. This is expressed as the number of remote procedure calls between flushes. You can update this property dynamically. Any changes will affect all current and new brokers and/or agents.</td>
</tr>
</tbody>
</table>
Table 69: WebSpeed Agent Logging Setting properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server log filename</td>
<td>The server log file name.</td>
</tr>
<tr>
<td>Server logging level</td>
<td>A value that specifies the amount of information to be written to the agent log. You can select from the following values in the drop-down list:</td>
</tr>
<tr>
<td>(dynamic property)</td>
<td>- <strong>None</strong> — Log no entries. This is equivalent to turning logging off.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Error Only</strong> — Log OpenEdge error messages. This includes all error messages and is unrelated to the entry types specified. Errors continue to be logged at all higher levels.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Basic</strong> — Server logging entry types determine the logged information. Each entry type generates at least some output. This is the default.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Verbose</strong> — Logging entry types determine the logged information, but it is typically more information than <strong>Basic</strong>.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Extended</strong> — Server logging entry types determine the logged information, but it is typically more information than <strong>Verbose</strong>.</td>
</tr>
<tr>
<td></td>
<td>You can update this property dynamically. Any changes will affect all current and new brokers and/or agents.</td>
</tr>
<tr>
<td>Append to server log file</td>
<td>Indicates whether a new agent log file should be created when the WebSpeed Transaction Server is started.</td>
</tr>
</tbody>
</table>
Chapter 7: Configuring WebSpeed Messengers and WebSpeed Transaction Servers

Table 69: WebSpeed Agent Logging Setting properties (2 of 4)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Server logging entry types</strong>&lt;br&gt;(dynamic property)</td>
<td>A comma-separated list of valid entry types for agent logging:</td>
</tr>
<tr>
<td></td>
<td>• <strong>4GLTrace</strong> — Turns on the logging for the execution of the following ABL statements: RUN, FUNCTION (user-defined functions), PUBLISH and SUBSCRIBE. Logs information at <strong>Basic</strong> or higher.</td>
</tr>
<tr>
<td></td>
<td>• <strong>ASDefault</strong> — Combines the <strong>ASPlumbing</strong> and <strong>DB.Connects</strong> log entry types. This is the default value for AppServer agents.</td>
</tr>
<tr>
<td></td>
<td>• <strong>ASPlumbing</strong> — Turns on logging for different actions depending on the logging level specified:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Basic</strong> — Logs messages when an AppServer agent starts and stops, and when client processes connect and disconnect from the AppServer agent.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Verbose</strong> — Same as <strong>Basic</strong>, plus logging of the execution of the top level procedure (that is, the procedure that the client process asks the AppServer agent to execute on its behalf, but not any procedure that the top level procedure executes).</td>
</tr>
<tr>
<td></td>
<td>• <strong>Extended</strong> — Same as <strong>Verbose</strong>, plus messages about agent state, and status messages about read and write socket operations. Do not turn this on unless instructed to do so by Progress Software Technical Support.</td>
</tr>
<tr>
<td></td>
<td>• <strong>DB.Connects</strong> — Turns on logging of database connections and disconnections. The messages include database name and user ID number. Logs information at <strong>Basic</strong> or higher.</td>
</tr>
<tr>
<td></td>
<td>• <strong>DynObjects.DB</strong> — Turns on logging for dynamic database-related objects (for example, TEMP-TABLE or DATASET) as they are created and destroyed. Logs information at <strong>Basic</strong> or <strong>Verbose</strong>.</td>
</tr>
<tr>
<td></td>
<td>• <strong>DynObjects.XML</strong> — Turns on logging for dynamic XML-related objects (for example, x-document or x-noderef) as they are created and destroyed. Logs information at <strong>Basic</strong> or <strong>Verbose</strong>.</td>
</tr>
</tbody>
</table>
### Server logging entry types (continued)

- **DynObjects.Other** — Turns on logging for dynamic objects that do not match the other dynamic object categories (for example, procedure or server socket) as they are created and destroyed. Logs information at **Basic** or **Verbose**.

- **ProEvents.UI.Char** — Turns on logging of keystroke events for printable characters. Logs information at **Basic** or higher.

- **ProEvents.UI.Command** — Turns on logging of keystroke events for nonprintable characters at the following logging levels:
  - **Basic** — Logs nonprintable keystrokes and a subset of Windows GUI events
  - **Verbose** — Logs nonprintable keystrokes and all Windows GUI events

- **ProEvents.Other** — Turns on logging of COM, asynchronous, and server socket events. Logs information at **Basic** or higher, beginning with events for which you have written triggers.

- **QryInfo** — Turns on database query information logging at the following logging levels, with the same amount of information logged for each level:
  - **Basic** — Logs static queries and initial dynamic queries
  - **Verbose** — Logs static queries and multiple instances of dynamic queries

- **SAX** — Turns on logging for various stages of execution using the ABL SAX parser.

You can update this property dynamically. Any changes will affect all current and new brokers and/or agents.

You can specify a separate logging level for each log entry type, and you can use a wildcard (*) to specify multiple entry types. For example:

```
4GLTrace:3,ASPlubing:2,DynObjects.*
```

### Server log file threshold size

A value of 0 or a value between 500,000 and 2,147,483,647, where 0 means there is no limit on the agent log file size other than what the operating system imposes.
Maximum number of server log files

The number of rolled-over log files to keep. The value can be 0 or a value between 2 and 999999, where 0 means there is no limit on the number of agent log files to keep. The specified number represents the maximum total number of log files to keep on disk at any time, including the current log file being written to.

When the file becomes equal to or greater than the Server log file threshold size, the client process renames it and creates a new log file. The file is renamed as follows, where ###### is a number starting at 000001 and increasing to 999999, after which it rolls back over to 000001:

```
filename.#####.extension
```

Server watch dog interval

The interval (in seconds) for the server log file watchdog thread. If the Server log file threshold size property is specified, the broker will roll over the server log file when its size is larger than the logThreshold value.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum number of server log files</td>
<td>The number of rolled-over log files to keep. The value can be 0 or a value between 2 and 999999, where 0 means there is no limit on the number of agent log files to keep. The specified number represents the maximum total number of log files to keep on disk at any time, including the current log file being written to. When the file becomes equal to or greater than the Server log file threshold size, the client process renames it and creates a new log file. The file is renamed as follows, where ###### is a number starting at 000001 and increasing to 999999, after which it rolls back over to 000001: <code>filename.#####.extension</code>.</td>
</tr>
<tr>
<td>Server watch dog interval</td>
<td>The interval (in seconds) for the server log file watchdog thread. If the Server log file threshold size property is specified, the broker will roll over the server log file when its size is larger than the logThreshold value.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial number of servers to start</td>
<td>The number of agents you want the broker to start initially.</td>
</tr>
<tr>
<td>Minimum servers</td>
<td>The minimum number of agents before the broker starts additional agents. If you trim the number of agents below this value, the server starts any additional agents needed to maintain the specified minimum when the next client connects.</td>
</tr>
<tr>
<td>Maximum servers</td>
<td>The maximum number of agents that this server can have running at the same time.</td>
</tr>
</tbody>
</table>
Table 71: **WebSpeed Agent Advanced Features properties**  

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application URL</td>
<td>A server-relative URL (starting with a &quot;&quot;) to the path of the application. If the Web server redirects an incoming URL such that the SCRIPT_NAME and PATH_INFO variables do not refer to the application, setting this property allows an application to create self-referencing URLs when SCRIPT_NAME and/or PATH_INFO are not correct. If this option is set, the value of the applicationURL global variable available to applications is set to this value instead of a combination of SCRIPT_NAME and PATH_INFO.</td>
</tr>
<tr>
<td>Default cookie path</td>
<td>A server-relative URL (starting with a &quot;&quot;) to use for all cookies if not explicitly specified in the set-cookie( ) or delete-cookie( ) functions. The default is the value of the AppURL global variable.</td>
</tr>
<tr>
<td>Default cookie domain</td>
<td>Domain to send cookies. The default is blank, causing the Web browser to set cookies only for the current host. If cookies must be available to multiple hosts in the same domain by default, set that domain name here; for instance, .progress.com. Unless a value is specified for the domain option for set-cookie( ) and delete-cookie( ) functions, this value is used.</td>
</tr>
<tr>
<td>File upload directory</td>
<td>Specifies a directory where WebSpeed servers can upload files. Refer to the ubroker.properties file for an HTML snippet that shows how to use this feature. By default this capability is disabled. To enable it, specify an upload directory. Only text file upload is supported, not binary.</td>
</tr>
</tbody>
</table>
| Binary upload max size    | Specifies the maximum size (in bytes) of binary files that WebSpeed handles. The application can access the contents of the binary files through the ABL language. Note that this property is not related to the File upload directory property; therefore, WebSpeed will not upload the file to a directory. You can set any of the following options:  
  - 0 — Binary upload is disabled.  
  - -1 — No maximum size of a binary upload; binary upload enabled.  
  You can alternatively type any non-zero value greater than 0 as the maximum size of the binary upload. |
Chapter 7: Configuring WebSpeed Messengers and WebSpeed Transaction Servers

### Viewing or modifying WebSpeed SSL properties

You can set properties in the following WebSpeed Broker SSL categories: general properties and advanced features.

#### To view or modify WebSpeed SSL properties:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the WebSpeed Transaction Server broker whose SSL properties you want to view or modify.

3. Click the WebSpeed Transaction Server broker. The **WebSpeed** details page appears.

4. In the **Command and control** section of the page, click **Configuration**. The **WebSpeed Configuration** page opens with the **Broker** properties in focus.

5. Click the **SSL** tab.

From this page, you can:

- View a read-only display of the WebSpeed SSL properties as described in the **Table 72** and **Table 73**. Both of these tables follow this procedure.

- Click **Edit** to modify the SSL properties. For details, see the "Editing a WebSpeed Transaction Server configuration" section on page 256.

---

### Table 71: WebSpeed Agent Advanced Features properties (2 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Debug mode</td>
<td>Whether the debug mode is enabled, disabled, or using the default.</td>
</tr>
</tbody>
</table>
| 4GL debugger enabled   | Whether the debugger is enabled or disabled:  
  - **Disabled** — Specifies ?debug=on in the URL has no effect  
  - **Enabled** — Allows debug information if ?debug=on is in the URL |
| Execution Time Limit   | Sets the maximum time in seconds that a remote procedure can execute on a given Web server.  
  The time limit applies to all remote procedures that execute on the Web server.  
  The default value is zero, which indicates that no time limit exists. |
### WebSpeed SSL General properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable SSL client connections</td>
<td>If the check box is selected, this property specifies that all connections to this WebSpeed Transaction Server must use SSL tunneling.</td>
</tr>
<tr>
<td>Private key/Digital certificate alias name</td>
<td>The alias name within the OpenEdge keystore of the private key and digital certificate entry to use to authenticate all connections to this WebSpeed Transaction Server. If not otherwise selected, the WebSpeed Transaction Server uses the default_server server certificate alias.</td>
</tr>
<tr>
<td>Password to access the key/certificate</td>
<td>The password to use for accessing the private key and digital certificate. You must specify a password when you specify the select an OpenEdge keystore alias name.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Your password is encrypted in the ubroker.properties file. If you set the password in ubroker.properties manually, you must specify the password as an encrypted value. You can obtain this value using the genpassword utility located in the bin directory of your OpenEdge installation.</td>
</tr>
<tr>
<td></td>
<td>If you use the default_server server certificate, it also has a default password that you do not need to specify.</td>
</tr>
<tr>
<td>Password to access the key/certificate Confirm</td>
<td>Confirmation of the username account password. This field appears only if you are editing the configuration.</td>
</tr>
<tr>
<td></td>
<td>Note that this field does not appear if the AdminServer is running on a non-Windows host machine.</td>
</tr>
</tbody>
</table>

### WebSpeed SSL Advanced Features properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disable SSL session caching</td>
<td>If this property is checked, caching for the SSL client session is disabled.</td>
</tr>
<tr>
<td>SSL session cache timeout</td>
<td>The length of time, in seconds, that an SSL client session is held in the session cache, during which an SSL client can resume its session. The default is 180.</td>
</tr>
</tbody>
</table>
Viewing or modifying WebSpeed Messaging properties

You can view or modify several WebSpeed Messaging properties.

To view or modify WebSpeed Messaging properties:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the WebSpeed Transaction Server broker whose messaging properties you want to view or modify.

3. Click the WebSpeed Transaction Server broker. The WebSpeed details page appears.

4. In the Command and control section of the page, click Configuration. The WebSpeed Configuration page opens with the Broker properties in focus.

5. Click the Messaging tab.

From this page, you can:

- View a read-only display of the WebSpeed Messaging properties as described in Table 74, which follows this procedure.

- Click Edit to modify the WebSpeed Messaging properties. For details, see the "Editing a WebSpeed Transaction Server configuration" section on page 256.

Table 74: WebSpeed Messaging properties (1 of 3)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SonicMQ ServerConnect enabled</td>
<td>Allows the asbroker or wsbroker agent to start a personal SonicMQ Adapter instead of going to a separate one running elsewhere.</td>
</tr>
<tr>
<td>SonicMQ ServerConnect broker log filename</td>
<td>A valid path and file name.</td>
</tr>
</tbody>
</table>
### WebSpeed Messaging properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| Broker logging level                  | A value that specifies the amount of information to be written to the SonicMQ ServerConnect broker log. You can select from the following values in the drop-down list:  
  - None — Log no entries. This is equivalent to turning logging off.  
  - Error Only — Log error messages. This includes all error messages and is unrelated to the entry types specified. Errors continue to be logged at all higher levels.  
  - Basic — Broker logging entry types determine the logged information. Each entry type generates at least some output. This is the default.  
  - Verbose — Broker logging entry types determine the logged information, but it is typically more information than Basic.  
  - Extended — Broker logging entry types determine the logged information, but it is typically more information than Verbose.  
  You can update this property dynamically. Any changes will affect all current and new brokers and/or agents. |
| Append to broker log file             | Indicates whether a new broker log file should be created when the SonicMQ ServerConnect broker is started, even if the broker log file already exists.                                                         |
| SonicMQ ServerConnect server log filename | A valid path and file name.                                                                                                                                                                                      |
Table 74:  WebSpeed Messaging properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| Server logging level            | A value that specifies the amount of information to be written to the SonicMQ ServerConnect server log. Each logging level name has the indicated numeric value. You can select from the following values in the drop-down list:  
  - **None** — Log no entries. This is equivalent to turning logging off.  
  - **Error Only** — Log error messages. This includes all error messages and is unrelated to the entry types specified. Errors continue to be logged at all higher levels.  
  - **Basic** — Broker logging entry types determine the logged information. Each entry type generates at least some output. This is the default.  
  - **Verbose** — Broker logging entry types determine the logged information, but it is typically more information than **Basic**.  
  - **Extended** — Broker logging entry types determine the logged information, but it is typically more information than **Verbose**.  
You can update this property dynamically. Any changes will affect all current and new brokers and/or agents. |
| Append to server log file       | Indicates whether a new server log file should be created, even if the broker log file already exists, when the SonicMQ ServerConnect instance is started. |
Setting or deleting WebSpeed environment variables

You can set or delete environment variables. For WebSpeed, environment variables that appear in the list are scoped to:

- The process in which the WebSpeed Transaction Server broker executes
- Every WebSpeed Transaction Server agent that the WebSpeed Transaction Server broker starts

To set or delete WebSpeed environment variables:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the WebSpeed Transaction Server broker whose environmental variables you want to set.

3. Click the WebSpeed Transaction Server broker. The WebSpeed details page appears.

4. In the Command and control section of the page, click Configuration. The broker’s Configuration page appears.

5. Click Edit.

6. Click Environment Variables.

7. Enter each variable name and then either provide its value in the format name=value, or remove an existing variable.

8. Click Save when you finish adding (or deleting) the variables and their values.
Chapter 7: Configuring WebSpeed Messengers and WebSpeed Transaction Servers

Edit a WebSpeed Transaction Server configuration

You can edit a WebSpeed Transaction Server configuration.

To edit a WebSpeed Transaction Server configuration:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the WebSpeed Transaction Server broker whose configuration you want to edit.
3. Click the WebSpeed Transaction Server broker. The WebSpeed details page appears.
4. In the Command and control section of the page, click Configuration. The WebSpeed Configuration page opens.
5. Click Edit. Make the changes you want to the properties. (For details about the properties, see Table 62 through Table 74.)
6. Click Save. Any changes you make are also saved in the ubroker.properties file.

Creating a WebSpeed broker

OpenEdge Management and OpenEdge Explorer provide a sample WebSpeed broker instance (wsbroker1) with a default configuration. Each new instance you create uses the same default configuration, which you can edit.

To create a WebSpeed broker instance:

1. From the drop-down for Resources on the management console menu, click New OpenEdge Resource→WebSpeed. The WebSpeed Configuration page appears.
2. Type the name of the new WebSpeed broker instance in the field provided.

Note: The WebSpeed broker instance name is case sensitive and can include any character except a period (.) or square brackets ([ ]). The name must be unique among all configured WebSpeed broker instance names.

3. Click Save. The WebSpeed Configuration page appears, allowing you to configure the WebSpeed broker’s properties.

Each new WebSpeed broker instance you create uses the default configuration. However, several properties, such as port numbers, must have unique values for each WebSpeed broker in order for the instance to operate properly.
Starting and stopping a WebSpeed broker

From the management console, you can start either a local WebSpeed broker or a WebSpeed broker running under a remote AdminServer on another machine. You must first connect to the AdminServer that manages the WebSpeed broker you want to start.

You can also choose to start a WebSpeed broker automatically when the AdminServer starts.

**Note:** Prior to starting a WebSpeed broker using SonicMQ ServerConnect, you must first configure the WebSpeed broker as SonicMQ ServerConnect enabled.

To start or stop a WebSpeed broker, refer to the **Starting or Stopping OpenEdge resources** section in *OpenEdge Management: Servers, DataServers, Messengers, and Adapters*.

Starting a WebSpeed broker automatically

You can set a WebSpeed broker to start automatically when the AdminServer starts.

**To start a WebSpeed broker automatically:**

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the WebSpeed Transaction Server broker you want to start automatically.
3. Click the Edit icon. The **WebSpeed** details page appears.
4. In the **Command and control** section of the page, click **Configuration**. The **WebSpeed Configuration** page appears.
5. Click **Edit**.
6. Select the **Auto start** option by clicking in the check box.
7. Click **Save**.

Adding or trimming WebSpeed agents

Using OpenEdge Management or OpenEdge Explorer, you can start additional WebSpeed agents, up to the number specified in the Maximum Agents property setting.

You can also trim running WebSpeed agents, down to the Minimum Agents property setting.
More information about trimming WebSpeed Agents

To better understand the use of the Agent Auto-trim feature, consider a scenario involving a broker serving a client/server application running in stateless mode. Assume that the following property values are set:

- Minimum Agent/Server Instances: 10
- Initial Agent/Server Instances to Start: 10
- Maximum Agent/Server Instances: 50
- Agent/Server Auto-trim Interval: 1800

When the broker starts, it starts 10 agents. This also marks the start of the first Agent/Server Auto-trim Interval. Assume that the broker is idle for 30 minutes (it is just before people arrive for work, for instance). At the end of the 30-minute interval, the highest number of agents that were busy at the same time would be zero. (Since there were no requests made of any of the agents, none of the agents was busy.) However, since zero is less than the Minimum Agent/Server Instances value of 10, no automatic trimming takes place. This marks the start of the second interval.

As the second interval starts, employees begin their workday. There is a sudden demand for agents; in fact, the broker receives so many requests that it must start 20 additional agents. Assume that the high volume of workload, and demand on the agents, continues through the second interval. At the interval's end, the maximum number of busy agents is 30, which is the number the broker scaled up to in response to high demand. Thirty agents are running, and the maximum busy level is 30. No trimming occurs.

During the next interval, the request load diminishes somewhat. At the end of the interval, only 20 agents are concurrently busy. The maximum number of busy agents for the interval remains at 30, since 30 were busy at the start of the interval. Thirty agents are running, and the maximum busy level is 30. No trimming occurs.

Assume that all employees have to attend a company-wide meeting that begins midway through the interval. The highest number of agents busy at that same time was 20. At the end of the interval, there are 30 agents running, with zero agents currently busy. Since the maximum busy level is 20 (at the start of the interval), and the number of running agents is 30 (all are available since everyone is at the company meeting), the broker attempts to trim back the number of agents to 20.

Assume that the company meeting ran for another half-hour, and the entire next interval elapsed without any requests. The maximum busy agents is zero, with 20 currently running. Since the Minimum Agent/Server Instances property value is set to 10, the broker attempts to trim back to 10 agents.

Keep in mind that the lower you set the Agent/Server Auto-trim Interval value, the more quickly the broker reacts to a drop in the workload. Under a reasonably steady work load, the broker should trim the number of running agents to the number needed. However, if the workload fluctuates a great deal and the Agent/Server Auto-trim Interval is set too low, the broker may trim agents that it must then restart. As stated earlier, repetitive trimming and restarting causes the system to waste resources on unnecessary process management.
To add or trim WebSpeed agents:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the WebSpeed Transaction Server broker to which you want to add or trim agents.

3. Click the WebSpeed Transaction Server broker. The **WebSpeed** details page appears.

4. If the WebSpeed broker is not started, click **Broker Control**. Then click **Start WebSpeed**.

5. When the WebSpeed broker is running, return to the **Details** page.

6. In the **Command and control** section of the page, click **Agent Pool Control**.

7. Review the **Agent pool initial configuration**, the **Agents state**, and the **Agents pool summary** information.

8. To add or trim agents, choose the corresponding option in the **Add/Trim** dropdown menu, and type the number of agents in the box provided.

9. Click **Submit**.

Deleting a WebSpeed broker

You can delete any WebSpeed broker instance listed in the management console as long as the instance is not running. If the instance is running, you must stop it and then delete it.

To delete a WebSpeed broker instance:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the WebSpeed Transaction Server broker instance you want to delete.

3. Click the WebSpeed Transaction Server broker instance. The **WebSpeed** details page appears.

4. Click **Delete**, and then click **OK** to confirm the deletion.
For additional WebSpeed information

You can find additional information about using OpenEdge Management and/or OpenEdge Explorer with WebSpeed in OpenEdge Management: Servers, DataServers, Messengers, and Adapters. See that document for details, including specifics about the following:

- Working with the WebSpeed log file monitor and viewer
- Killing a WebSpeed process
Configuring SonicMQ Adapters

You can work with SonicMQ Adapter instances in OpenEdge Management and OpenEdge Explorer, as described in the following sections:

• SonicMQ Adapter Broker configuration and administration
• Working with the SonicMQ Adapter
• Configuring a SonicMQ Adapter
• Editing a SonicMQ Adapter configuration
• Creating a SonicMQ Adapter broker
• Deleting a SonicMQ Adapter broker
• Starting or Stopping a SonicMQ Adapter
• Viewing the status of a SonicMQ Adapter broker
• Viewing the SonicMQ Adapter log file
SonicMQ Adapter Broker configuration and administration

The OpenEdge SonicMQ® Adapter allows OpenEdge applications to communicate via JMS Messaging through SonicMQ. The SonicMQ Adapter consists of three connection modes:

- **OpenEdge Adapter for SonicMQ ClientConnect (ClientConnect)** — ClientConnect is for OpenEdge clients and runs transparently as a background process in conjunction with an ABL client or OpenEdge Application Server agent process, with a single adapter process per client process. The application running on the ABL client handles messaging control. ClientConnect takes little or no configuration.

- **OpenEdge Adapter for SonicMQ ServerConnect (ServerConnect)** — ServerConnect is for OpenEdge Application Servers (WebSpeed and AppServer). With this configuration there is a single adapter process per AppServer process, allowing multiple Application Server agents to connect to this single adapter process. ServerConnect is configured at the server.

- **OpenEdge Adapter for SonicMQ BrokerConnect (BrokerConnect)** — BrokerConnect is for ABL client applications. It runs as a separate server process to handle OpenEdge client requests. BrokerConnect is a Unified Broker product, part of the AppServer administration framework. You can use OpenEdge Management or OpenEdge Explorer and the command-line tools adaptconfig and adaptman on all supported platforms, to manage BrokerConnect. You can also edit its properties in the `ubroker.properties` file. No configuration is required within the SonicMQ environment.

**Note:** For BrokerConnect, the OpenEdge installation program creates one instance of the OpenEdge SonicMQ Adapter. In most circumstances, this single adapter instance is sufficient. Although it is possible to create additional instances, there is usually no reason to run multiple OpenEdge SonicMQ Adapter instances on the same host. Each instance of BrokerConnect runs as a broker process. This process is multi-threaded, with one thread for each active client application; it can connect to any SonicMQ Broker. With the SonicMQ Adapter, you can write ABL applications that access JMS messaging using SonicMQ.

You can use OpenEdge Management or OpenEdge Explorer to configure a new instance of BrokerConnect and administer the configuration of existing instances.
Working with the SonicMQ Adapter

To work with a SonicMQ Adapter, select the instance. For information about the tasks you can perform, see the following sections:

- Configuring a SonicMQ Adapter
- Starting or Stopping a SonicMQ Adapter
- Creating a SonicMQ Adapter broker
- Starting or Stopping a SonicMQ Adapter
- Viewing the status of a SonicMQ Adapter broker
- Viewing the status of a SonicMQ Adapter broker
- Deleting a SonicMQ Adapter broker
- Viewing the SonicMQ Adapter log file
Configuring a SonicMQ Adapter

From the management console, you can view or edit the following default configuration properties of a SonicMQ Adapter:

- Broker
- Server
- SSL

You can also create a new SonicMQ Adapter instance, set environment variables, view an instance’s status, delete an instance you no longer need, or view the instance’s log file.

Each of these tasks is described in the following sections.

Viewing or modifying SonicMQ Adapter broker properties

You can view or modify general, owner information, controlling NameServer, appservice name list, logging setting, and advanced features properties for a SonicMQ Adapter instance.

To view or modify SonicMQ Adapter broker properties:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the SonicMQ Adapter broker instance whose properties you want to view or modify.
3. Click the SonicMQ Adapter broker instance. The SonicMQ Adapter details page appears.
4. In the **Command and control** section of the page, click **Configuration**. The **SonicMQ Adapter Configuration** page opens with the **Broker** properties in focus.

From this page, you can:

- View the SonicMQ Adapter instance's broker properties as described in Table 75 through Table 80. All of these tables follow this procedure.

- Click **Edit** to modify the properties. For details, see the “Starting or Stopping a SonicMQ Adapter” section on page 278.

**Table 75: SonicMQ Adapter Broker General properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Working directory</td>
<td>The SonicMQ Adapter working directory.</td>
</tr>
<tr>
<td>Port number</td>
<td>The number of the TCP/IP port that the SonicMQ Adapter broker listens to. Each broker defined in the ubroker.properties file must have a unique port number.</td>
</tr>
<tr>
<td>Auto start</td>
<td>A check box that you select if you want the SonicMQ Adapter broker to start automatically when the AdminServer starts.</td>
</tr>
</tbody>
</table>

**Table 76: SonicMQ Adapter Broker Owner Information properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group name</td>
<td>(UNIX only) The name of the group in UNIX.</td>
</tr>
<tr>
<td>Username</td>
<td>The ID of a user account that has system-administration rights.</td>
</tr>
<tr>
<td>Password</td>
<td>The user account's password. Note that this field is labeled Group Name if the AdminServer runs on a non-Windows system.</td>
</tr>
</tbody>
</table>
### Table 77: SonicMQ Adapter Broker Controlling NameServer properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Register with NameServer</td>
<td>A check box that indicates whether or not to register the SonicMQ Adapter with a controlling NameServer.</td>
</tr>
<tr>
<td>Controlling NameServer</td>
<td>The name of the NameServer with which this SonicMQ Adapter registers. (If you did not select to register the broker with a NameServer, the field is dimmed and unavailable.)</td>
</tr>
<tr>
<td>Registration mode</td>
<td>How the broker specifies its host name if it is registering with a controlling NameServer. This host name information is passed onto a client application when it attempts to connect to an application service that the broker supports. The choices are:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Register-IP Address</strong> — (Default) This setting is the most efficient mechanism and can be used in most cases. It registers with the IP address of the machine where the broker is located.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Register-LocalHost</strong> — The broker registers with the host name of the machine that it runs on. Use this setting when the broker runs on a machine with a single host name and more than one IP address.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Register-HostName</strong> — The broker registers with the values specified in the host name property. Use this setting when your clients need a fully qualified host name to connect to a broker in a different DNS domain.</td>
</tr>
<tr>
<td>Use host name</td>
<td>Specifies the host name value to be used if the <strong>Registration mode</strong> property is set to <strong>Register-HostName</strong>. In all other cases, this property is ignored.</td>
</tr>
</tbody>
</table>

### Table 78: SonicMQ Adapter Broker AppServer Name List property

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application service names</td>
<td>The application services that this SonicMQ Adapter broker registers with the controlling NameServer. A SonicMQ Adapter client connecting to the broker must specify one of these service names. You can add or delete service names from the list.</td>
</tr>
</tbody>
</table>
### Table 79: SonicMQ Adapter Broker Logging Setting properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Broker log filename</td>
<td>The broker log file for this SonicMQ Adapter broker instance.</td>
</tr>
<tr>
<td>Broker logging level</td>
<td>The amount of information written to the broker log. The possible values are as follows:</td>
</tr>
<tr>
<td></td>
<td>• None — Log no entries. This is equivalent to turning logging off.</td>
</tr>
<tr>
<td></td>
<td>• Error Only — Log OpenEdge error messages. This includes all error messages and is unrelated to the entry types specified. Errors continue to be logged at all higher levels.</td>
</tr>
<tr>
<td></td>
<td>• Basic — Logging entry types determine the logged information. Each entry type generates at least some output. This is the default.</td>
</tr>
<tr>
<td></td>
<td>• Verbose — Logging entry types determine the logged information, but it is typically more information than Basic.</td>
</tr>
<tr>
<td></td>
<td>• Extended — Logging entry types determine the logged information, but it is typically more information than Verbose.</td>
</tr>
<tr>
<td>Append to broker log file</td>
<td>A check box that you select if you want the SonicMQ Adapter broker to look for the log file each time it starts up. If the log file is present, the new information is added to the end of the existing file. If there is no log file, a new file is created.</td>
</tr>
<tr>
<td>Broker logging entry types</td>
<td>A single entry or comma delimited list of logging entry types.</td>
</tr>
<tr>
<td>Broker log file threshold size</td>
<td>A limit on how large the log file can get (in bytes). When the log is full, the broker creates a new log file with a sequence number (1 to 999,999). Possible values: 500000 - 2147483647.</td>
</tr>
<tr>
<td>Maximum number of broker log files</td>
<td>A limit on how many broker log files are kept on the system when there is a threshold limit. The value represents the number of log files including the current one. Includes any rolled over log files that already exist when process started.</td>
</tr>
</tbody>
</table>
Table 80: SonicMQ Adapter Broker Advanced Features properties (1 of 2)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Maximum client instances</strong></td>
<td>The maximum number of client connections the SonicMQ Adapter broker can support concurrently. The default value 512 is high enough to ensure that the number of client connections is unlimited. Realistically, however, the system-level resources required to support an unlimited number of clients might be exhausted before this limit is reached. If you see broker log entries that contain Exception Messages for out of space or out of memory, you might need to reduce the maximum number of client connections to a more reasonable value. Note that lowering this value can cause some client requests to be rejected (Exceeded Max Clients). Configuring and starting multiple brokers to handle higher client loads will alleviate the problem.</td>
</tr>
<tr>
<td><strong>Priority weight</strong></td>
<td>Defines a load-balancing priority for this broker relative to other brokers. You can update this property dynamically. Any changes will affect all current and new brokers.</td>
</tr>
<tr>
<td>(dynamic property)</td>
<td></td>
</tr>
<tr>
<td><strong>Registration retry</strong></td>
<td>The number of seconds that pass between broker keep-alive messages. After registering with its controlling NameServer, the SonicMQ Adapter broker periodically sends keep-alive messages to the NameServer to let the NameServer know that the broker is still active. (The default is 30.)</td>
</tr>
<tr>
<td><strong>TCP/IP version</strong></td>
<td>The TCP/IP version.</td>
</tr>
<tr>
<td><strong>Enable dynamic property updates</strong></td>
<td>Allows dynamic changes to occur to several of the instance’s properties, without requiring you to shut down the instance. This property is disabled by default. <strong>Note:</strong> Any property that is dynamic is labeled as such in this document.</td>
</tr>
<tr>
<td><strong>AppServer Keepalive</strong></td>
<td>Allows the AppServer to recognize that a client bound to it is no longer connected.</td>
</tr>
</tbody>
</table>
Table 80: SonicMQ Adapter Broker Advanced Features properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>serverASK Activity Timeout</td>
<td>Specifies the number of seconds with no activity that the AppServer waits before sending a message to the client requesting that the connection be maintained. The property is used only if the AppServer Keepalive property is set to allowServerASK.</td>
</tr>
<tr>
<td>serverASK Response Timeout</td>
<td>Specifies the number of seconds with no activity after the AppServer sends the client a message (requesting that the connection be maintained) before the client is disconnected. This property is used only if the AppServer Keepalive property is set to allowServerASK.</td>
</tr>
</tbody>
</table>

Viewing or modifying SonicMQ Adapter server properties

You can view or modify general, logging setting, or pool range properties for a SonicMQ Adapter server instance.

To view or modify SonicMQ Adapter server properties:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the SonicMQ Adapter broker instance whose server properties you want to view or modify.
3. Click the SonicMQ Adapter broker instance. The SonicMQ Adapter details page appears.
4. In the Command and control section of the page, click Configuration. The SonicMQ Adapter Configuration page opens with the Broker properties in focus.
5. Click the **Server** tab.

From this page, you can:

- View a read-only display of the SonicMQ Adapter instance’s server properties as described in **Table 81** through **Table 83**. These tables follow this procedure.
- Click **Edit** to modify the broker properties. For details, see the “Starting or Stopping a SonicMQ Adapter” section on page 278.

**Table 81: SonicMQ Adapter Server General property**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server startup parameters</td>
<td>The startup parameters that you want to specify for the SonicMQ server</td>
</tr>
</tbody>
</table>

**Table 82: SonicMQ Adapter Server Logging Settings properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server log filename</td>
<td>The server log file for this SonicMQ Adapter broker instance.</td>
</tr>
<tr>
<td>Server logging level</td>
<td>The amount of information written to the server log. The possible values are as follows:</td>
</tr>
<tr>
<td></td>
<td>- <strong>None</strong> — Log no entries. This is equivalent to turning logging off.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Error Only</strong> — Log OpenEdge error messages. This includes all error messages and is unrelated to the entry types specified. Errors continue to be logged at all higher levels.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Basic</strong> — Logging entry types determine the logged information. Each entry type generates at least some output. This is the default.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Verbose</strong> — Logging entry types determine the logged information, but it is typically more information than <strong>Basic</strong>.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Extended</strong> — Logging entry types determine the logged information, but it is typically more information than <strong>Verbose</strong>.</td>
</tr>
<tr>
<td>Append to server log file</td>
<td>A check box that you select if you want the SonicMQ Adapter broker to look for the log file each time it starts up. If the log file is present, the new information is added to the end of the existing file. If there is no log file, a new file is created.</td>
</tr>
</tbody>
</table>
### Viewing or modifying SonicMQ Adapter SSL properties

You can view or modify general and advanced features SSL properties for a SonicMQ Adapter instance.

To view or modify SonicMQ Adapter SSL properties:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the SonicMQ Adapter broker instance whose SSL properties you want to view or modify.
3. Click the SonicMQ Adapter broker instance. The SonicMQ Adapter details page appears.
4. In the Command and control section of the page, click Configuration. The SonicMQ Adapter Configuration page opens with the Broker properties in focus.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial number of threads to start</td>
<td>The number of threads you want to start initially within a SonicMQ Adapter broker process. The default is 20.</td>
</tr>
<tr>
<td>Minimum threads</td>
<td>The minimum number of threads to start. The default is 10.</td>
</tr>
<tr>
<td>Maximum threads</td>
<td>The maximum number of threads to start. The default is 30.</td>
</tr>
</tbody>
</table>
5. Click the SSL tab.

From this page, you can:

- View a read-only display of the SonicMQ Adapter instance’s SSL properties as described in Table 84 and Table 85. These tables follow this procedure.

- Click Edit to modify the broker properties, as described in Table 84 and Table 85. For details, see the “Editing a SonicMQ Adapter configuration” section on page 275.

Table 84: SonicMQ Adapter SSL General properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable SSL client connections</td>
<td>If this property is selected, all connections to this SonicMQ Adapter must use SSL tunneling.</td>
</tr>
<tr>
<td>Private key/digital certificate alias name</td>
<td>The alias name within the OpenEdge keystore of the private key and digital certificate entry to use to authenticate all connections to this SonicMQ Adapter. If not otherwise selected, the SonicMQ Adapter uses the default_server server certificate alias.</td>
</tr>
<tr>
<td>Password to access the key/certificate</td>
<td>The password to use for accessing the private key and digital certificate. You must specify a password when you specify the select an OpenEdge keystore alias name.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Your password is encrypted in the ubroker.properties file. If you set the password in ubroker.properties manually, you must specify the password as an encrypted value that you can obtain using the genpassword utility located in the bin directory of your OpenEdge installation.</td>
</tr>
<tr>
<td></td>
<td>If you use the default_server server certificate, it also has a default password that you do not need to specify.</td>
</tr>
<tr>
<td>Password to access the key/certificate Confirm</td>
<td>Confirmation of the username account password. This field appears only if you are editing the configuration.</td>
</tr>
<tr>
<td></td>
<td>Note that this field does not appear if the AdminServer is running on a non-Windows host machine.</td>
</tr>
</tbody>
</table>
Table 85: SonicMQ Adapter SSL Advanced Features properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disable SSL session caching</td>
<td>If this property is checked, caching for the SSL client session is disabled.</td>
</tr>
<tr>
<td>SSL session cache timeout</td>
<td>The length of time, in seconds, that an SSL client session is held in the session cache, during which an SSL client can resume its session. The default is 180.</td>
</tr>
</tbody>
</table>
Chapter 8: Configuring SonicMQ Adapters

Setting or deleting SonicMQ Adapter environment variables

You can set or delete SonicMQ Adapter environment variables.

To set or delete SonicMQ Adapter environment variables:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the SonicMQ Adapter broker instance whose environment variable you want to set.
3. Click the SonicMQ Adapter broker instance. The SonicMQ Adapter details page appears.
4. In the Command and control section of the page, click Configuration. The broker’s SonicMQ Adapter Configuration page appears.
5. Click Edit.
6. Click Environment Variables.
7. Enter each variable name and then provide its value in the format name=value.
8. Click Save when you finish adding (or deleting) the variables and their values.

Enabling or disabling a SonicMQ Adapter broker

You can enable or disable a SonicMQ Adapter broker.

To enable or disable a SonicMQ Adapter broker:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the SonicMQ Adapter broker instance you want to enable or disable.
3. Click the SonicMQ Adapter broker instance. The SonicMQ Adapter details page appears.
4. In the Command and control section of the page, click Control. The broker’s Control page appears.
5. Click Edit, and either select the Enabled check box to enable the broker or clear the check box to disable the broker.
6. Click Save.
Editing a SonicMQ Adapter configuration

You can edit a SonicMQ Adapter configuration in the management console.

To edit a SonicMQ Adapter configuration:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the SonicMQ Adapter broker instance whose configuration you want to edit.

3. Click the SonicMQ Adapter broker instance. The SonicMQ Adapter details page appears.

4. In the Command and control section of the page, click Configuration. The Configuration page appears.

5. Click Edit. (For details about the properties, see Table 75 through Table 85.)

6. Make the configuration changes, and then click Save.
Creating a SonicMQ Adapter broker

OpenEdge Management and OpenEdge Explorer provide a sample SonicMQ Adapter broker instance (SonicMQ1) and use it as a default. A new instance you create also uses the sample configuration, which you can modify.

To create a SonicMQ Adapter broker:

1. From the drop-down for Resources on the management console menu, click **New OpenEdge Resource** → **SonicMQ Adapter**. The **SonicMQ Adapter Configuration** page appears.

2. Type the name of the new broker instance in the field provided.

   **Note:** The SonicMQ Adapter broker instance name is case sensitive and can include any character except a period (.) or square brackets ([ ]). The name must be unique among all configured SonicMQ Adapter broker instance names.

3. Click **Save**. The **SonicMQ Adapter Configuration** page appears, allowing you to configure the instance’s properties.

You can now set the configuration properties for this instance. When a new instance is created, it has a copy of the default configuration. However, several properties, such as port numbers, must have unique values for each SonicMQ Adapter broker in order for the instance to operate properly.
Deleting a SonicMQ Adapter broker

You can delete an inactive SonicMQ Adapter broker instance from the management console. If the instance is running, you cannot delete it. You must stop the instance before you can delete it.

To delete a SonicMQ Adapter broker:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the SonicMQ Adapter broker instance you want to delete.

3. Click the SonicMQ Adapter broker instance. The SonicMQ Adapter details page appears.

4. Click Delete, and then click OK to confirm the deletion.
Starting or Stopping a SonicMQ Adapter

From the management console, you can start or stop a SonicMQ Adapter instance. You can also choose to start the instance automatically when the AdminServer starts.

To start or stop a SonicMQ Adapter instance, refer to the Starting or Stopping OpenEdge resources section in OpenEdge Management: Servers, DataServers, Messengers, and Adapters.

Starting a SonicMQ Adapter automatically

You can set a SonicMQ Adapter to start automatically when the AdminServer starts.

To start a SonicMQ Adapter automatically:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the SonicMQ Adapter broker instance you want to start automatically.
3. Click the SonicMQ Adapter broker instance. The SonicMQ Adapter details page appears.
4. In the Command and control section of the page, click Configuration. The SonicMQ Adapter Configuration page appears.
5. Click Edit.
6. On the Broker tab, select the Auto start option by clicking in the check box.
7. Click Save.
Viewing the status of a SonicMQ Adapter broker

You can view the status of an active SonicMQ Adapter broker in the management console.

To view the status of a SonicMQ Adapter broker:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the SonicMQ Adapter broker instance you want to start automatically.
3. Click the SonicMQ Adapter broker instance. The SonicMQ Adapter details page appears.
4. In the Operations Views section of the page, click Status. A Status summary appears and provides the following information:
   - **Host** — The name of the host machine.
   - **Broker Name** — The name of the broker whose status you are viewing.
   - **Operating Mode** — For SonicMQ Adapter brokers, the operating mode is stateless.
   - **Broker Status** — The current state of the broker.
   - **Broker Port** — The TCP/IP port number that the broker listens to.
   - **Broker PID** — The process ID number of the broker.
   - **Active Servers** — The number of active SonicMQ Adapter brokers.
   - **Busy Servers** — The number of SonicMQ Adapter brokers that are currently processing requests.
   - **Locked Servers** — The number of SonicMQ Adapter brokers that have been locked.
   - **Available Servers** — The number of available SonicMQ Adapter brokers.
   - **Active Clients (now, peak)** — The number of client connections at the current point, and the highest number of connections since the broker started.
   - **Client Queue Depth (cur, max)** — The number of clients waiting for agents to become available to service their request. The current value (cur) represents the number of waiting clients at the moment the status is displayed, and the maximum value (max) represents the largest number of clients waiting concurrently since the broker was started.
   - **Total Requests** — The total number of requests from clients since the broker started.
• **Rq Wait (max, avg)** — The maximum and average time the client had to wait for an available broker.

• **Rq Duration (max, avg)** — The maximum and average time the request required.

For each server, the following details are also provided:

• **Svr#** — The process ID number of the broker process

• **State** — The current state of the broker process

• **Port** — The TCP/IP port number used by the broker

• **nRq** — The total number of messages sent to the broker since it started

• **nRcvd** — The total number of messages received by the broker since it started

• **nSent** — The total number of requests sent by the broker

• **Started** — The time stamp of when the broker process started

• **Last Change** — The time stamp of when the server process last changed state
Viewing the SonicMQ Adapter log file

You can view the log file for a SonicMQ Adapter by using the log file viewer. The log file viewer allows you to examine the log file through an HTML interface.

To access the log file viewer:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the SonicMQ Adapter broker instance whose log file viewer you want to access.
3. In the Command and control section of the page, click Log File Viewer of Broker or Log File Viewer of Servers.

You can use the Log File Viewer in the following ways:

- Use the Show field to control how many database log file entries display at one time. The number entered into the Show field cannot 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 cannot be more 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.

- Click Reload after changing the values in either the Show field or the Overlap field. If you do not reload, the viewer continues to display the previous values.
- Click Go To to control which numbered entry in the log file the viewer begins its display with. For example, a value of 10 entered into the Go To field will begin the display from the tenth log file entry.

**Note:** You must click Go To after entering a value in the Go To field or the viewer will not update its display.

The default display of entries is in ascending order; choose Descending to change the display. Note that the Show field dictates the number of entries shown, regardless of whether they display in ascending or descending order.

- Click First to display the first \( x \) entries, where \( x \) is the value in the Show field.
- Click Previous 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 last \( x \) entries, where \( x \) is the value in the Show field.
- To view additional log file entries without changing your current starting log file entry, leave the Go To field blank, change the value in the Show field, and click Reload.

- If the contents of the log file have changed since you opened the viewer, the log file viewer indicates this in the Log file status field.

- OpenEdge Management and OpenEdge Explorer consider a viewer that has been inactive for more than four hours stale and at that point release ninety-five percent of any memory being held. If you try to use a stale viewer, OpenEdge Management and OpenEdge Explorer automatically reload the file. Because additional resource activity might have occurred during the viewer's inactivity, the reloaded log file view might not match the previous log file view of that resource.

- OpenEdge Management and OpenEdge Explorer consider a viewer that has been inactive for forty-eight hours dead. Once a viewer dies, OpenEdge Management and OpenEdge Explorer release all of its memory. To return to the log file displayed in a dead view, you must renavigate to it, even if you pinned up the view or saved a link to it before the viewer died.
Configuring OE Web Servers

You can work with OE Web Server instances in OpenEdge Management and OpenEdge Explorer, as described in the following sections:

- Configuring and managing OE Web Servers
- Managing OE Web Servers
- Working with REST Web Applications
Configuring and managing OE Web Servers

You can configure and manage OE Web Servers by:

- REST Management Agent
- OE Web Server
- REST Web Applications
- Apache Tomcat
- Installing a REST Management Agent into a Java container
- Managing REST Web Applications with the OE Web Server

Each of these definitions and tasks are described in the following sections.

REST Management Agent

A REST Management Agent is a component of Progress OpenEdge that you use to deploy, configure, and manage a Java container’s OpenEdge REST Web Applications. The REST Management Agent is a Java Web application that acts as an intermediary for you to communicate with the deployed REST Web Applications. It helps you enable/disable the application, collect statistics, and provide a run-time configuration for the REST Web Applications.

OpenEdge ships with a default REST Management Agent (oerm.war).

OE Web Server

An OE Web Server is a remote Web Server application that manages the deployment and management of REST Web Applications.

OpenEdge provides three clients—OpenEdge Management, OpenEdge Explorer, and RESTMAN utility—that use an OE Web Server to manage REST Web Applications.

REST Web Applications

A REST Web Application is a Web application archive file (WAR file) that includes one or more REST services. A REST service is a service defined to expose your REST service annotated ABL applications to a Web Server as a REST Web application.

A REST Web application typically includes java classes, web pages (HTML), images, and other files that together make up a web application.

Apache Tomcat

The default Web Server shipped with OpenEdge is where you can deploy the REST Web Applications that you want to make available to clients. The Tomcat server is available in your OpenEdge installation directory, OpenEdge-install-dir/servers/tomcat.
Installing a REST Management Agent into a Java container

As part of the OpenEdge installation, REST Management Agent is installed in the OpenEdge installation directory, OpenEdge-install-dir/servlets/rest.

**Note:** You need not perform the following steps if you are using the Java container installed with the Apache Tomcat shipped with OpenEdge installation as Tomcat already has the OE Web Server installed.

To start working with OE Web Server and REST Web Applications, you must configure your Web Server to install the REST Management Agent in the Java container associated with the Web Server. To configure your Web Server:

1. Copy the REST Management Agent, oerm.war, from the directory, OpenEdge-install-dir/servlets/rest, to the Web Server deployment directory. For example, if you are using a Tomcat Web Server, you copy oerm.war to the $CATALINA_HOME/webapps directory.

   **Note:** If you are using Tomcat Web Server you can also deploy oerm.war using the Tomcat manager. For more information, see the Tomcat manager documentation.

   **Note:** If the Web Server is already running, you may need to restart the Web Server for the new configuration settings to take effect. New applications are normally picked up automatically by the Tomcat Web Server.

You can query the OE Web Server Web application by browsing to the URL http://host-name:port-number/oerm/index.html to check whether the REST Management Agent has been correctly installed and configured. For instance, if you are using the Tomcat server and you are using the default ports, you can browse the URL http://localhost:8080/oerm/index.html.

Managing REST Web Applications with the OE Web Server

The OE Web Server is installed in a Web Server as a single Web application that you can instantiate as one or more OE Web Server instances. Each OE Web Server instance provides a single point of management for one or more deployed REST Web Applications. Thus, to begin working with the OE Web Server in OpenEdge Management or OpenEdge Explorer:

1. Create a new OE Web Server instance if none exists or you need another, and configure it for operation. (The OE Web Server comes installed with one local OE Web Server instance (restmgr1)).

2. For an OE Web Server instance where you want to deploy a new REST Web application, optionally review and update the default property settings that govern the run-time behavior of new REST Web application deployments. You can also
adjust these settings for each REST Web application individually after you deploy it.

3. Deploy the new REST Web application into the context of an OE Web Server instance that you have configured.

4. As you test the REST Web application, review the OE Web Server instance status and statistics, and update any OE Web Server instance properties as needed. Such changes affect all REST Web Applications deployed to the same OE Web Server instance.

**Note:** To apply changes to most REST Web application properties, you must restart the Web Server where the REST Web application is installed. You can make temporary changes to selected REST Web application properties at run time without restarting the Web Server. However, you must update the REST Web application instance configuration to make these changes permanent.

5. As you client-test the REST Web application, review the REST Web application status and statistics, and update any REST Web application properties as needed. Such changes affect only the selected REST Web application. For most of these changes, you must disable the REST Web application before you make them, then enable the REST Web application afterward to evaluate the effect.
Managing OE Web Servers

You can manage OE Web Servers by:

- Starting and Stopping Tomcat server
- Creating an OE Web Server instance
- Configuring an OE Web Server instance
- Editing an OE Web Server instance configuration
- Setting OE Web Server application defaults
- Deploying a REST Web application (.war file)
- Viewing the status of an OE Web Server instance
- Viewing OE Web Server instance statistics
- Changing OE Web Server run-time properties
- Deleting an OE Web Server instance
- Enabling or Disabling an OE Web Server instance
- Viewing the OE Web Server log file

Each of these tasks is described in the following sections.

Starting and Stopping Tomcat server

OpenEdge is shipped with a Tomcat server. The installed and configured Tomcat server provides a Java container and a web server for the REST Web application development and testing. The Tomcat server is available in your OpenEdge installation directory, $DLC/servers/tomcat.

Before you define/publish an OE Web Server or REST Service, you must start the Apache Tomcat server. You can start Apache Tomcat server installed with OpenEdge either using the Resources panel in the grid frame, or using Proenv.

To start the Tomcat server using the Resources panel:

1. Click Resources in the management console menu bar. The resource types appear in the list frame.

2. Expand the Web Server category, and click on the Tomcat instance. The Details page for that instance appears in the detail frame.

3. In the Command and control section of the page, click Control. The Apache Tomcat installed with OpenEdge page opens.

4. Click on the Start Web Server button to start Tomcat server.

Note: The button is a Toggle button depending on the status of the Tomcat server instance. If the Web server is already started, the button displays the Stop
option. To stop the Web server that is already running, click the Stop Web Server button.

To start the Tomcat server using Proenv:

1. On the Windows Start > Programs menu, select Progress > OpenEdge > Proenv. Proenv opens a command prompt and configures it for your OpenEdge installation, displaying the configuration:

   proenv>

2. Enter the following command:

   protc start

A Tomcat shell window opens displaying the status of the server as running.

**Note:** To stop the Tomcat Server, enter `protc stop` at the `proenv>` command prompt.

Creating an OE Web Server instance

You can create new OE Web Server instances from the Resources tab.

**Note:** Make sure that the REST Management Agent is deployed before you create the OE Web Server instance. You must copy the oerm.war file to the deployment server.

To create a new OE Web Server instance:

1. From the management console bar, click the Resources tab’s dropdown arrow. Choose New OpenEdge Resource → OE Web Server. The New OE Web Server name page appears.

2. Type the name of the new OE Web Server in the New OE Web Server name field.

**Note:** The OE Web Server name is case sensitive and can include any character except a period (.) or square brackets ([ ]). The name must be unique among all configured OE Web Server names.

3. Type the URL in the URL field. You can now configure the properties of the new instance. Initially, the instance is enabled only for administration, allowing you to modify its properties and deploy REST services to it without allowing premature user access to an incomplete configuration. Form the root URL according to the following format:

   http://hostmachine:port
Where:

- **hostmachine** — The name of the machine where the OEM is installed (Default: localhost).
- **port** — The HTTP port number on the machine (Default: 8089).

4. Select the OE Web Server installation location for this instance from these choices:

- **local** — The OE Web Server is installed on the machine where the selected AdminServer is running.
- **remote** — The OE Web Server is installed on a network machine that is remote from the selected AdminServer.

5. Under Deploy OE Web Server section, you can choose to deploy the OE Web Server instance on the Tomcat Manager (Apache Tomcat installed along with the Progress Developer Studio for OpenEdge).

   **Note:** Ensure that you have Tomcat Manager API enabled and a Tomcat user setup to enable this functionality.

   - **Deploy OE Web Server** — Select this check box to deploy the OE Web Server instance. Selecting this option requires you to enter the **Tomcat Manager Details** fields: Username, Password, and URL.
     - **username** — A username required to access the Tomcat Server.
     - **password** — The password for the specified username.
     - **URL** — The URL where you have the Tomcat Server installed. For example, if you are using the default ports in the Tomcat server, you can enter the URL as `http://localhost:8980/manager/text/deploy`.

   **Note:** You can choose to create an OE Web Server instance without deploying the OE Web Server.

6. Click **Save**. The new OE Web Server instance appears under the OE Web Server folder.

### Configuring an OE Web Server instance

Before deploying REST services, you must configure an OE Web Server instance to host the services that you want to deploy or import. To configure an OE Web Server instance, you set properties for it. After the OE Web Server instance is configured, you can deploy a service and configure additional properties that affect execution of the service.

To configure an OE Web Server instance, you can view or edit properties in the following categories:

- Location (read)
• Proxy Server Setting (read and edit)
• Security (read and edit)

**Setting OE Web Server instance properties**

You can set location and proxy server setting for an OE Web Server instance.

**To set OE Web Server instance properties:**

1. Click **Resources** in the management console menu bar. The resource types appear in the list frame.

2. Expand the **OE Web Server** category, and click on an instance. The **Details** page for that instance appears in the detail frame.

3. In the **Command and control** section of the page, click **Configuration**. The **OE Web Server Configuration** page opens. From this page, you can:
   • View and display of the OE Web Server properties.
   • Click **Edit** to modify the properties. For details, see the “Editing an OE Web Server instance configuration” section on page 292.

**Location properties**

You set the properties, shown in Table 86, in this category when you first create the OE Web Server instance. For more information see the “Creating an OE Web Server instance” section on page 288.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location</td>
<td>Indicates that the machine on which the OE Web Server is installed is remote.</td>
</tr>
<tr>
<td>URL</td>
<td>Specifies the URL for the OE Web Server. For example, <a href="http://localhost:8080/oerm">http://localhost:8080/oerm</a></td>
</tr>
</tbody>
</table>
Managing OE Web Servers

Proxy Server Setting properties

The properties in this category, as shown in Table 87, apply to both local and remote OE Web Server instances. They specify the information required to access any proxy server for the network on which the OE Web Server resides.

Table 87: OE Web Server Proxy Server Setting properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proxy host</td>
<td>The TCP host name for the proxy server</td>
</tr>
<tr>
<td></td>
<td>Installed Default: None</td>
</tr>
<tr>
<td>Proxy port</td>
<td>The TCP port number of the proxy service</td>
</tr>
<tr>
<td></td>
<td>Installed Default: 0</td>
</tr>
<tr>
<td>Proxy username</td>
<td>A valid username required to access the proxy server</td>
</tr>
<tr>
<td></td>
<td>Installed Default: None</td>
</tr>
<tr>
<td>Proxy password</td>
<td>The password for the specified username</td>
</tr>
<tr>
<td></td>
<td>Installed Default: None</td>
</tr>
</tbody>
</table>

Security properties

The properties in this category, as shown in Table 88, apply to both local and remote OE Web Server instances. They specify the information required to authenticate the type of user who will login to OE Web Server.

Table 88: OE Web Server Security properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>No user authentication</td>
<td>Access the OE Web Server without any user authentication (no username and password).</td>
</tr>
<tr>
<td>Use default authentication</td>
<td>Access the OE Web Server using the default username and password based on the Web Server authorization setting. This option is selected by default.</td>
</tr>
<tr>
<td>Login to Web Server with username and password</td>
<td>Access the OE Web Server using the new username and password provided by you. Selecting this option requires you to enter the username and password details in the Login page.</td>
</tr>
</tbody>
</table>
Editing an OE Web Server instance configuration

You can edit the configuration properties of an OE Web Server instance from the management console.

1. Click Resources in the management console menu bar. The resource types appear in the list frame.

2. Expand the OE Web Server category, and click on an instance. The Details page for that instance appears in the detail frame.

3. In the Command and control section of the page, click Configuration. The OE Web Server Configuration page opens.

4. Click Edit, and make the changes you want to the properties.

5. Click Save.
Setting OE Web Server application defaults

You can view and set the default values used to initialize properties for REST Web Applications that you deploy. Any changes to these defaults affect only REST Web Applications that you deploy after confirming the change. To set the values of properties for deployed REST Web Applications, set the property values individually for each REST Web application.

To view and set OE Web Server application defaults:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the OE Web Server instance whose default values you want to view.

3. Click the OE Web Server instance. The OE Web Server details page appears.

4. In the Command and control section of the page, click Defaults. The OE Web Servers Defaults page appears.

5. Click Edit.

6. Select the Runtime property whose default value you want to change. For details on the available properties, see OpenEdge Application Server: Administration.

7. Type the new value in the available field. For logical values, typing any value other than true results in a setting of false.

8. Click Submit to change the default value.

To return to the default values, you can click Reset from the Defaults page.
Deploying a REST Web application (.war file)

You can deploy a REST Web application (.war file) to the Servlet container where the OpenEdge Manager Servlet instance is installed.

A deployed REST Web application receives its initial property values from the default.props file for the REST instance. You can change these default property values after you deploy the .war file. A REST Web application always deploys with a disabled status to prevent premature or unintended client access.

To deploy a REST Web application:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the OE Web Server instance where you want to deploy the REST Web application.
3. Click the OE Web Server instance. The REST Applications page appears.
4. In the REST Applications section of the page, click the Deploy button.
5. In the Deploy REST Application .war dialog that comes up, browse for the path and name of the WAR (.war) file for the REST Web application that you want to deploy.
6. You can optionally provide the REST Web application name in the Application name field.
7. Click Deploy to deploy the REST Web application with the default settings. A confirmation message appears, and the WAR file name is identified in the REST Applications section of the REST Applications page.

Deployment generates files in the OE Web Server instance directory, one of which is applicationName.props, an XML file containing the current REST Web application property settings (initially set from default.props)

Caution: You must disable the REST Web application before you try to modify any of the properties. Two of the properties, serviceLogEntryLevel and serviceLoggingLevel can be modified even when the application is Enabled.

After deployment, to make the same REST Web application available using different information, deploy a new REST Web application with the new information, using a different REST Web application name in the same OE Web Server context. You could also deploy the REST Web application to a different OE Web Server context.

However, you can change the run-time properties of a deployed REST Web application at any time.
Viewing the status of an OE Web Server instance

You can view the status of a selected OE Web Server instance to see if the instance is running. If the instance is running, the status indicates that it is enabled.

To view the status of an OE Web Server instance:

1. Click Resources in the management console menu bar. The resource types appear in the list frame.

2. Expand the OE Web Server category, and click on an instance. The Details page for that instance appears in the detail frame.

3. Under Operations views, click Status. The following status information appears:

   • Whether the OE Web Server instance is running
   • Whether access to administrative functions is enabled

Note: You can also view the status of the OE Web Server instance on the top right corner of the Details page.
Viewing OE Web Server instance statistics

You can view and reset run-time statistics for a selected OE Web Server instance. Each OE Web Server instance creates and begins accumulating these statistics, Start Time, when it is first started by the Java Servlet Engine and Creation Time. You can also reset the statistics at any point during OE Web Server execution, so that the statistics begin accumulating again from a new Start Time.

To view the statistics of an OE Web Server instance:

1. Click **Resources** in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the OE Web Server instance whose statistics you want to view.
3. Click the OE Web Server instance. The OE Web Server details page appears.
4. Under **Operational Views**, click **Statistics**. The following information appears in the Statistics page:
   - **Active Requests** — Requests currently queued and being acted upon by the OE Web Server
   - **Authentication Errors** — Total number of authentication errors
   - **HTTP Request Errors** — Requests received from the HTTP listener, including administrative, and OE Web Server requests
   - **HTTP Requests** — Total requests received from the HTTP listener, including administrative, and REST requests
   - **Method not allowed errors** — Total errors returned by the OE Web Server, with error counts broken out at the bottom of the list for each of several error categories when total errors are greater than zero (0)
   - **REST Disabled errors** — calls while OE Web Server is disabled
   - **Services disabled** — REST Web Applications deployed to this OE Web Server that are disabled from client access
   - **URL not found errors** — calls made to unidentified resources
   - **Creation Time** — Date and time that counting began for the statistics
   - **Start Time** — Start Date and time of the OE Web Server instance
5. To reset the statistics and start accumulating all statistics from zero, click **Reset** if you want to reset the statistics.
Changing OE Web Server run-time properties

You can temporarily change some OE Web Server properties at run time without restarting your Java Servlet Engine (JSE). This is most useful for testing and debugging. The next time you restart your JSE, these settings revert to the current configuration settings for these properties stored in the properties file for the OE Web Server. The OE Web Server property file is stored in the WEB-INF directory of the deployed OE Web Server.

To change OE Web Server run-time properties:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the OE Web Server instance whose run-time properties you want to change.

3. Click the OE Web Server instance. The OE Web Server details page appears.


5. Click Edit to modify any of the following properties:

   - **adminEnabled** — Specifies whether administrator privileges are in effect, either *enabled* or *disabled*.
   - **description** — Specifies the description of the OE Web Server instance.
   - **logAppend** — Specifies whether to start a new log file each time the server is restarted, either *true* or *false*.
   - **logEntryTypes** — Specifies a list of logging entry types separated by commas.
   - **logFile** — Specifies the location and name of the OE Web Server log file.
   - **loggingLevel** — Specifies the amount of information to be written to the broker log.
   - **managerPropertiesURI** — Specifies the identifier for the OE Web Server resource.
   - **name** — Specifies the name of the OE Web Server.
   - **propertiesFileName** — Specifies the path and name of the oerm.properties file.
   - **webAppEnabled** — Controls the OE Web Server’s ability to accept and process requests to any of its deployed OE Web Server applications.
   - **workDir** — Specifies the path of the OpenEdge working directory.

6. Click Submit.
Deleting an OE Web Server instance

You can delete any OE Web Server instance listed in the management console as long as it has no REST services deployed to it.

To delete an OE Web Server instance:

1. Click Resources in the management console menu bar. The resource types appear in the list frame.

2. Expand the OE Web Server category, and click on an instance. The Details page for that instance appears in the detail frame.

3. Click Delete, and click OK to confirm the deletion.

Note: You can choose to stop the OE Web Server before deleting it. However, all the running applications get undeployed when you delete an OE Web Server.

Note: Deleting the OE Web Server instance does not undeploy it from the application server. To undeploy the OE Web Server instance from the application server, you must uninstall the OE Web Server instance from the application server after removing the instance from OEM.

Enabling or Disabling an OE Web Server instance

From the management console, you can enable or disable monitoring of an OE Web Server instance.

To enable or disable monitoring of an OE Web Server instance:

1. Click Resources in the management console menu bar. The resource types appear in the list frame.

2. Expand the OE Web Server category, and click on an instance. The Details page for that instance appears in the detail frame.

3. Click Control. That instance's OE Web Server page appears.

4. Click Edit.

5. Select the Enabled check box to enable the OE Web Server monitoring or clear the check box to disable it.
Viewing the OE Web Server log file

You can view the log file for an OE Web Server by using the log file viewer. The log file viewer allows you to examine the log file through an HTML interface.

To access the log file viewer:

1. Click Resources in the management console menu bar. The resource types appear in the list frame.

2. Expand the OE Web Server category, and click on an instance. The Details page for that instance appears in the detail frame.

3. Click Log File Viewer. The OE Web Server Log File Contents page opens. You can use the Log File Viewer in the following ways:

   • Use the Show field to control how many database log file entries display at one time. The number entered into the Show field cannot 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 cannot be more 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.

   • Choose Descending to change the default display that is in ascending order. Note that the Show field dictates the number of entries shown, regardless of whether they display in ascending or descending order.

   • Click Reload after you have changed any values for the number of lines or lines per page. If you do not reload, the viewer continues to display the previous values.

   • Click Go To to control the numbered entry in the log file from where the viewer begins its display. For example, a value of 10 entered into the Go To field will begin the display from the tenth log file entry.

   • 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 last x entries, where x is the value in the Show field. If the contents of the log file have changed since you opened the viewer, the log file viewer indicates this in the Log file status field.
Working with REST Web Applications

You can perform the following REST Web application administration tasks from the management console:

- Enabling or disabling a REST Web application for client access
- Viewing REST Web application statistics
- Using the REST Web application log file viewer
- Updating a deployed REST Web application
- SSL support for REST Web Applications
- Undeploying a REST Web application

Enabling or disabling a REST Web application for client access

You can enable any deployed REST Web application, making it accessible or inaccessible to user requests on the network. A deployed REST Web application is disabled by default.

To enable or disable a REST Web application for client access:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.
2. Filter or search for the OE Web Server instance where the REST Web application you want to enable or disable is deployed.
3. Click the OE Web Server instance. The OE Web Server Details page appears.
4. Click the REST Applications link. The REST Applications page appears.
5. Under the REST Applications section, select the deployed REST Web application.
6. Click Enable in the Actions column to enable the application. The Enabled column displays the status of the Enabled option.
7. To disable the enabled application, click Disable in the Actions column.
Viewing REST Web application statistics

You can view (and reset) run-time statistics for a selected REST Web application. Each REST Web application creates and begins accumulating these statistics (Start Time) from the time that its parent OE Web Server instance starts up (Creation Time). You can reset the REST Web application statistics at any point while the OE Web Server instance is running. A reset clears and begins accumulating the statistics again with a new Start Time.

To view the run-time statistics for a REST Web application:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the OE Web Server instance where the REST Web application whose runtime statistics you want to view is deployed.

3. Click the OE Web Server instance. The OE Web Server details page appears.

4. Click the REST Applications link. The REST Applications page appears.

5. Under the REST Applications section, select the deployed REST Web application.

6. Under REST Application Details section, click the Statistics tab to view a read-only statistics presentation with the following information:

   - **Adapter errors** — Total number of OE Web Server failures
   - **Application errors** — Total number of errors related to REST Web application
   - **Average response time** — Average response time of each request response.
   - **Failed admin logins** — Total number of failed administrator logins
   - **Failed admin requests** — Total number of administrator requests that failed due to missing request information
   - **Failed app server connections** — Total number of failed AppServer connections in the connection pool
   - **Failed user logins** — Total number of failed user logins
   - **Failed user requests** — Total number of requests that failed due to missing request information
   - **Max response time** — Maximum response time of each request response
   - **Successful admin logins** — Total number of successful administrator logins
   - **Successful admin requests** — Total number of successful administrator requests
- **Successful app server connections** — Total number of successful AppServer connections in the connection pool for this REST Web application
- **Successful user logins** — Total number of successful user logins
- **Successful user requests** — Total number of successful user requests handled by the selected REST Web application
- **Creation time** — Date and time that counting began for the statistics
- **Start time** — Start date and time of the REST Web application

7. To reset the statistics and start accumulating from zero, click **Reset** if you want to reset the statistics.
Using the REST Web application log file viewer

To view the contents of a REST Web application log file, access the viewer associated with each individual log file.

The log file viewer allows you to examine the contents of a REST Web application log file in HTML format. You can access a log file viewer from the following location:

- Click the Log File Viewer button in the REST Applications section of the REST Applications page. The log file viewer button appears with every deployed application.

The following information will help you use the REST Web application log file viewer:

- Use the Show field to control how many OE Web Server log file entries appear at one time. The number entered into the Show field cannot 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 not be more 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.

- Click Reload after changing the values in either the Show field or Overlap field. OpenEdge Management prompts you to click Reload. A warning message that reads changed, reload needed appears in the File log status field in the log file summary section of the page.

If you do not reload, the viewer displays the previous values.

- Click Go To to control which numbered entry in the log file the viewer begins its display with. For example, a value of 10 entered into the Go To field begins the display from the tenth log file entry.

Note: You must click Go To after entering a value in the Go To field, or the viewer will not update its display.

- The default display of entries is in ascending order. Choose Descending to change the display. Note that the Show field dictates the number of entries shown, whether they appear in ascending or descending order.

- 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 last $x$ entries, where $x$ is the value in the Show field.

- To view additional log file entries without changing your current starting log file entry, leave the Go To field blank, change the value in the Show field, and click Reload.
Chapter 9: Configuring OE Web Servers

Updating a deployed REST Web application

You can update the definition and deployment information for a deployed OE Web Server at any time.

Caution: You can freely update the definition and deployment information for a deployed REST Web application during development. However, once you deploy and enable a REST Web application for client access, avoid making any changes to this information, as client implementations depend on its stability.

After deployment, to make the same REST Web application available using different information, deploy a new REST Web application with the new information, using a different REST Web application name in the same OE Web Server context. You can also deploy the REST Web application to a different OE Web Server context.

Setting OE Web Server application properties

You can view and set the value of each property for a deployed REST Web application, or you can reset the values of all properties to the current REST Web application defaults (stored in the default.props file for the OE Web Server instance where the REST Web application is deployed).

You must disable a REST Web application before you can set most properties for it. You can verify if the REST Web application is disabled by checking its status. The only properties that you can set for an enabled OE Web Server are:

- serviceLogEntryLevel
- serviceLoggingLevel

If you set other properties while the REST Web application is enabled, the property value changes take effect only after you disable and then enable the REST Web application again.

Note: Unlike OE Web Server instance properties, the values for REST Web application properties are not stored in the ubroker.properties file. They are stored in the applicationname.props XML file located in the Web application Servlet directory for the OE Web Server instance where the REST Web application is deployed.

To view and set REST Web application properties:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the OE Web Server instance where the REST Web application whose properties you want to view and set is deployed.

3. Click the OE Web Server instance. The OE Web Server details page appears.

4. Click the REST Applications link. The REST Applications page appears.

5. Under the REST Applications section, select the deployed REST Web application.
6. Under **REST Application Details** section, select the **Properties** tab to view the application properties of the deployed REST Web application.

For details on the properties that can appear in this list, see *OpenEdge Application Server: Administration*.

**Note:** The actual list of properties displayed is a subset of all the properties, and changes depending on whether the session model of the OE Web Server is session managed or session free. To verify the session model of an OE Web Server, check its status. For more information, see the "To disable the enabled application, click Disable in the Actions column." section on page 300.

7. Type the new value in the available field. For logical values, entering any value other than true results in a setting of false.

8. Click **Save** when you finish. The changes are then written to the `applicationname.props` file for the selected REST Web application.

### SSL support for REST Web Applications

To have a secure connection between the REST Web application and the AppServer, the following conditions must be met:

- You must obtain and install public key certificates for the host machine in which the OE Web Server resides.

  **Note:** You can add any additional public certificates to psccerts.jar (located in WEB-INF/lib folder and contains all the certificates for the REST Web application required to connect to the OpenEdge AppServer) using the certutil.bat. If you do not want to use pscerts.jar, you can uncomment the property certlocation in the REST Web application's `web.xml` file.

- The REST Web application must send SSL requests to the AppServer to process the client requests. To configure the service to send SSL requests, set the value of the **ApplicationService Protocol** property to AppServerS or AppServerDCS. For more information on setting properties for REST Web Applications, see Setting OE Web Server application properties.

- The AppServer must be SSL-enabled, meaning that it accepts SSL requests from the OE Web Server (or other clients). OpenEdge supports Secure Socket Layer (SSL)-enabled communications between the Web server where the REST Management Agent runs and a Web server client. To achieve this, you must enable the REST Web application for HTTPS (SSL) connections. For more information on enabling SSL, see Viewing or modifying AppServer broker properties.

For details on SSL support in OpenEdge, including configuring and operating a client of an SSL-enabled AppServer, see *OpenEdge Getting Started: Core Business Services - Security and Auditing*. 
Undeploying a REST Web application

You can undeploy a deployed REST Web application, which removes its deployment information, its properties, and other settings from the OE Web Server instance context.

You must disable a REST Web application before you can undeploy it. You can verify if the REST Web application is disabled by checking its status.

To undeploy a REST Web application:

1. Click Resources in the management console menu. All resources managed by your console appear in the grid frame.

2. Filter or search for the OE Web Server instance where the REST Web application you want to enable or disable is deployed.

3. Click the OE Web Server instance. The OE Web Server details page appears.

4. Click the REST Applications link. The REST Applications details page appears.

5. Under the REST Applications section, select the deployed REST Web application.

6. Click Undeploy in the OE Web Server's column to undeploy the application.

The application is undeployed. If the REST Web application is already disabled, it is now undeployed from the OE Web Server instance context and removed from the management console. If the REST Web application is still enabled, an error appears indicating that you must disable the REST Web application before undeploying it.

Note: You can undeploy multiple REST Web Applications at the same time, by selecting the checkbox against the application name and clicking the Undeploy button on the top left corner of the REST Application section.
Third Party Acknowledgements

One or more products in the Progress OpenEdge v11.3 release includes third party components covered by licenses that require that the following documentation notices be provided:

Progress OpenEdge v11.3 may incorporate ANT v1.5.4. Such technology is subject to the following terms and conditions: The Apache Software License, Version 1.1, applies to all versions of up to ant 1.6.0 included. The Apache Software License, Version 1.1 - Copyright (C) 2000-2003 The Apache Software Foundation. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The names "Ant" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org. 5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation. THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
Appendix A: Third Party Acknowledgements

ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

Progress OpenEdge v11.3 may incorporate Apache Avalon 2002/08/06, Apache Batik v1.1, Apache Jakarta POI v1.1, Apache POI 2003/05/14, Regexp (Pure Java Regular Expression) v1.2, and XMLTask v1.15.1 from The Apache Software Foundation. Such technologies are subject to the following terms and conditions: The Apache Software License, Version 1.1 Copyright (c) 2000 The Apache Software Foundation. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The names "Apache" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org. 5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation. THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>. Portions of this software are based upon public domain software originally written at the National Center for Supercomputing Applications, University of Illinois, Urbana-Champaign.

Progress OpenEdge v11.3 may incorporate Apache JAXP v1.3.04 from The Apache Software Foundation. Such technology is subject to the following terms and conditions: The Apache Software License, Version 1.1 Copyright (c) 1999-2003 The Apache Software Foundation. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments

308
normally appear. 4. The names "Xalan" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org. 5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation. THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation and was originally based on software copyright (c) 1999, Lotus Development Corporation., http://www.lotus.com. For more information on the Apache Software Foundation, please see http://www.apache.org/.

Progress OpenEdge v11.3 may incorporate Crimson v1.1.3 from Progress Extensions for Eclipse v2.2.1. Such technology is subject to the following terms and conditions: The Apache Software License, Version 1.1 Copyright (c) 1999-2003 The Apache Software Foundation. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The names "Xerces" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org. 5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation. THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation and was originally based on software copyright (c) 1999, International Business Machines, Inc., http://www.ibm.com. For more information on the Apache Software Foundation, please see <http://www.apache.org/>. 
Appendix A: Third Party Acknowledgements

Progress OpenEdge v11.3 may incorporate jfor v0.7.2 from jfor. Such technology is subject to the following terms and conditions: jfor Apache-Style Software License. Copyright (c) 2002 by the jfor project. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the jfor project (http://www.jfor.org)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The name "jfor" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact info@jfor.org. 5. Products derived from this software may not be called "jfor", nor may "jfor" appear in their name, without prior written permission of info@jfor.org. THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE JFOR PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate Apache LogKit v1.2 from The Apache Software Foundation. Such technologies are subject to the following terms and conditions: The Apache Software License, Version 1.1 -- Copyright (C) 1997-2003 The Apache Software Foundation. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The names "LogKit", "Jakarta" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org. 5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation. THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. This
software consists of voluntary contributions made by many individuals on behalf of the
Apache Software Foundation. For more information on the Apache Software
Foundation, please see <http://www.apache.org/>.

Progress OpenEdge v11.3 may incorporate Xerces for Java XML Parser v2.6.2. Such
technology is subject to the following terms and conditions: The Apache Software
License, Version 1.1 Copyright (c) 1999 The Apache Software Foundation. All rights
reserved. Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of
conditions and the following disclaimer.
2. Redistributions in binary form must
reproduce the above copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the distribution.
3. The
end-user documentation included with the redistribution, if any, must include the
following acknowledgment: "This product includes software developed by the Apache
Software Foundation (http://www.apache.org/)." Alternately, this acknowledgment
may appear in the software itself, if and wherever such third-party acknowledgments
normally appear.
4. The names "Xerces" and "Apache Software Foundation" must not
be used to endorse or promote products derived from this software without prior written
permission. For written permission, please contact apache@apache.org.
5. Products
derived from this software may not be called "Apache", nor may "Apache" appear in
their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED
WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE. This software consists of voluntary contributions
made by many individuals on behalf of the Apache Software Foundation and was
originally based on software copyright (c) 1999, International Business Machines, Inc.,
http://www.ibm.com. For more information on the Apache Software Foundation,
please see <http://www.apache.org/>.

Progress OpenEdge v11.3 may incorporate SOAP v2.3.1 from Apache Foundation.
Such technology is subject to the following terms and conditions: The Apache Software
License, Version 1.1

Copyright (c) 1999 The Apache Software Foundation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are
permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of
conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of
conditions and the following disclaimer in the documentation and/or other materials
provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation ([http://www.apache.org/](http://www.apache.org/))."

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "SOAP" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

Progress OpenEdge v11.3 may incorporate ANTLR (Another Tool for Language Recognition) v2.7.6. Such technology is subject to the following terms and conditions: ANTLR 3 License [The BSD License] Copyright (c) 2003-2006, Terence Parr All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of the author nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate Castor v0.9.3 from castor.org. Such technology is subject to the following terms and conditions: Original Intalio license -
Appendix A: Third Party Acknowledgements

REPRESENTATION OR WARRANTY OF ANY KIND CONCERNING THE MERCHANTABILITY OF THIS SOFTWARE OR ITS FITNESS FOR ANY PARTICULAR PURPOSE. decstrtod.c License: The author of this software is David M. Gay. Copyright (C) 1998-2001 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

decstrtof.c License: The author of this software is David M. Gay. Copyright (C) 1998, 2000 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

dmisc.c License: The author of this software is David M. Gay. Copyright (C) 1998 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
gdtoa.c License: The author of this software is David M. Gay. Copyright (C) 1998, 1999 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY
SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. gdtoa.h License: The author of this software is David M. Gay. Copyright (C) 1998 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. gdtoaimp.h License: The author of this software is David M. Gay. Copyright (C) 1998-2000 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. gethex.c License: The author of this software is David M. Gay. Copyright (C) 1998 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. gmisc.c License: The author of this software is David M. Gay. Copyright (C) 1998 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of Lucent or any of its entities not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT
SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY SPECIAL,
INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER
RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION
OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF
OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

hd_init.c License: The author of this software is David M. Gay. Copyright (C) 2000 by
Lucent Technologies All Rights Reserved Permission to use, copy, modify, and
distribute this software and its documentation for any purpose and without fee is hereby
granted, provided that the above copyright notice appear in all copies and that both that
the copyright notice and this permission notice and warranty disclaimer appear in
supporting documentation, and that the name of Lucent or any of its entities not be
used in advertising or publicity pertaining to distribution of the software without specific,
written prior permission. LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD
TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL LUCENT OR ANY OF ITS
ENTITIES BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL
DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR
OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE
USE OR PERFORMANCE OF THIS SOFTWARE.

misc.c License: The author of this software is David M. Gay. Copyright (C) 1998, 1999 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted, provided that the
above copyright notice appear in all copies and that both that the copyright notice and
this permission notice and warranty disclaimer appear in supporting documentation,
and that the name of Lucent or any of its entities not be used in advertising or publicity
pertaining to distribution of the software without specific, written prior permission.
LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE,
INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS.
IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY
SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION
OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF
OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

smisc.c License: The author of this software is David M. Gay. Copyright (C) 1998, 1999 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted, provided that the
above copyright notice appear in all copies and that both that the copyright notice and
this permission notice and warranty disclaimer appear in supporting documentation,
and that the name of Lucent or any of its entities not be used in advertising or publicity
pertaining to distribution of the software without specific, written prior permission.
LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE,
INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS.
IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY
SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION
OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF
OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

strtodg.c License: The author of this software is David M. Gay. Copyright (C) 1998-2001 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted, provided that the above copyright
notice appear in all copies and that both that the copyright notice and this permission
notice and warranty disclaimer appear in supporting documentation, and that the name
of Lucent or any of its entities not be used in advertising or publicity pertaining to
distribution of the software without specific, written prior permission.

LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE,
INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS.
IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY
SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS,
WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS
ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THIS SOFTWARE.

strtord.c License: The author of this software is David M. Gay. Copyright (C) 1998, 2000 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted, provided that the
above copyright notice appear in all copies and that both that the copyright notice and
this permission notice and warranty disclaimer appear in supporting documentation,
and that the name of Lucent or any of its entities not be used in advertising or publicity
pertaining to distribution of the software without specific, written prior permission.

LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE,
INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS.
IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY
SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS,
WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS
ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THIS SOFTWARE.

sum.c License: The author of this software is David M. Gay. Copyright (C) 1998 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted, provided that the above copyright
notice appear in all copies and that both that the copyright notice and this permission
notice and warranty disclaimer appear in supporting documentation, and that the name
of Lucent or any of its entities not be used in advertising or publicity pertaining to
distribution of the software without specific, written prior permission.

LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE,
INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS.
IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY
SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS,
WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS
ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

ulp.c License: The author of this software is David M. Gay. Copyright (C) 1998, 1999 by Lucent Technologies All Rights Reserved Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted, provided that the above copyright
notice appear in all copies and that both that the copyright notice and this permission notice and warranty disclaimer appear in
supporting documentation, and that the name of Lucent or any of its entities not be
used in advertising or publicity pertaining to distribution of the software without specific,
written prior permission.

LUCENT DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE,
INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS.
IN NO EVENT SHALL LUCENT OR ANY OF ITS ENTITIES BE LIABLE FOR ANY
SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS,
WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
Appendix A: Third Party Acknowledgements

Progress OpenEdge v11.3 may incorporate DOM4J v1.6.1 from MetaStuff, Ltd. Such technology is subject to the following terms and conditions: Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved. Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain copyright statements and notices. Redistributions must also contain a copy of this document. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The name "DOM4J" must not be used to endorse or promote products derived from this Software without prior written permission of MetaStuff, Ltd. For written permission, please contact dom4j-info@metastuff.com. 4. Products derived from this Software may not be called "DOM4J" nor may "DOM4J" appear in their names without prior written permission of MetaStuff, Ltd. DOM4J is a registered trademark of MetaStuff, Ltd. 5. Due credit should be given to the DOM4J Project - http://www.dom4j.org THIS SOFTWARE IS PROVIDED BY METASTUFF, LTD. AND CONTRIBUTORS "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL METASTUFF, LTD. OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate GraphicsMagick v1.3.14 from GraphicsMagick Group. Such technology is subject to the following terms and conditions:

.. This text is in reStucturedText format, so it may look a bit odd. .. See http://docutils.sourceforge.net/rst.html for details.

======================================
GraphicsMagick Copyrights and Licenses
======================================

This file is part of the GraphicsMagick software distributed by the GraphicsMagick Group.

[*Please note that the legal community considers 15 or more total lines of code or text (not necessarily contiguous) to be significant for the purposes of copyright. Repeated changes such as renaming a symbol has similar significance to changing one line of code.*]

The licenses which components of this software fall under are as follows.

1) In November 2002, the GraphicsMagick Group created GraphicsMagick from ImageMagick Studio's ImageMagick and applied the "MIT" style license: Copyright (C) 2002 - 2012 GraphicsMagick Group
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

2) In October 1999, ImageMagick Studio assumed the responsibility for the development of ImageMagick (forking from the distribution by E. I. du Pont de Nemours and Company) and applied a new license:

Copyright (C) 2002 ImageMagick Studio, a non-profit organization dedicated to making software imaging solutions freely available.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files ("ImageMagick"), to deal in ImageMagick without restriction, including without limitation

- the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of ImageMagick, and to permit persons to whom the ImageMagick is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of ImageMagick.

The software is provided "as is", without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose and noninfringement. In no event shall ImageMagick Studio be liable for any claim, damages or other liability, whether in an action of contract, tort or otherwise, arising from, out of or in connection with ImageMagick or the use or other dealings in ImageMagick.

Except as contained in this notice, the name of the ImageMagick Studio shall not be used in advertising or otherwise to promote the sale, use or other dealings in ImageMagick without prior written authorization from the ImageMagick Studio.

3) From 1991 to October 1999 (through ImageMagick 4.2.9), ImageMagick was developed and distributed by E. I. du Pont de Nemours and Company: Copyright 1999 E. I. du Pont de Nemours and Company

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files ("ImageMagick"), to deal in ImageMagick without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of ImageMagick, and to permit persons to whom the ImageMagick is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of ImageMagick.
The software is provided "as is", without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose and noninfringement.

In no event shall E. I. du Pont de Nemours and Company be liable for any claim, damages or other liability, whether in an action of contract, tort or otherwise, arising from, out of or in connection with ImageMagick or the use or other dealings in ImageMagick.

Except as contained in this notice, the name of the E. I. du Pont de Nemours and Company shall not be used in advertising or otherwise to promote the sale, use or other dealings in ImageMagick without prior written authorization from the E. I. du Pont de Nemours and Company.

4) The GraphicsMagick Base64Decode() and Base64Encode() functions are based on source code obtained from OpenSSH. This source code is distributed under the following license: Copyright (c) 2000 Markus Friedl. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

5) Many of the pattern images in coders/logo.c are derived from XFig, which is distributed under the following license:

| FIG : Facility for Interactive Generation of figures |
| Copyright (c) 1985-1988 by Supoj Sutanthavibul |
| Parts Copyright (c) 1989-2000 by Brian V. Smith |
| Parts Copyright (c) 1991 by Paul King |

Any party obtaining a copy of these files is granted, free of charge, a full and unrestricted irrevocable, world-wide, paid up, royalty-free, nonexclusive right and license to deal in this software and documentation files (the "Software"), including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons who receive copies from any such party to do so, with the only requirement being that this copyright notice remain intact.
6) The documentation for the composition operators is copied from the rlecomp manual page, which is authored by Rod Bogart and John W. Peterson. Rlecomp is part of the Utah Raster Toolkit distributed by the University of Michigan and the University of Utah. The copyright for this manual page is as follows: Copyright (c) 1986, University of Utah

This software is copyrighted as noted below. It may be freely copied, modified, and redistributed, provided that the copyright notice is preserved on all copies.

There is no warranty or other guarantee of fitness for this software, it is provided solely "as is". Bug reports or fixes may be sent to the author, who may or may not act on them as he desires.

You may not include this software in a program or other software product without supplying the source, or without informing the end-user that the source is available for no extra charge.

If you modify this software, you should include a notice giving the name of the person performing the modification, the date of modification, and the reason for such modification.

7) The source code comprising swab.c is originally derived from libriff which has the following license:

| Copyright (c) 1988-1997 Sam Leffler
| Copyright (c) 1991-1997 Silicon Graphics, Inc.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that (i) the above copyright notices and this permission notice appear in all copies of the software and related documentation, and (ii) the names of Sam Leffler and Silicon Graphics may not be used in any advertising or publicity relating to the software without the specific, prior written permission of Sam Leffler and Silicon Graphics.

THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL SAM LEFFLER OR SILICON GRAPHICS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

8) The C++ API known as "Magick++", and which resides in the Magick++ directory, is distributed under the following license: Copyright 1999 - 2003 Bob Friesenhahn <bfriesen@simple.dallas.tx.us>

Permission is hereby granted, free of charge, to any person obtaining a copy of the source files and associated documentation files ("Magick++"), to deal in Magick++ without restriction, including without limitation of the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of Magick++, and to permit persons to whom the Magick++ is furnished to do so, subject to the following conditions:

This copyright notice shall be included in all copies or substantial portions of Magick++. The copyright to Magick++ is retained by its author and shall not be subsumed or replaced by any other copyright.
The software is provided "as is", without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose and noninfringement. In no event shall Bob Friesenhahn be liable for any claim, damages or other liability, whether in an action of contract, tort or otherwise, arising from, out of or in connection with Magick++ or the use or other dealings in Magick++.

9) The GraphicsMagick HaldClutImagePixels() function in magick/hclut.c is based on source code from the HaldCLUT package by Eskil Steenberg (http://www.quelsolaar.com/technology/clut.html) which is distributed under the following license: Copyright (c) 2005 Eskil Steenberg. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

10) GraphicsMagick makes use of third-party "delegate" libraries to support certain optional features. These libraries bear their own copyrights and licenses, which may be more or less restrictive than the GraphicsMagick license. For convenience, when GraphicsMagick is bundled with (or compiled with) "delegate" libraries, a copy of the licenses for these libraries is provided in a "licenses" directory.

--------------------------------------------------------------------------

... |copy|   unicode:: U+000A9 .. COPYRIGHT SIGN

Copyright |copy| GraphicsMagick Group 2002 - 2011

OVERVIEW and LEGAL ISSUES from jpeg.txt file (from GraphicsMagick):

The Independent JPEG Group's JPEG software

=========================================================================

README for release 6b of 27-Mar-1998

=========================================================================
This distribution contains the sixth public release of the Independent JPEG Group’s free JPEG software. You are welcome to redistribute this software and to use it for any purpose, subject to the conditions under LEGAL ISSUES, below.

Serious users of this software (particularly those incorporating it into larger programs) should contact IJG at jpeg-info@uunet.uu.net to be added to our electronic mailing list. Mailing list members are notified of updates and have a chance to participate in technical discussions, etc.

This software is the work of Tom Lane, Philip Gladstone, Jim Boucher, Lee Crocker, Julian Minguillon, Luis Ortiz, George Phillips, Davide Rossi, Guido Vollbeding, Ge‘ Weijers, and other members of the Independent JPEG Group. IJG is not affiliated with the official ISO JPEG standards committee.

OVERVIEW
========

This package contains C software to implement JPEG image compression and decompression. JPEG (pronounced "jay-peg") is a standardized compression method for full-color and gray-scale images. JPEG is intended for compressing "real-world" scenes; line drawings, cartoons and other non-realistic images are not its strong suit. JPEG is lossy, meaning that the output image is not exactly identical to the input image. Hence you must not use JPEG if you have to have identical output bits. However, on typical photographic images, very good compression levels can be obtained with no visible change, and remarkably high compression levels are possible if you can tolerate a low-quality image. For more details, see the references, or just experiment with various compression settings. This software implements JPEG baseline, extended-sequential, and progressive compression processes. Provision is made for supporting all variants of these processes, although some uncommon parameter settings aren’t implemented yet.

For legal reasons, we are not distributing code for the arithmetic-coding variants of JPEG; see LEGAL ISSUES. We have made no provision for supporting the hierarchical or lossless processes defined in the standard.

We provide a set of library routines for reading and writing JPEG image files, plus two sample applications "cjjpeg" and "djjpeg", which use the library to perform conversion between JPEG and some other popular image file formats. The library is intended to be reused in other applications.

In order to support file conversion and viewing software, we have included considerable functionality beyond the bare JPEG coding/decoding capability; for example, the color quantization modules are not strictly part of JPEG decoding, but they are essential for output to colormapped file formats or colormapped displays. These extra functions can be compiled out of the library if not required for a particular application. We have also included "jpegtran", a utility for lossless transcoding between different JPEG processes, and "rdjpgcom" and "wrjpgcom", two simple applications for inserting and extracting textual comments in JFIF files.

The emphasis in designing this software has been on achieving portability and flexibility, while also making it fast enough to be useful. In particular, the software is not intended to be read as a tutorial on JPEG. (See the REFERENCES section for introductory material.) Rather, it is intended to be reliable, portable, industrial-strength code. We do not claim to have achieved that goal in every aspect of the software, but we strive for it.
We welcome the use of this software as a component of commercial products.

No royalty is required, but we do ask for an acknowledgement in product documentation, as described under LEGAL ISSUES.

LEGAL ISSUES

==========

In plain English:

1. We don't promise that this software works. (But if you find any bugs, please let us know!)

2. You can use this software for whatever you want. You don't have to pay us.

3. You may not pretend that you wrote this software. If you use it in a program, you must acknowledge somewhere in your documentation that you've used the IJG code.

In legalese:

The authors make NO WARRANTY or representation, either express or implied, with respect to this software, its quality, accuracy, merchantability, or fitness for a particular purpose. This software is provided "AS IS", and you, its user, assume the entire risk as to its quality and accuracy.

This software is copyright (C) 1991-1998, Thomas G. Lane.

All Rights Reserved except as specified below.

Permission is hereby granted to use, copy, modify, and distribute this software (or portions thereof) for any purpose, without fee, subject to these conditions:

(1) If any part of the source code for this software is distributed, then this README file must be included, with this copyright and no-warranty notice unaltered; and any additions, deletions, or changes to the original files must be clearly indicated in accompanying documentation.

(2) If only executable code is distributed, then the accompanying documentation must state that "this software is based in part on the work of the Independent JPEG Group".

(3) Permission for use of this software is granted only if the user accepts full responsibility for any undesirable consequences; the authors accept NO LIABILITY for damages of any kind.

These conditions apply to any software derived from or based on the IJG code, not just to the unmodified library. If you use our work, you ought to acknowledge us.

Permission is NOT granted for the use of any IJG author's name or company name in advertising or publicity relating to this software or products derived from it. This software may be referred to only as "the Independent JPEG Group's software".

We specifically permit and encourage the use of this software as the basis of commercial products, provided that all warranty or liability claims are assumed by the product vendor.

ansi2knr.c is included in this distribution by permission of L. Peter Deutsch, sole proprietor of its copyright holder, Aladdin Enterprises of Menlo Park, CA.
ansi2knr.c is NOT covered by the above copyright and conditions, but instead by the usual distribution terms of the Free Software Foundation; principally, that you must include source code if you redistribute it. (See the file ansi2knr.c for full details.) However, since ansi2knr.c is not needed as part of any program generated from the IJG code, this does not limit you more than the foregoing paragraphs do.

The Unix configuration script "configure" was produced with GNU Autoconf.

It is copyright by the Free Software Foundation but is freely distributable.

The same holds for its supporting scripts (config.guess, config.sub, ltconfig, ltmain.sh). Another support script, install-sh, is copyright by M.I.T. but is also freely distributable.

It appears that the arithmetic coding option of the JPEG spec is covered by patents owned by IBM, AT&T, and Mitsubishi. Hence arithmetic coding cannot legally be used without obtaining one or more licenses. For this reason, support for arithmetic coding has been removed from the free JPEG software. (Since arithmetic coding provides only a marginal gain over the unpatented Huffman mode, it is unlikely that very many implementations will support it.)

So far as we are aware, there are no patent restrictions on the remaining code.

The IJG distribution formerly included code to read and write GIF files.

To avoid entanglement with the Unisys LZW patent, GIF reading support has been removed altogether, and the GIF writer has been simplified to produce "uncompressed GIFs". This technique does not use the LZW algorithm; the resulting GIF files are larger than usual, but are readable by all standard GIF decoders.

We are required to state that "The Graphics Interchange Format(c) is the Copyright property of CompuServe Incorporated. GIF(sm) is a Service Mark property of CompuServe Incorporated."

Contents of png.txt file (from GraphicsMagick):

This copy of the libpng notices is provided for your convenience. In case of any discrepancy between this copy and the notices in the file png.h that is included in the libpng distribution, the latter shall prevail.

COPYRIGHT NOTICE, DISCLAIMER, and LICENSE:

If you modify libpng you may insert additional notices immediately following this sentence.

libpng versions 1.0.7, July 1, 2000, through 1.2.0, September 1, 2001, are Copyright (c) 2000 Glenn Randers-Pehrson and are distributed according to the same disclaimer and license as libpng-1.0.6 with the following individuals added to the list of Contributing Authors

Simon-Pierre Cadieux

Eric S. Raymond

Gilles Vollant

and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your
particular purposes or needs. This library is provided with all faults, and the entire risk
of satisfactory quality, performance, accuracy, and effort is with the user.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are Copyright (c)
1998, 1999 Glenn Randers-Pehrson, and are distributed according to the same
disclaimer and license as libpng-0.96, with the following individuals added to the list of
Contributing Authors:

Tom Lane

Glenn Randers-Pehrson

Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are Copyright (c) 1996, 1997
Andreas Dilger

Distributed according to the same disclaimer and license as libpng-0.88, with the
following individuals added to the list of Contributing Authors:

John Bowler

Kevin Bracey

Sam Bushell

Magnus Holmgren

Greg Roelofs

Tom Tanner

libpng versions 0.5, May 1995, through 0.88, January 1996, are Copyright (c) 1995,
1996 Guy Eric Schalnat, Group 42, Inc. For the purposes of this copyright and license,
"Contributing Authors" is defined as the following set of individuals:

Andreas Dilger

Dave Martindale

Guy Eric Schalnat

Paul Schmidt

Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group
42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the
warranties of merchantability and of fitness for any purpose. The Contributing Authors
and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary,
or consequential damages, which may result from the use of the PNG Reference
Library, even if advised of the possibility of such damage. Permission is hereby granted
to use, copy, modify, and distribute this source code, or portions hereof, for any
purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.

2. Altered versions must be plainly marked as such and must not be misrepresented
as being the original source.
3. This Copyright notice may not be removed or altered from any source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

A "png_get_copyright" function is available, for convenient use in "about" boxes and the like:

```c
printf("%s",png_get_copyright(NULL));
```

Also, the PNG logo (in PNG format, of course) is supplied in the files "pngbar.png" and "pngbar.jpg" (88x31) and "pngnow.png" (98x31).

Libpng is OSI Certified Open Source Software. OSI Certified Open Source is a certification mark of the Open Source Initiative.

Glenn Randers-Pehrson
randeg@alum.rpi.edu
September 1, 2001

Contents of tiff.txt file (from GraphicsMagick):

Copyright (c) 1988-1997 Sam Leffler
Copyright (c) 1991-1997 Silicon Graphics, Inc.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that (i) the above copyright notices and this permission notice appear in all copies of the software and related documentation, and (ii) the names of Sam Leffler and Silicon Graphics may not be used in any advertising or publicity relating to the software without the specific, prior written permission of Sam Leffler and Silicon Graphics. THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND,

EXPRESSION, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL SAM LEFFLER OR SILICON GRAPHICS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Contents of zlib.txt file (from GraphicsMagick):

zlib 1.1.3 is a general purpose data compression library. All the code is thread safe.
The data format used by the zlib library is described by RFCs (Request for Comments) 1950 to 1952 in the files ftp://ds.internic.net/rfc/rfc1950.txt (zlib format), rfc1951.txt (deflate format) and rfc1952.txt (gzip format). These documents are also available in other formats from ftp://ftp.uu.net/graphics/png/documents/zlib/zdoc-index.html

All functions of the compression library are documented in the file zlib.h (volunteer to write man pages welcome, contact jloup@gzip.org). A usage example of the library is...
given in the file example.c which also tests that the library is working correctly. Another example is given in the file minigzip.c. The compression library itself is composed of all source files except example.c and minigzip.c.

To compile all files and run the test program, follow the instructions given at the top of Makefile. In short "make test; make install" should work for most machines. For Unix: "configure; make test; make install"

For MSDOS, use one of the special makefiles such as Makefile.msc.

For VMS, use Make_vms.com or descrip.mms.

Questions about zlib should be sent to <zlib@quest.jpl.nasa.gov>, or to Gilles Vollant <info@winimage.com> for the Windows DLL version.

The zlib home page is http://www.cdrom.com/pub/infozip/zlib/

The official zlib ftp site is ftp://ftp.cdrom.com/pub/infozip/zlib/

Before reporting a problem, please check those sites to verify that you have the latest version of zlib; otherwise get the latest version and check whether the problem still exists or not.

Mark Nelson <markn@tiny.com> wrote an article about zlib for the Jan. 1997 issue of Dr. Dobb's Journal; a copy of the article is available in http://web2.airmail.net/markn/articles/zlibtool/zlibtool.htm

The changes made in version 1.1.3 are documented in the file ChangeLog.

The main changes since 1.1.2 are:

- fix "an inflate input buffer bug that shows up on rare but persistent occasions" (Mark)
- fix gzread and gztell for concatenated .gz files (Didier Le Botlan)
- fix gzseek(..., SEEK_SET) in write mode
- fix crc check after a gzeek (Frank Faubert)
- fix miniunzip when the last entry in a zip file is itself a zip file (J Lilge)
- add contrib/asm586 and contrib/asm686 (Brian Raiter)
  See http://www.muppetlabs.com/~breadbox/software/assembly.html
- add support for Delphi 3 in contrib/delphi (Bob Dellaca)
- add support for C++Builder 3 and Delphi 3 in contrib/delphi2 (Davide Moretti)
- do not exit prematurely in untgz if 0 at start of block (Magnus Holmgren)
- use macro EXTERN instead of extern to support DLL for BeOS (Sander Stoks)
- added a FAQ file

plus many changes for portability.
Unsupported third party contributions are provided in directory "contrib". A Java implementation of zlib is available in the Java Development Kit 1.1
http://www.javasoft.com/products/JDK/1.1/docs/api/package-java.util.zip.html

See the zlib home page http://www.cdrom.com/pub/infozip/zlib/ for details.

A Perl interface to zlib written by Paul Marquess <pmarquess@bfssec.bt.co.uk> is in the CPAN (Comprehensive Perl Archive Network) sites, such as:

A Python interface to zlib written by A.M. Kuchling <amk@magnet.com> is available in Python 1.5 and later versions, see
http://www.python.org/doc/lib/module-zlib.html

A zlib binding for TCL written by Andreas Kupries <a.kupries@westend.com> is available at http://www.westend.com/~kupries/doc/trf/man/man.html

An experimental package to read and write files in .zip format, written on top of zlib by Gilles Vollant <info@winimage.com>, is available at http://www.winimage.com/zLibDll/unzip.html and also in the contrib/minizip directory of zlib.

Notes for some targets:

- To build a Windows DLL version, include in a DLL project zlib.def, zlib.rc and all .c files except example.c and minigzip.c; compile with -DZLIB_DLL

  The zlib DLL support was initially done by Alessandro Iacopetti and is now maintained by Gilles Vollant <info@winimage.com>. Check the zlib DLL home page at http://www.winimage.com/zLibDll

  From Visual Basic, you can call the DLL functions which do not take a structure as argument: compress, uncompress and all gz* functions.

  See contrib/visual-basic.txt for more information, or get
  http://www.tcfb.com/dowseware/cmp-z-it.zip

- For 64-bit Irix, deflate.c must be compiled without any optimization. With -O, one libpng test fails. The test works in 32 bit mode (with the -n32 compiler flag). The compiler bug has been reported to SGI.

- zlib doesn't work with gcc 2.6.3 on a DEC 3000/300LX under OSF/1 2.1 it works when compiled with cc.

- on Digital Unix 4.0D (formely OSF/1) on AlphaServer, the cc option -std1 is necessary to get gzprintf working correctly. This is done by configure.

- zlib doesn't work on HP-UX 9.05 with some versions of /bin/cc. It works with other compilers. Use "make test" to check your compiler.

- gzdopen is not supported on RISCOS, BEOS and by some Mac compilers.

- For Turbo C the small model is supported only with reduced performance to avoid any far allocation; it was tested with -DMAX_WBITS=11 -DMAX_MEM_LEVEL=3

- For PalmOs, see http://www.cs.uit.no/~perm/PASTA/pilot/software.html

Per Harald Myrvang <perm@stud.cs.uit.no>

Acknowledgments:
Appendix A: Third Party Acknowledgements

The deflate format used by zlib was defined by Phil Katz. The deflate and zlib specifications were written by L. Peter Deutsch. Thanks to all the people who reported problems and suggested various improvements in zlib; they are too numerous to cite here.

Copyright notice:

(C) 1995-1998 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.

3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly       Mark Adler
jloup@gzip.org         madler@alumni.caltech.edu

If you use the zlib library in a product, we would appreciate *not* receiving lengthy legal documents to sign. The sources are provided for free but without warranty of any kind. The library has been entirely written by Jean-loup Gailly and Mark Adler; it does not include third-party code.

If you redistribute modified sources, we would appreciate that you include in the file ChangeLog history information documenting your changes.

Progress OpenEdge v11.3 may incorporate International Classes for Unicode (International Components for Unicode) v2.4 from IBM. Such technology is subject to the following terms and conditions: ICU License - The ICU project is licensed under the X License (see also the x.org original), which is compatible with GPL but non-copyleft. The license allows ICU to be incorporated into a wide variety of software projects using the GPL license. The X license is compatible with the GPL, while also allowing ICU to be incorporated into non-open source products. License ICU License - ICU 1.8.1 and later COPYRIGHT AND PERMISSION NOTICE Copyright (c) 1995-2003 International Business Machines Corporation and others All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate Jaxen v1.1-beta-3 from jaxen.org. Such technology is subject to the following terms and conditions: Project License $Id: LICENSE.txt 1128 2006-02-05 21:49:04Z elharo $ Copyright 2003-2006 The Werken Company. All Rights Reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of the Jaxen Project nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate Jing 20030619 portion of Progress Extensions for Eclipse v2.2.1. Such technology is subject to the following terms and conditions: Jing Copying Conditions. Copyright (c) 2001-2003 Thai Open Source Software Center Ltd. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of the Thai Open Source Software Center Ltd nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE. Third-party JARs - This distribution includes some additional JAR files, which have their own copying conditions:

saxon.jar Comes from the Saxon 6.5.2 distribution and is covered by these conditions
xercesImpl.jar xml-apis.jar Come from the Xerces-J 2.4.0 distribution and are covered by the Apache Software License
isorelax.jar Comes from ISO RELAX 2003/01/08 distribution and is covered by the following license: Copyright (c) 2001-2002, SourceForge ISO-RELAX Project (ASAMI Tomoharu, Daisuke Okajima, Kohsuke Kawaguchi, and MURATA Makoto) Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Progress OpenEdge v11.3 may incorporate Sun's Jmaki framework v1.0 and Sun's Jmaki framework v1.0.3 from Sun Microsystems, Inc. Such technology is subject to the following terms and conditions: Copyright 1994-2006 Sun Microsystems, Inc. All Rights Reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
Redistribution of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistribution in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of Sun Microsystems, Inc. or the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission. This software is provided "AS IS," without a warranty of any kind. ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY EXCLUDED. SUN MICROSYSTEMS, INC. ("SUN") AND ITS LICENSORS SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THIS SOFTWARE OR ITS DERIVATIVES. IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE THIS SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. You acknowledge that this software is not designed, licensed or intended for use in the design, construction, operation or maintenance of any nuclear facility.

Progress OpenEdge v11.3 may incorporate Jmaki framework v1.8.0 from Sun Microsystems, Inc. Such technology is subject to the following terms and conditions:
Copyright 1994-2006 Sun Microsystems, Inc. All Rights Reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: * Redistribution of source code must retain the above copyright notice, this list of conditions and the following disclaimer. *
Redistribution in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of Sun Microsystems, Inc. or the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission. This software is provided "AS IS," without a warranty of any kind. ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY EXCLUDED. SUN MICROSYSTEMS, INC. ("SUN") AND ITS LICENSORS SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THIS SOFTWARE OR ITS DERIVATIVES. IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE THIS SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. You acknowledge that this software is not designed, licensed or intended for use in the design, construction, operation or maintenance of any nuclear facility.

Progress OpenEdge v11.3 may incorporate JSTL v1.0 from Sun Microsystems, Inc. Such technologies are subject to the following terms and conditions: Code sample License Copyright 1994-2006 Sun Microsystems, Inc. All Rights Reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistribution of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistribution in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of Sun Microsystems, Inc. or the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission. This software is provided "AS IS," without a warranty of any kind. ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY EXCLUDED. SUN MICROSYSTEMS, INC. ("SUN") AND ITS LICENSORS SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THIS SOFTWARE OR ITS DERIVATIVES. IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE THIS SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. You acknowledge that this software is not designed, licensed or intended for use in the design, construction, operation or maintenance of any nuclear facility.

Progress OpenEdge v11.3 may incorporate Looks v2.1.2 from JGoodies. Such technology is subject to the following terms and conditions: The BSD License for the JGoodies Looks - Copyright (c) 2001-2007 JGoodies Karsten Lentzhsc. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of JGoodies Karsten Lentzsch nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written
permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate OpenSSL v0.9.8 from The OpenSSL Project. Such technology is subject to the following terms and conditions: OpenSSL License. Copyright (c) 1998-2006 The OpenSSL Project. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. All advertising materials mentioning features or use of this software must display the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)." 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org. 5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project. 6. Redistributions of any form whatsoever must retain the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)." THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

OpenSSL includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com). Original SSLeay License. Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com) All rights reserved. This package is an SSL implementation written by Eric Young eay@cryptsoft.com). The implementation was written so as to conform with Netscapes SSL. This library is free for commercial and non-commercial use as long as the following conditions are aheared to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, Ihash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com). Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author.
of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. All advertising materials mentioning features or use of this software must display the following acknowledgement: "This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)." The word 'cryptographic' can be left out if the routines from the library being used are not cryptographic related :-). 4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement: "This product includes software written by Tim Hudson (tjh@cryptsoft.com)." THIS SOFTWARE IS PROVIDED BY ERIC YOUNG `AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. The licence and distribution terms for any publically available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence].

Progress OpenEdge v11.3 may incorporate Quartz Enterprise Job Scheduler v1.3.2 from James House. Such technologies are subject to the following terms and conditions: Copyright James House (c) 2001-2003 All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS `AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION). HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. This product uses and includes within its distribution, software developed by the Apache Software Foundation (http://www.apache.org/).

Progress OpenEdge v11.3 may incorporate relaxngDatatype.jar portion of Progress Extensions for Eclipse v2.2. Such technology is subject to the following terms and conditions: Copyright (c) 2001, Thai Open Source Software Center Ltd, Sun Microsystems. All rights reserved. Redistribution and use in source and binary forms,
with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the names of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate SLF4J v1.6.2 and SLF4J Log4J v1.6.1 from sfl4j.org. Such technology is subject to the following terms and conditions:
Copyright (c) 2004-2008 QOS.ch All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Progress OpenEdge v11.3 may incorporate STAX API (JSR 173) v3.1.1 from Woodstox Project. Such technology is subject to the following terms and conditions:
Copyright (c) 2004-2010, Woodstox Project (http://woodstox.codehaus.org/) All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the Woodstox XML Processor nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
Appendix A: Third Party Acknowledgements

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate Trang 20030619 portion of Progress Extensions for Eclipse v2.2.1. Such technology is subject to the following terms and conditions: Copyright (c) 2002, 2003 Thai Open Source Software Center Ltd. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of the Thai Open Source Software Center Ltd nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate xpp3-1.1.3.4.O portion of Progress Extensions for Eclipse v2.2.1. Such technology is subject to the following terms and conditions: Indiana University Extreme! Lab Software License Version 1.1.1 Copyright (c) 2002 Extreme! Lab, Indiana University. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

"This product includes software developed by the Indiana University Extreme! Lab (http://www.extreme.indiana.edu/)."
Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Indiana University" and "Indiana University Extreme! Lab" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact http://www.extreme.indiana.edu/.

5. Products derived from this software may not use "Indiana University" name nor may "Indiana University" appear in their name, without prior written permission of the Indiana University.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS, COPYRIGHT HOLDERS OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate Xpp3 v1.1.3.4 from Indiana University. Such technology is subject to the following terms and conditions: Indiana University Extreme! Lab Software License Version 1.1.1 Copyright (c) 2002 Extreme! Lab, Indiana University. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Indiana University Extreme! Lab (http://www.extreme.indiana.edu/)." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The names "Indiana University" and "Indiana University Extreme! Lab" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact http://www.extreme.indiana.edu/.

5. Products derived from this software may not use "Indiana University" name nor may "Indiana University" appear in their name, without prior written permission of the Indiana University. THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS, COPYRIGHT HOLDERS OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Progress OpenEdge v11.3 may incorporate XStream v1.3.1 from XStream. Such technology is subject to the following terms and conditions: Copyright (c) 2003-2006, Joe Walnes Copyright (c) 2006-2007, XStream Committers All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of XStream nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate YAJL v0.4.0 from Lloyd Hilaiel. Such technology is subject to the following terms and conditions: Copyright 2007, Lloyd Hilaiel. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. Neither the name of Lloyd Hilaiel nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate GNU ZLIB v1.1.4 and GNU ZLIB v1.2.3 from Jean-loup Gailly & Mark Adler. Such technology is subject to the following terms and conditions: Copyright notice: (C) 1995-2002 Jean-loup Gailly and Mark Adler This software is provided 'as-is', without any express or implied warranty. Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions: 1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required. 2. Altered source versions must be plainly marked as such, and must not be
misrepresented as being the original software. 3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly       Mark Adler
jloup@gzip.org          madler@alumni.caltech.edu

Progress OpenEdge v11.3 may incorporate ZLIB.NET Free v1.0.4 from ComponentAce. Such technology is subject to the following terms and conditions: Copyright (c) 2006-2007, ComponentAce http://www.componentace.com All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of ComponentAce nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Progress OpenEdge v11.3 may incorporate LALR Parser Generator in Java v0.10k. Such technology is subject to the following terms and conditions: CUP Parser Generator Copyright Notice, License, and Disclaimer – Copyright 1996-1999 by Scott Hudson, Frank Flannery, C. Scott Ananian - Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the names of the authors or their employers not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. The authors and their employers disclaim all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the authors or their employers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software.

Progress OpenEdge v11.3 includes the RSA Data Security, Inc. MD5 Message-Digest Algorithm. Copyright ©1991-2, RSA Data Security, Inc. Created 1991. All rights reserved. (MD5 Encryption Library v3.0 and MD5 Encryption vMD5C.C) These technologies are subject to the following terms and conditions: RSA Data Security MD5 message-digest algorithm RSA Data Security, Inc. MD5C.C - RSA Data Security, Inc., MD5 message-digest algorithm Copyright (C) 1991-2, RSA Data Security, Inc. Created 1991. All rights reserved. License to copy and use this software is granted provided that it is identified as the "RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing this software or this function. License is also granted to make and use derivative works provided that such works are identified as "derived from the RSA Data Security, Inc. MD5 Message-Digest
Algorithm\textsuperscript{\textregistered} in all material mentioning or referencing the derived work. RSA Data Security, Inc. makes no representations concerning either the merchantability of this software or the suitability of this software for any particular purpose. It is provided "as is" without express or implied warranty of any kind. These notices must be retained in any copies of any part of this documentation and/or software.
Index

A

Activate property 156
Admin role
   Web Services Adapter 201
AdminServer 22
After image buffers property 37
After image process property 37
After image stall property 37
After-image archival directory list property 38
After-image archival interval property 38
After-image Writer 54
Agent maximum port property 63
Agent minimum port property 62
Agent name property 62
Agent Shutdown Action property 65
AIW 54
Alternate buffer pool property 34
Apache Tomcat 284
Append to broker log file property
   AppServer broker 142
   AppServer messaging 161
   MS SQL Server DataServer 102
   ODBC DataServer 75
Oracle DataServer 88
   WebSpeed broker 235
   WebSpeed messaging 253
Append to log file property 198
Append to NameServer log file property 119
Append to server log file property
   AppServer messaging 162
   WebSpeed agent 245
   WebSpeed messaging 254
Application service names property
   AppServer 141
   WebSpeed 235
Application URL property 249
AppServer Internet Adapters
   actions allowed on 175
   configuration overview 175
   creating instances 183
   deleting instances 185
   editing a configuration 182
   enabling and disabling 185
   properties 176
      advanced features 182
      controlling NameServer 177
      general 177
      logging setting 179
      SSL 181
   viewing log files 183
AppServer Keepalive property 182, 268
AppServers
   actions allowed on 137
   additional management 174
agent properties 150
  advanced features 156
  general 151
  logging setting 152
  pool range 156
agents
  adding or trimming 166
broker properties 139
  advanced features 146
  AppService Name List 141
  controlling NameServer 140
  general 140
  logging setting 142
  owner information 140
configuration overview 136
creating instances 164
deleting instances 172, 173
editing a configuration 163
environment variables 162
features 137
messaging properties 160, 161
property storage 138
SSL properties 158
  advanced features 159
  general 158
starting
  automatically 165
  manually 164
status 167
viewing log files 173
APW 54
Assigning database default
configurations 47
Asynchronous Page Writer 54
Asynchronous page writers property 36
Authorized IP list property 229
Auto start property
  AppServer 140
  MS SQL Server DataServer 101
  NameServer 118
  ODBC DataServer 74
  OpenEdge database 31
  Oracle DataServer 87
  WebSpeed Transaction Server 233
Auto trim timeout property
  AppServer 147
  WebSpeed Transaction Server 240
Automatically add After imaging extents
property 69
Backup method property 69
Base index number property 44
Base table number property 44
Before image buffers property 36
Before image cluster age property 37
Before image delay writers property 37
Before image process property 36
Before image stall property 36
Before image threshold property 37
Before-image Writer 54
Begin After imaging after transition
property 69
Binary upload max size property 249
BIW 54
Blocks in database buffers property 34
Broker keep alive timeout property 118
Broker log file threshold size property
  AppServer 145
  MS SQL Server DataServer 102
  ODBC DataServer 75
  Oracle DataServer 89
  WebSpeed Transaction Server 238
Broker log filename property
  AppServer 142
  MS SQL Server DataServer 102
  ODBC DataServer 75
  Oracle DataServer 88
  WebSpeed Transaction Server 235
Broker logging entry types property
  AppServer 143
  MS SQL Server DataServer 102
  ODBC DataServer 75
  Oracle DataServer 88
  WebSpeed Transaction Server 236
Broker logging level property
  AppServer broker 142
  AppServer messaging 161
  MS SQL Server DataServer 102
  ODBC DataServer 75
  Oracle DataServer 88
  WebSpeed broker 235
  WebSpeed messaging 253

B
  Background writers 54
Index

Changing
AppServer
agent properties 150
broker properties 139
environment variables 162
messaging properties 160
SSL properties 158
AppServer Internet Adapter
properties 176
MS SQL Server DataServer
broker properties 100
environment variables 105
server properties 104
NameServer
environment variables 124
properties 116
ODBC DataServer
broker properties 73
environment variables 78
server properties 77
OpenEdge database
basic properties 31
default properties 33
server group properties 48
Oracle DataServer
broker properties 86
environment variables 92
server properties 90
SonicMQ Adapter
broker properties 264
environment variables 274
server properties 269
SSL properties 271
Web Services Adapter
advanced features properties 202
location properties 195, 290
logging setting properties 197
properties 194, 290
proxy server setting properties 196, 291
run-time properties 214, 297
security properties 200
WSDL properties 196
WebSpeed Messenger
properties 225
WebSpeed Transaction Server
agent properties 243
broker properties 233
environment variables 255
messaging properties 252
SSL properties 250
Changing OE Web Server run-time properties 297
Client Type property 50
Cluster enabled property 32
Collation table property 38
Configuration group 28
Configuring
AppServer components 135
AppServer Internet Adapters 175
AppServers 136
DataServers 71
MS SQL Server DataServers 99
NameServers 113
ODBC DataServers 72
OpenEdge databases 27
OpenEdge Web Services 187
Oracle DataServers 85
SonicMQ Adapters 261, 283
Web service instances 216, 300
Web Services Adapter instances 191
WebSpeed components 223
WebSpeed Messengers 224
WebSpeed Transaction Servers 232
conmgr.properties file 28
Connect property 156
Connection Timeout property
replication
agent 62
control agent 66
Controlling NameServer property
AppServer 140
MS SQL Server DataServer 101
ODBC DataServer 74
Oracle DataServer 87
WebSpeed Messenger 225
WebSpeed Transaction Server 234
Conversion map property 38
CPU statistic 57, 172
Create after-image archival directories property 38
Creating
AppServer instances 164
AppServer Internet Adapter instances 183
MS SQL Server DataServer instances 107
NameServer instances 126
ODBC DataServer instances 79
OpenEdge database configurations 46
server groups 52
Oracle DataServer instances 93
SonicMQ Adapter instances 276
Web Services Adapter instances 192
WebSpeed Transaction Server instances 256
Critical agent property 66
Index

D

Data service name list property
  MS SQL Server DataServer 102
  ODBC DataServer 75
  Oracle DataServer 88

Database group 28

Database name property 31

Database property
  replication
    agent 62
    control agent 66
    server 64

Database Role property 67

Database service communication area size property 40

DataServers 71
  See also ODBC DataServers, Oracle DataServers, MS SQL Server DataServers

Deactivate property 156

Debug clients property 203

Debug mode property 250

Default configuration property 32

Default cookie domain property 249

Default cookie path property 249

default.props file 206

Defer Agent Startup property 64

Deleting
  AppServer instances 172, 173
  AppServer Internet Adapter instances 185
  MS SQL Server DataServer instances 107
  NameServer instances 127
  ODBC DataServer instances 80
  OpenEdge database configurations 58
  references 58
  server groups 59
  Oracle DataServer instances 94
  SonicMQ Adapter instances 277
  Web Services Adapter instances 215, 298
  WebSpeed Transaction Server instances 259

Deploying a REST application (.war file) 294

Direct I/O property 35

Disable client verification of SSL host name property 228

Disable SSL session cache property 45

Disable SSL session caching property 159, 228, 251

Disable SSL session reuse property 228

Disconnect property 156

E

Editing
  AppServer Internet Adapter properties 182
  AppServer properties 163
  MS SQL Server DataServer properties 106
  NameServer properties 124
  ODBC DataServer properties 79
  OpenEdge database default properties 46
  server group properties 51
  Oracle DataServer properties 93
  SonicMQ Adapter properties 275
  Web Services Adapter properties 204
  WebSpeed Messenger properties 229
  WebSpeed Transaction Server properties 256

Enable dynamic property updates property 268
  WebSpeed Transaction Server 241

Enable SSL client connections property
  AppServer 158
  WebSpeed Messenger 228
  WebSpeed Transaction Server 251

Enable SSL for remote connections property 45

Enable Web Services property 200

Enable WSA administration property 200

Enable WSDL listing retrieval property 197

Enable WSDL retrieval property 196

Environment property 122

Execution Time Limit property 250
F
Fault tolerance 128
  connection-level 128
  server-level 128
File upload directory property 249
4GL debugger enabled property
  AppServer 156
  WebSpeed Transaction Server 250
Full on-line and off-line backup
  arguments property 69

G
Getting started 24
Group name property
  AppServer 140
  MS SQL Server DataServer 101
  ODBC DataServer 74
  Oracle DataServer 87
  WebSpeed Transaction Server 233

H
Hash table entries property 34
Host name property 117
Host property
  database server group 51
  replication control agent 66
HTTP error page property 203

I
Incremental backup arguments property 69
Index range size property 44
Initial number of servers to start property
  AppServer 156
  WebSpeed Transaction Server 248
Internal administration command
  property 229

J
Java classpath property 44
jvmargs property
  AppServer 147
  WebSpeed Transaction Server 240

K
Keep Alive property 64
Kernel time statistic 57, 172
Key alias name property 45
Key alias password property 45

L
Load balancing 128
  priority weight 129
Location property
  NameServer 117
  Web Services Adapter 195, 290
Location transparency 129
Lock table entries property 34
Lock table hash table size property 40
Log code page property 39
Log file threshold size property
  NameServer 120
  Web Services Adapter 199
  WebSpeed Messenger 228
Log filename property 197
  WebSpeed Messenger 227
Log files
  AppServer 173
  AppServer Internet Adapter 183
  MS SQL Server DataServer 111
  NameServer 134
  ODBC DataServer 83
  OpenEdge database 29
  Oracle DataServer 97
  SonicMQ Adapter 281
  WebSpeed Messenger 230
Log message threshold size property
  198
Logging entry types property
  NameServer 120
  Web Services Adapter 199
  WebSpeed Messenger 227
Logging level property
  Web Services Adapter 198
  WebSpeed Messenger 227
Login
  Web server 194
M

Managed databases 28
Managing database instances 30
Managing Web Services Adapter instances 283
Max servers property 35
Max users property 34
Maximum client instances property
  AppServer 146
  MS SQL Server DataServer 103
  ODBC DataServer 76
  Oracle DataServer 89
  WebSpeed Transaction Server 239
Maximum clients per server property 50
Maximum dynamic port property 50
Maximum message property 66
Maximum NameServer client port property 226
Maximum number of broker log files property
  AppServer 145
  MS SQL Server DataServer 103
  ODBC DataServer 76
  Oracle DataServer 89
  WebSpeed Transaction Server 238
Maximum number of database areas property 40
Maximum number of JTA transactions property 44
Maximum number of log files property
  NameServer 120
  Web Services Adapter 199
  WebSpeed Messenger 228
Maximum number of server log files property
  AppServer 155
  WebSpeed Transaction Server 248
Maximum port number property
  AppServer 152
  MS SQL Server DataServer 105
  ODBC DataServer 78
  Oracle DataServer 91
  WebSpeed Transaction Server 244
Maximum private database buffers per user property 35
Maximum servers property

AppServer 156
  WebSpeed Transaction Server 248
Maximum shared-memory segment size property 42
Message buffer size property 49
Messenger executable filename property 225
Minimum clients per server property 50
Minimum dynamic port property 51
Minimum NameServer client port property 226
Minimum port number property
  AppServer 151
  MS SQL Server DataServer 105
  ODBC DataServer 78
  Oracle DataServer 91
  WebSpeed Transaction Server 244
Minimum servers property
  AppServer 156
  WebSpeed Transaction Server 248
Modifying
  See Changing
Monitored property 45

MS SQL Server DataServers actions allowed on 99
  broker properties 100
    advanced features 103
    controlling NameServer 101
    data service 102
    general 100
    logging setting 102
    owner information 101
  configuration overview 99
  creating instances 107
  deleting instances 107
  editing a configuration 106
  environment variables 105
  server properties 104
    general 104
    logging setting 105
    starting
      automatically 108
      manually 108
    status 109
    viewing log files 111

N

NameServer logging level property 119
NameServers actions allowed on 115
configuration overview 114
configuring local instances 116
configuring remote instances 123
creating instances 126
deleting instances 127
editing a configuration 124
environment variables
  setting and deleting 124
features 114
  fault tolerance 128
  load balancing 128
  location transparency 129
firewalls and UDP 133
local and remote 115
properties
  advanced features 121
  environment variables 122
  general 117
  location 117
  logging setting 118
starting
  automatically 130
  manually 130
status 131
viewing log files 134
Nap interval property 40
Nap max interval property 41
Nap time increment property 41
Nap time steps between nap time property 41
Neighboring NameServers property 121
Normal startup arguments property 68
Number of open SQL cursors property 44
Number of servers property 49
O
ODBC DataServers
  actions allowed on 72
  broker properties 73
    advanced features 76
    controlling NameServer 74
    data service 75
    general 73
    logging setting 75
    owner information 74
    configuration overview 72
    creating instances 79
    deleting instances 80
    editing a configuration 79
  environment variables 78
  server properties 77
    general 77
logging setting 78
starting
  automatically 80
  manually 80
status 81
viewing log files 83
OE Web Server 284
  Configuring and managing 284
OE Web Servers
  Managing 287
OpenEdge databases
  actions allowed on 29
  assigning a default configuration 47
  background writers 54
  basic properties 31
  configuration overview 28
  configuring replication-enabled 60
  See also Replication-enabled databases
  conmgr.properties file 28
  creating a configuration 46
  creating a server group 52
  default configuration
    editing 46
  default properties 33
  deleting references 58
  managed 28
  managing instances 30
  process details 56, 171
  property settings
    awareness 29
    basic 31
    default 33
    scripted 28
  server group properties 48
    editing 51
  shared memory systems 54
starting
  automatically 53
  manually 53
status 55
stopping 53
viewing log files 29
watchdog process 54
OpenEdge Explorer
  comparing OpenEdge Management 24
  introduction 21
  overview 23
OpenEdge Management
  comparing OpenEdge Explorer 24
  introduction 21
  overview 22
OpenEdge Web Services
  application server 188
  configuration overview 188, 284
  default property settings 206, 293
default.props file 206
deploying to a WSA 207, 294
enabling or disabling for access 216, 300
exporting from a WSA 209
importing to a WSA 210
listing deployed to a WSA 211
login 194
managing 216, 300
managing with the WSA 189, 285
run-time statistics 218, 301
session models 189
status of deployed Web service 217
undeploying 222, 306
updating 219, 304
definition and deployment information 219
property settings 220, 304

Operating mode property
AppServer 140
MS SQL Server DataServer 100
ODBC DataServer 73
Oracle DataServer 86
WebSpeed Transaction Server 233

Operator role
Web Services Adapter 201

Oracle DataServers
actions allowed on 85
broker properties 86
advanced features 89
controlling NameServer 87
data service 88
genround 86
logging setting 88
owner information 87
configuration overview 85
creating instances 93
deleting instances 94
editing a configuration 93
environment variables 92
server properties 90
genround 91
logging setting 92
starting
automatically 94
manually 94
status 95
viewing log files 97

Other arguments property 35
replication
agent 63
core agent 66
server 64

Page writer queue delay property 41
Page writer queue minimum property 41
Page writer scan delay property 41
Page writer scan property 41

Password confirm property
AppServer broker 140
AppServer SSL 159
WebSpeed broker 234

Password property
AppServer 140
MS SQL Server DataServer 101
ODBC DataServer 74
Oracle DataServer 87
WebSpeed Transaction Server 234

Password to access the key/certificate
Confirm property
WebSpeed SSL 251

Password to access the key/certificate property
AppServer 159
WebSpeed Transaction Server 251

Pending connection timeout property 41

Pin shared memory segments property 42

Polling delay maximum property
replication
agent 62
server 65

Polling delay minimum property
replication
agent 62
server 65

Port number property
AppServer 140
MS SQL Server DataServer 101
NameServer 117
ODBC DataServer 74
Oracle DataServer 87
WebSpeed Transaction Server 233

Port property 66

Priority weight property
AppServer 146
MS SQL Server DataServer 103
ODBC DataServer 76
Oracle DataServer 90
WebSpeed Transaction Server 239

Private key/Digital certificate alias name property
AppServer 158
WebSpeed Transaction Server 251
Process details
OpenEdge database 56, 171
Process time statistic 57, 172

PROPATH property
AppServer 151
MS SQL Server DataServer 105
ODBC DataServer 77
Oracle DataServer 91
WebSpeed Transaction Server 244

Properties
AppServer
agent 150
broker 139
messaging 160
SSL 158
AppServer Internet Adapter 176
MS SQL Server DataServer
broker 100
server 104
ODBC DataServer
broker 73
server 77
OpenEdge database
basic 31
commgr.properties file 28
default 33
server group 48
setting awareness 29
Oracle DataServer
broker 86
server 90
SonicMQ Adapter
broker 264
server 269
SSL 271
Web Services Adapter 194, 290
WebSpeed Messenger 225
WebSpeed Transaction Server
agent 243
broker 233
messaging 252
SSL 250

Proxy host property 196, 291
Proxy password Confirm property 196
Proxy password property 196, 291
Proxy port property 196, 291
Proxy username property 196, 291

R
Register with NameServer property
AppServer 140

MS SQL Server DataServer 101
ODBC DataServer 74
Oracle DataServer 87
WebSpeed Messenger 225
WebSpeed Transaction Server 234

Registration host name property
AppServer 141
MS SQL Server DataServer 101
ODBC DataServer 74
Oracle DataServer 87
WebSpeed Transaction Server 234

Registration mode property
AppServer 141
MS SQL Server DataServer 101
ODBC DataServer 74
Oracle DataServer 87
WebSpeed Transaction Server 234

Registration retry property
AppServer 146
MS SQL Server DataServer 103
ODBC DataServer 76
Oracle DataServer 90
WebSpeed Transaction Server 239

Replication method property 66

Replication-enabled databases 60
property settings
agent 61
categories 61
control agent 65
rules 60
server 63
transition 67

Reporting interval property 49

Request timeout property
AppServer 147
WebSpeed Transaction Server 240

Require Web Services authorization property 201

Require WSA administration authorization property 200

Require WSDL Retrieval authorization property 201

Resident size statistic 57, 172

Responsibility property 67

REST application
Undeploying 306
Updating a deployed REST application 304

REST applications
SSL support 305
Index

Restart after transition property 68
Roles
  Web Services Adapter 201

Server Transition Timeout property 64
Server watch dog interval property
  AppServer 155
  WebSpeed Transaction Server 248
serverASK Activity Timeout property 269
serverASK Response Timeout property 269
ServerGroup group 28
Service name or port property 49
Session connection ID property 229
Session-Free
  AppServers 136
  Web services 190
Session-Managed
  AppServers 136
  Web services 190
Shared memory overflow size property 42
Shared memory systems 54
Shutdown property 157
SOAP-AVM gateway 188
SonicMQ Adapters
  actions allowed on 263
  broker properties 264
    advanced features 268
    AppService Name List 266
    controlling NameServer 266
    general 265
    logging setting 267
    owner information 265
    configuration overview 262
    creating instances 276
    deleting instances 277
    editing a configuration 275
    enabling or disabling for access 274
    environment variables 274
    server properties 269
      general 270
      logging setting 270
      pool range 271
    SSL properties 271
      advanced features 273
      general 272
    starting
      manually 278
    status 279
    viewing log files 281
SonicMQ ServerConnect broker log filename property
AppServer 161
WebSpeed Transaction Server 252

SonicMQ ServerConnect enabled property
AppServer 161
WebSpeed Transaction Server 252

SonicMQ ServerConnect server log filename property
AppServer 161
WebSpeed Transaction Server 253

Source startup arguments property 68
source.repl.properties file 60
Spin lock retries property 43
SQL sorting memory property 44
SQL sorting on disk 44
SQL stack size property 44
SQL statement cache size property 44
SQL temporary data page size property 44
SSL session cache timeout property 228
AppServer 159
WebSpeed Transaction Server 251

Starting
AppServers
automatically 165
manually 164
MS SQL Server DataServers
automatically 108
manually 108
NameServers
automatically 130
manually 130
ODBC DataServers
automatically 80
manually 80
OpenEdge databases
automatically 53
manually 53
Oracle DataServers
automatically 94
manually 94
SonicMQ Adapters
manually 278
Web Services Adapters 205
WebSpeed Transaction Servers
automatically 257
manually 257

Startup parameters property
AppServer 157

Status
AppServer 167
deployed Web Services 217
MS SQL Server DataServer 109
NameServer 131
ODBC DataServer 81
OpenEdge database 55
Oracle DataServer 95
SonicMQ Adapter 279
Web Services Adapter 211
run-time 212, 296

Stopping
OpenEdge databases 53
Web Services Adapters 214

Storage object cache size property 43
Structure file containing After image area definitions property 69

Supports default service property
AppServer 141
MS SQL Server DataServer 102
ODBC DataServer 75
Oracle DataServer 88
WebSpeed Transaction Server 235

Synchronous or Asynchronous property 66

T
Table range size property 44
Target startup arguments property 68
target.repl.properties file 60
TCP/IP version property
AppServer 147
NameServer 121
OpenEdge database 50
WebSpeed Transaction Server 240

To enable administration roles property 200

Transition to Agents property 68
TXE commit lock skip limit property 43

U
ubroker.properties file 24
Unified Broker products 114

Updating
See Changing
URL property 195, 290
User MUX latches property 43
User time statistic 57, 172
Username property
  AppServer 140
  MS SQL Server DataServer 101
  ODBC DataServer 74
  Oracle DataServer 87
  WebSpeed Transaction Server 233

V

Viewing
  AppServer
    agent properties 150
    broker properties 139
    environment variables 162
    log files 173
    messaging properties 160
    SSL properties 158
    status 167
  AppServer Internet Adapter
    log files 183
  AppServer Internet Adapter properties 176
deployed Web service
  run-time statistics 218, 301
  statistics 213
  status 217
MS SQL Server DataServer
  broker properties 100
  environment variables 105
  log files 111
  server properties 104
  status 109
NameServer
  environment variables 124
  log files 134
  properties 116
  status 131
ODBC DataServer
  broker properties 73
  environment variables 78
  log files 83
  server properties 77
  status 81
OpenEdge database
  basic properties 31
  default properties 33
  log files 29
  process details 56, 171
  server group properties 48
  status 55
Oracle DataServer
  broker properties 86
  environment variables 92
  log files 97
  server properties 90

status 95
SonicMQ Adapter
  broker properties 264
  environment variables 274
  log files 281
  server properties 269
  SSL properties 271
  status 279
Web Services Adapter
  advanced features properties 202
  location properties 195, 290
  logging setting properties 197
  properties 194, 290
  proxy server setting properties 196, 291
  security properties 200
  status 211
  run-time 212, 296
  WSDL properties 196
WebSpeed Messenger
  log files 230
  properties 225
WebSpeed Transaction Server
  agent properties 243
  broker properties 233
  environment variables 255
  messaging properties 252
  SSL properties 250

Viewing OE Web Server instance
  statistics 296
Viewing the status of an OE Web Server instance 295
Virtual size statistic 57, 172

W

Watchdog process 54
Watchdog process property 36
Web services
  See OpenEdge Web Services
Web Services Adapter
  Admin or Operator role 201
Web Services Adapters
  actions allowed on 191
  changing run-time properties 214, 297
  configuration overview 193
  creating instances 192
  deleting instances 215, 298
  deploying Web services 207, 294
  description 188, 284
  editing a configuration 204
  enabling or disabling Web services 216, 300
  exporting Web services 209
importing Web services 210
listing deployed Web services 211
properties 194, 290
  advanced features 202
location 195, 290
logging setting 197
proxy server setting 196, 291
security 200
WSDL 196
setting defaults for Web services 206, 293
starting 205
status 211
  run-time 212, 296
stopping 214
undeploying Web services 222, 306
updating Web services 219, 304
viewing Web service statistics 213
Web service run-time statistics 218, 301
Web Services Protocol Value property 195
WebSpeed
  additional management 260
WebSpeed Messengers
  configuration overview 224
  editing a configuration 229
  properties 225
    advanced features 229
    controlling NameServer 225
    general 225
    logging setting 227
    SSL 228
    WebSpeed Transaction Server Location 226
    viewing log files 230
WebSpeed transaction server host property 226
WebSpeed transaction server port property 226
WebSpeed Transaction Servers
  actions allowed on 232
  additional management 260
  agent properties 243
    advanced features 249
    general 243
logging setting 245
  pool range 248
agents
  adding or trimming 257
broker properties 233
  advanced features 239
AppService Name List 235
controlling NameServer 234
general 233
logging setting 235
  owner information 233
configuration overview 232
  creating instances 256
  deleting instances 259
editing a configuration 256
environment variables 255
messaging properties 252
SSL properties 250
  advanced features 251
  general 251
starting
  automatically 257
  manually 257
Weighted CPU statistic 57, 172
Windows Event Level property 40
Working directory property
  AppServer 140
  MS SQL Server DataServer 100
  NameServer 117
  ODBC DataServer 73
  Oracle DataServer 86
  Web Services Adapter 197
  WebSpeed Messenger 225
  WebSpeed Transaction Server 233
Working with REST applications 283
WSA
  See Web Services Adapter
WSA administration authentication property 200
WSA administration SOAP action property 202
WSDL inaccessible page property 196
WSDL listing page property 197