

PROGRESS[®] OPENEDGE[®]

OpenEdge. Getting Started:
Installation and Configuration

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

Apama, Business Empowerment, Business Making Progress, Corticon, Corticon (and design), DataDirect (and design), DataDirect Connect, DataDirect Connect64, DataDirect XML Converters, DataDirect XQuery, Empowerment Center, Fathom, Making Software Work Together, OpenEdge, Powered by Progress, PowerTier, Progress, Progress Control Tower, Progress Dynamics, Progress Business Empowerment, Progress Empowerment Center, Progress Empowerment Program, Progress OpenEdge, Progress Profiles, Progress Results, Progress RPM, Progress Software Business Making Progress, Progress Software Developers Network, ProVision, PS Select, RulesCloud, RulesWorld, SequeLink, SpeedScript, Stylus Studio, Technical Empowerment, WebSpeed, Xcalia (and design), and Your Software, Our Technology—Experience the Connection are registered trademarks of Progress Software Corporation or one of its affiliates or subsidiaries in the U.S. and/or other countries. AccelEvent, Apama Dashboard Studio, Apama Event Manager, Apama Event Modeler, Apama Event Store, Apama Risk Firewall, AppsAlive, AppServer, BusinessEdge, Cache-Forward, DataDirect Cloud, DataDirect Spy, DataDirect SupportLink, Future Proof, High Performance Integration, OpenAccess, ProDataSet, Progress Arcade, Progress ESP Event Manager, Progress ESP Event Modeler, Progress Event Engine, Progress RFID, Progress Responsive Process Management, Progress Software, PSE Pro, SectorAlliance, SeeThinkAct, SmartBrowser, SmartComponent, SmartDataBrowser, SmartDataObjects, SmartDataView, SmartDialog, SmartFolder, SmartFrame, SmartObjects, SmartPanel, SmartQuery, SmartViewer, SmartWindow, WebClient, and Who Makes Progress are trademarks or service marks of Progress Software Corporation and/or its subsidiaries or affiliates in the U.S. and other countries. Java is a registered trademark of Oracle and/or its affiliates. Any other marks contained herein may be trademarks of their respective owners.

Third party acknowledgements — See the table of contents for the "Third Party Acknowledgements" appendix.



July 2013

Last updated with new content: Release 11.3.0

Product Code: 4496; R11.3.0

For the latest documentation updates see [OpenEdge Product Documentation](http://communities.progress.com/pcom/docs/DOC-16074) on PSDN (<http://communities.progress.com/pcom/docs/DOC-16074>).

Contents

Preface	17
 Part 1 Installation	
1. Windows Installation Requirements	31
System requirements	32
Ensuring you have the most up-to-date system requirements information	32
Java considerations	32
Windows system requirements	34
Supported platforms	37
Supported products by platform	38
Disk space requirements	39
Server compatibility	43
OpenEdge clients	43
General connectivity and compatibility rules	44
OpenEdge SQL	45
Deployment rules supported by a DataServer broker	45
Development rules related to schema holder compatibility	45
Apache Tomcat	45
Required third-party applications	46
Microsoft .NET Framework	46
Infragistics NetAdvantage	47
DataDirect ODBC branded drivers	47
Licensing	51
 2. UNIX Systems Installation Requirements	 53
UNIX system requirements	54
Requirements for using Java	54
Requirements for running OpenEdge applications	59
Supported platforms	60
Supported products by platform	61
Disk space requirements	63

Licensing	65
3. OpenEdge Installation Prerequisites	67
Tasks overview	68
Gathering information to plan your installation	69
Determining your installation method	70
Determining the type of installation	71
How product selection can affect your installation tasks.	71
Obtaining an Electronic License Addendum file	72
Shared Network Installation utility	74
Windows-specific installation considerations	75
OpenEdge working directory reminder	75
Read-only .dll and .ocx files	75
Required software to run OpenEdge products or components.	75
Saving an existing OpenEdge or Progress installation in Windows	76
Reviewing the Windows installation directory structure	79
Reviewing the Eclipse installation directory structure	81
Integrating OpenEdge with Windows Explorer	81
UNIX-specific installation considerations	84
JDK and JRE considerations	84
Upgrade an existing OpenEdge or Progress installation on UNIX platforms	84
Review the UNIX system installation directory structure	85
OpenEdge Replication	88
Installing OpenEdge Replication for the first time	88
Upgrading an existing version of OpenEdge Replication	89
OpenEdge Management or OpenEdge Explorer	90
Installing OpenEdge Management or Explorer for the first time	90
System requirements	91
Support for multiple Eclipse frameworks	92
Integration after installation.	92
Service pack updates to plug-ins	92
Uninstall	92
WebSpeed configuration choices	94
Developing Web applications with WebSpeed	94
Deploying Web applications with WebSpeed	94
Accessing product documentation and samples	95
Accessing documentation and samples from Progress Download Center	95
4. Performing an OpenEdge Installation in Windows.	97
Installation overview	98
Loading the installation media	98
Performing the installation	98
Finishing the installation	99
Post Installation Considerations	100
Running the Progress Dynamics Configuration Utility	101
Before you begin.	101
Completing the DCU wizard	102
Editing Progress Dynamics files	107
Editing installed files	108
Editing the Progress Dynamics XML configuration file	109
Starting a development session	111
Stopping and restarting Progress Dynamics.	112
Updating session types.	112
Running the Entity Import tool.	114
Recompiling application code	115
Setting up for Web development.	115

Additional product installation activities	116
Using an Electronic License Addendum file	116
Installing additional products	117
Installing additional components to previously installed products	118
Viewing registry information	120
Downloading executables for heterogeneous environments	121
Configuring an Apache Tomcat Java Servlet Engine	121
OpenEdge Silent installation overview	123
Selecting a data input option for a Silent installation	124
Understanding the response.ini file contents	124
Running the Silent installation	133
Checking the status of the Silent installation log file	134
Optional data input activities	136
Performing postinstallation tasks	137
Uninstalling OpenEdge in Windows	138
Using the Uninstall or Add/Remove Programs utility	138
Manually removing OpenEdge	140
Sharing an OpenEdge installation on a network overview	145
Primary tasks	145
Networking overview	146
Determining a shared network to clients connection	146
Setting up the shared network	147
Running Shared Network Installation Utility to setup a client connection	147
Reviewing local intranet security settings	150
Uninstalling the Shared Network Installation Utility	151
Running the Silent installation option for the Shared Network Installation Utility	152
Creating a user-defined response file	152
Executing NetSetup with the Silent installation option	154
5. Performing an OpenEdge Installation on UNIX or Linux	155
Installation overview	156
Starting the installation process	156
Performing the installation	157
Finishing the installation	158
Additional product installation activities	159
Using an electronic license addendum file	159
Installing additional products	160
Adding components to previously installed products	161
Downloading executables for heterogeneous environments	164
OpenEdge Silent installation overview	165
Data input options for a Silent installation	166
Understanding the response.ini file contents	166
Running the Silent installation	172
Checking the status of the Silent Installation log file	173
Optional data input activities	174
Performing postinstallation tasks	175
Setting AdminServer security	175
Performing a rolling upgrade of OpenEdge Management	177
Making port updates	177
Installing a new console	178
Upgrading a remote container	179
Uninstalling OpenEdge on UNIX and Linux operating systems	181
Uninstalling OpenEdge Replication	182
Manually removing earlier OpenEdge versions	182

6. Administration Utilities	183
Using the License Update utility	184
Changes to accommodate license updates	184
Displaying license information using the SHOWCFG utility	186
Using the SHOWCFG utility in Windows	186
Using the SHOWCFG utility on UNIX or Linux platforms	188
Displaying license information in Windows	189
Managing user licenses on all supported platforms	190
Removing license for installed product	191
Removing license from Windows platform	191
Removing license from UNIX platform	191
OpenEdge license information	192
Using the OpenEdge license file	192
Using OpenEdge resources in Windows	195
Shared memory	195
Processes on Windows and UNIX platforms	195
Manage memory and system configurations on UNIX platforms	196
Calculating memory needs	196
Managing shared memory and process resources	199
Reducing memory usage	200
Swap space	200
Shared memory and kernel configuration	200
UNIX troubleshooting tips	203
Error messages	203
Altered or missing progress.cfg file	203
Tailoring startup scripts	204
OpenEdge event logging	206
OpenEdge event log file	206
Managing the OpenEdge event log file size	206
Event logging in Windows	208

Part 2 Configuration

7. Working in the OpenEdge Environment in Windows	215
Reviewing environment variables	216
System environment variables	216
Java environment variables	216
Windows registry and the progress.ini file	218
Environment variables	218
Additional details for Java-related environment variables	222
Setting OpenEdge Program Item properties	223
Using the Proenv utility	224
Getting started with the AdminServer	225
OpenEdge products supported by the AdminServer	226
AdminServer considerations	226
AdminServer group name conventions and restrictions	227
Creating and configuring an OpenEdge database server	228
Running OpenEdge	229
Maintaining OpenEdge and Progress	230
OpenEdge key and certificate stores	231
Support for IPv6	232
Specifying IPv6	232
Windows 64-bit	234
Notes for the 64-bit GUI client	234
Application development and deployment	235

Product and database interactions	237
8. Working in the OpenEdge Environment on UNIX.	239
Default environment variables settings	240
UNIX environment variables	241
Setting Java environment variables	246
Setting the JDK environment variable	246
Setting SQL client environment variables	247
Using the Proenv utility	248
Getting started with the AdminServer	249
OpenEdge products supported by the AdminServer	249
AdminServer considerations	250
How to implement the User-Group Authorization feature.	250
Understanding the built-in terminal definitions	251
Terminal issues	251
Terminal identifiers	251
Additional terminal identifier considerations	253
OpenEdge key and certificate stores	254
9. Managing OpenEdge Key and Certificate Stores	255
Managing key stores for OpenEdge servers	256
Establishing a trusted SSL server identity	256
Using pkiutil to manage an OpenEdge key store	257
Understanding key store content	258
Using genpassword to obtain a key store password-encrypted value	260
Managing certificate stores for OpenEdge clients and servers	261
Installing trusted CA/root certificates	261
Using certutil to manage an OpenEdge root certificate store	263
Using mkhashfile to install root certificates in the OpenEdge root certificate store	264
10. Configuration	265
Introducing OpenEdge Management and OpenEdge Explorer	266
Overview of OpenEdge Management or OpenEdge Explorer	268
Introduction.	268
OEM or OEE elements and descriptions	269
Additional OEM or OEE considerations	274
Configure OpenEdge Explorer post installation	275
Working with Unified Brokers	276
Running locally	276
Running remotely	276
Unified Broker common elements	276
Using default sample brokers.	277
Configuring and starting Unified Broker instances	278
Understanding and using the AdminServer	282
Starting the AdminServer	283
Stopping the AdminServer	283
Changing the default port	284
Changing the startup setting	285
Running more than one AdminServer	285
Querying the AdminServer.	286
AdminServer-related authorization option	287
Using OpenEdge Management or OpenEdge Explorer	288
OpenEdge Servers supported by OEM or OEE.	288
Saving configurations	290

Mergeprop utility overview	291
Operating interfaces	291
Property value	291
Using the mergeprop utility	292
Mergeprop parameter details	293
Mergeprop examples	296
Java API details	299
Logical structure and syntax of property files	301
Ubroker.properties file and product configurations	304
Unified Broker products and associated clients	305
Unified Broker installation prerequisites	305
Ubroker.properties file structure	306
Specifying IP version for underlying Java code	309
Command-line utilities reference	311
11. Starting and Running OpenEdge	313
Starting OpenEdge in Windows	314
Startup and shutdown	314
Starting OpenEdge as a Windows service	317
Starting single-user OpenEdge in interactive mode	318
Starting single-user OpenEdge in batch or background mode	318
Starting the multi-user server or broker	319
Starting the multi-user server or broker as a Windows service	319
Starting OpenEdge on UNIX platforms	321
Startup and shutdown commands	321
Starting single-user OpenEdge in interactive mode	323
Starting single-user OpenEdge in batch or background mode	323
Starting the multi-user server or broker	324
Running OpenEdge clients and servers on a network	325
Using network startup parameters	325
Specifying the network type (-N)	326
Network addressing (-S and -H)	326
Starting applications on a network	327
Starting multiple brokers using the same protocol	328
Accessing a server behind a firewall	329
Start and run multi-user OpenEdge in interactive mode in Windows	330
Start and run multi-user OpenEdge in interactive mode on UNIX	330
Start and run multi-user OpenEdge clients in batch or background mode in Windows	331

Part 3 OpenEdge Products and Components

12. OpenEdge Installation Products and Components in Windows	335
OpenEdge installation options	336
Complete installation option	336
Custom installation option	336
OpenEdge product components and subcomponents	337
4GL Development System	337
AppServer Internet Adapter (AIA)	340
Client Networking	341
NameServer	343
NameServer Load Balancer	344
OpenEdge Adapter for Sonic ESB	344
OpenEdge Application Server—Basic	345

OpenEdge Application Server—Enterprise	347
Progress Developer Studio for OpenEdge	350
OpenEdge DataServer for MS SQL Server	354
OpenEdge DataServer for ODBC	356
OpenEdge DataServer for Oracle	358
OpenEdge Development Server	360
OpenEdge Enterprise RDBMS	364
OpenEdge Personal RDBMS	367
OpenEdge Replication	369
OpenEdge Replication Plus	370
OpenEdge SQL Client Access	371
OpenEdge Studio	372
OpenEdge Ultra Controls	377
OpenEdge Workgroup RDBMS	377
Query/Results	380
Mobile REST Management Agent	382
Translation Manager	383
Visual Translator	383
Web Services Adapter	385
WebSpeed Messenger	386
WebSpeed Workshop	387
OpenEdge Management SE	391
OpenEdge Business Process Modeler	391
OpenEdge Business Process Server	391
SNMP Adapter	392
OpenEdge TDE	393
OpenEdge Multi-tenant Table	393
13. OpenEdge Installation Products and Components on UNIX.....	395
OpenEdge installation options	396
Complete installation option	396
Custom installation option	396
OpenEdge product components and subcomponents	397
4GL Development System	397
AppServer Internet Adapter (AIA)	400
Client Networking	400
OpenEdge Replication	402
OpenEdge Replication Plus	403
NameServer	403
NameServer Load Balancer	404
OpenEdge Adapter for Sonic ESB	404
OpenEdge Application Server—Basic	405
OpenEdge Application Server—Enterprise	407
OpenEdge DataServer for Oracle	409
OpenEdge Development Server	411
OpenEdge Enterprise RDBMS	414
OpenEdge Personal RDBMS	417
OpenEdge Workgroup RDBMS	419
OpenEdge SQL Client Access	421
Query/Results	422
WebSpeed Messenger	424
Web Services Adapter	424
OpenEdge Management SE	425
SNMP Adapter	425
OpenEdge TDE	426
OpenEdge Multi-tenant Table	426

A. Preinstallation Checklist for Windows.....	427
Before you start	428
Products to install	428
Prerequisite third-party software (Windows 32-bit only)	428
Values from your existing OpenEdge installation (Windows 32-bit only) ..	429
Installation and working directories	430
Installation type	430
Recommended and optional components	430
Database	432
OpenEdge Adapter for Sonic ESB	432
Options to install your Progress Developer Studio for	
OpenEdge plug-ins to additional targets	433
Web server	433
Progress Dynamics (Windows 32-bit only)	434
Language in which online messages appear	435
Character set, date, and number formats	435
Web Services Adapter (WSA)	435
Options to secure your AdminServer	436
OpenEdge Business Process Server	436
B. Preinstallation Checklist for UNIX	439
Before you start	439
Java platform requirements	440
Products to install	440
Values from your existing OpenEdge installation	440
Installation and working directories	440
Installation type	441
Database	441
OpenEdge Management or OpenEdge Explorer	441
OpenEdge Adapter for Sonic ESB	442
Web server	443
Language in which messages appear	443
Character set, date, and number formats	444
Web Services Adapter (WSA)	444
OpenEdge Business Process Server	444
OpenEdge product scripts and program modules	445
Identical file exists in the installation directory	446
C. Command and Utility Reference.....	447
Administering and configuring Unified Broker products	448
ASBMAN	449
DBMAN	452
Mergeprop	454
NSMAN	456
PROADSV	458
WTBMAN	461
Installing and managing keys and digital certificates	463
certutil	464
genpassword	466
mkhashfile	467
pkiutil	468
D. OpenEdge National Language Support.....	471
Packaging	472
Directory structure	474
Contents of each directory	475

Implementing regional support	476
International databases	477
Progress messages	478
File protection	478
Environment variables of the SQL client	482
Regional parameter files	483
Progress.ini file and the Windows registry	485
E. NameServer and NameServer Load Balancing Details	487
NameServer overview	488
Unified Broker and Name Server relationship	489
Configuring NameServer communications	489
Understanding load balancing	491
Percentage weight factors	491
Arbitrary sum weight factors	492
Fail-over weight factor	492
Understanding server-level and connection-level fault tolerance	493
Connection-level fault tolerance	494
Using UDP broadcasting	494
Using NameServer replication	495
Using NameServer neighbors	498
Performance implications of broadcasting	500
Configuring OpenEdge NameServer instances	501
Downloading NameServer executables	501
Order of configuration	501
Configuring and using NameServer instances	502
Configuring the NameServer in OEM or OEE	502
Starting and managing a NameServer using OEM or OEE	503
F. Configuration Models	505
Shared-memory configurations	506
Shared-memory architecture	507
Client/server configurations	509
Terminology	509
Simple client/server configurations	510
Client/server and OpenEdge AppServer in the network environment	512
OpenEdge TCP network support	512
Preparing to run OpenEdge on a TCP/IP network	518
Installing OpenEdge on your TCP/IP network	518
Typical TCP/IP configuration with a hard disk on each machine	519
Setting up network files to run OpenEdge	519
Configuring OpenEdge on a network operating system	520
G. AdminServer Authorization and Authentication	521
AdminServer logging details	522
Log format	522
Determine the data logged in the AdminServer log	524
Setting authentication option to start servers administered by the AdminServer ...	525
H. Third Party Acknowledgements	527
Index	563

Tables

Table 1:	JDK version requirement by component	32
Table 2:	Windows system requirements to run OpenEdge	34
Table 3:	Supported platforms	37
Table 4:	Products supported by platform in Windows	38
Table 5:	Product disk space requirements in Windows	39
Table 6:	Third-party product disk space requirements in Windows	42
Table 7:	OpenEdge clients	43
Table 8:	Windows driver files for SQL	48
Table 9:	Windows driver files for the OpenEdge DataServer for ODBC	49
Table 10:	Data-source components and version numbers	50
Table 11:	JDK version requirement by component	54
Table 12:	JRE/JDK requirements by platform	58
Table 13:	Minimum requirements for running OpenEdge applications	59
Table 14:	Supported platforms	60
Table 15:	Supported 32-bit and 64-bit products by platform	61
Table 16:	Unix disk space requirements by product	63
Table 17:	Preinstallation documentation resources	69
Table 18:	Installation options	71
Table 19:	OpenEdge-install-dir (%DLC%) directory structure	79
Table 20:	OpenEdge-install-dir (\$DLC) directory structure	85
Table 21:	Progress Dynamics files that you can edit	108
Table 22:	Managers for customized session types	113
Table 23:	Data input options for a Silent installation	124
Table 24:	Available Network Server and Client Shortcuts	149
Table 25:	Data input options for a Silent installation	166
Table 26:	User-group parameter options	176
Table 27:	Port defaults	177
Table 28:	Running the SHOWCFG utility	186
Table 29:	Display fields associated with the SHOWCFG utility	187
Table 30:	Components used to calculate memory needs	196
Table 31:	Size increments for increasing startup parameters by 1	198
Table 32:	Single-user memory requirements	198
Table 33:	Multi-user memory requirements	198
Table 34:	Formulas for calculating memory requirements	199
Table 35:	Shared memory and semaphore parameter settings	201
Table 36:	Error codes and kernel reconfiguration parameters	202
Table 37:	Error messages	203
Table 38:	Reasons for altered or missing progress.cfg file	204
Table 39:	Progress event logging components	208
Table 40:	Event Level values	209
Table 41:	Windows Application Event Log components	210
Table 42:	Windows environment variables	219
Table 43:	Values for specifying IP version	232
Table 44:	Specifying IP version	233
Table 45:	UNIX environment variables	242
Table 46:	User-Group parameter options	250
Table 47:	Terminal identifiers	251
Table 48:	Elements of OpenEdge Management or OpenEdge Explorer	271
Table 49:	Default sample broker for each Unified Broker product	277
Table 50:	Command line input to the mergeprop command	292
Table 51:	Property files managed by the mergeprop utility	294
Table 52:	New value formats supported in all property files	302
Table 53:	New value formats supported in mergeprop delta files only	303
Table 54:	Value formats supported prior to OpenEdge 10	303
Table 55:	Unified Broker products and the clients they support	305

Table 56:	Ubroker.properties file structure	306
Table 57:	Additional sources of information for property files	307
Table 58:	Java properties for IPv6	309
Table 59:	Command-line utilities to start and stop installed OpenEdge products	311
Table 60:	Command-line utilities to validate property files	312
Table 61:	Windows GUI startup and shutdown commands	315
Table 62:	Windows startup and shutdown commands	316
Table 63:	OpenEdge command components	321
Table 64:	UNIX startup and shutdown commands	322
Table 65:	Client network parameters	325
Table 66:	Server network parameters	325
Table 67:	Default network types	326
Table 68:	4GL Development System components and subcomponents	337
Table 69:	AppServer Internet Adapter (AIA) components and subcomponents	340
Table 70:	Client Networking components and subcomponents	341
Table 71:	NameServer components and subcomponents	343
Table 72:	NameServer Load Balancer components and subcomponents	344
Table 73:	OpenEdge Adapter for Sonic ESB components and subcomponents	344
Table 74:	OpenEdge Application Server—Basic components and subcomponents	345
Table 75:	OpenEdge Application Server—Enterprise components and subcomponents	347
Table 76:	Progress Developer Studio for OpenEdge components and subcomponents	350
Table 77:	OpenEdge DataServer for MS SQL Server components and subcomponents	354
Table 78:	OpenEdge DataServer for ODBC components and subcomponents	356
Table 79:	OpenEdge DataServer for Oracle components and subcomponents	358
Table 80:	OpenEdge Development Server components and subcomponents	360
Table 81:	OpenEdge Enterprise RDBMS components and subcomponents	364
Table 82:	OpenEdge Personal RDBMS components and subcomponents	367
Table 83:	OpenEdge Replication components and subcomponents	369
Table 84:	OpenEdge Replication Plus components and subcomponents	370
Table 85:	OpenEdge SQL Client Access components and subcomponents	371
Table 86:	OpenEdge Studio components and subcomponents	372
Table 87:	OpenEdge Ultra Controls components and subcomponents	377
Table 88:	OpenEdge Workgroup RDBMS components and subcomponents	377
Table 89:	Query/Results components and subcomponents	380
Table 90:	Mobile REST Management Agent components and subcomponents	382
Table 91:	Translation Manager components and subcomponents	383
Table 92:	Visual Translator components and subcomponents	383
Table 93:	Web Services Adapter components and subcomponents	385
Table 94:	WebSpeed Messenger components and subcomponents	386
Table 95:	WebSpeed Workshop components and subcomponents	387
Table 96:	OpenEdge Management SE components and subcomponents	391
Table 97:	OpenEdge Business Process Modeler components and subcomponents	391
Table 98:	OpenEdge Business Process Server components and subcomponents	392
Table 99:	SNMP Adapter components and subcomponents	392
Table 100:	OpenEdge TDE components and subcomponents	393
Table 101:	OpenEdge Multi-tenant Table components and subcomponents	393
Table 102:	4GL Development System components and subcomponents	397
Table 103:	AppServer Internet Adapter components and subcomponents	400
Table 104:	Client Networking components and subcomponents	400
Table 105:	OpenEdge Replication components and subcomponents	402
Table 106:	OpenEdge Replication Plus components and subcomponents	403
Table 107:	NameServer components and subcomponents	403
Table 108:	NameServer Load Balancer components and subcomponents	404
Table 109:	OpenEdge Adapter for Sonic ESB components and subcomponents	404

Table 110: OpenEdge Application Server—Basic components and subcomponents . . .	405
Table 111: OpenEdge Application Server—Enterprise components and subcomponents	407
Table 112: OpenEdge DataServer for Oracle components and subcomponents	409
Table 113: OpenEdge Development Server components and subcomponents	411
Table 114: OpenEdge Enterprise RDBMS components and subcomponents	414
Table 115: OpenEdge Personal RDBMS components and subcomponents	417
Table 116: OpenEdge Workgroup RDBMS components and subcomponents	419
Table 117: OpenEdge SQL Client Access components and subcomponents	421
Table 118: Query/Results components and subcomponents	422
Table 119: WebSpeed Messenger components and subcomponents	424
Table 120: Web Services Adapter components and subcomponents	424
Table 121: OpenEdge Management SE components and subcomponents	425
Table 122: SNMP Adapter components and subcomponents	425
Table 123: OpenEdge TDE components and subcomponents	426
Table 124: OpenEdge Multi-tenant Table components and subcomponents	426
Table 125: Command line input to the mergeprop command	455
Table 126: proadsv command-line options	460
Table 127: PROMSGS translations shipped with OpenEdge	472
Table 128: Supplemental PROMSGS translations available for download	473
Table 129: National language file descriptions	475
Table 130: PROMSGS file synchronization process	480
Table 131: Example of PROMSGS files being out of sync	481
Table 132: Startup parameters for a deployed application	483
Table 133: Environment variables	485
Table 134: Weight factors based on percentage	491
Table 135: Weight factors based on arbitrary sums	492
Table 136: TCP/IP network files	519

Figures

Figure 1:	OpenEdge Configuration Information dialogbox from SHOWCFG utility . . .	186
Figure 2:	Product configuration details display using SHOWCFG utility	188
Figure 3:	Windows Application Event Log components	209
Figure 4:	Windows application Event Properties dialog box	211
Figure 5:	Windows 64-bit deployment	236
Figure 6:	64-bit .NET Open Client model	237
Figure 7:	Overview of the OEM or OEE interactions	270
Figure 8:	Sample Unified Broker client services file	490
Figure 9:	Server-level and connection-level fault tolerance	493
Figure 10:	NameServer replication	495
Figure 11:	NameServer neighbors	499
Figure 12:	Shared-memory OpenEdge architecture	507
Figure 13:	Simple client/server configuration	510
Figure 14:	Multiple system client/server configuration	511
Figure 15:	Simple OpenEdge network configuration	513
Figure 16:	Network file server for application files	514
Figure 17:	Network file server as a database server	515
Figure 18:	Network file server for application and database files	516
Figure 19:	LAN configuration with the OpenEdge AppServer	517
Figure 20:	Typical TCP/IP configuration (file server not used)	519

Preface

This Preface contains the following sections:

- [Purpose](#)
- [Audience](#)
- [Organization](#)
- [Using this manual](#)
- [Typographical conventions](#)
- [Examples of syntax descriptions](#)
- [Example procedures](#)
- [OpenEdge messages](#)

Purpose

This book describes installation and configuration topics related to the Release 11.3 for the following operating systems:

- Windows 2003, Windows XP Professional, Windows Server 2003 x64, Windows 8, and Windows 2012 Server (64bit) for the Intel architecture

Unless otherwise noted, platform references throughout this guide have been simplified for readability. In general Windows refers to all supported Windows 32-bit operating systems: Windows 2003 and Windows XP Professional.

Windows 64-bit is a server-only platform. See [Chapter 1, “Windows Installation Requirements”](#) for information on the products supported on Windows 64-bit, and see [Chapter 7, “Working in the OpenEdge Environment in Windows”](#) for information on Windows 64-bit functionality.

Note: Virtualization software, such as Citrix Presentation Server and VMware, are not described in this manual. Details about virtualization software support are documented in *OpenEdge 11 Platform and Product Availability Guide*.

- UNIX and Linux

Unless otherwise noted, platform references throughout this guide have been simplified for readability. UNIX refers to UNIX and Linux.

Audience

Administrative and technical personnel responsible for installing and configuring OpenEdge® Release 11.3.

Organization

[Part 1, Installation](#)

[Chapter 1, “Windows Installation Requirements”](#)

Lists system and platform prerequisites and requirements for installing OpenEdge on Windows.

[Chapter 2, “UNIX Systems Installation Requirements”](#)

Lists system and platform prerequisites and requirements for installing OpenEdge on UNIX/Linux.

[Chapter 3, “OpenEdge Installation Prerequisites”](#)

Identifies prerequisite information to know and preliminary tasks to perform before you install OpenEdge in Windows or on UNIX.

[Chapter 4, “Performing an OpenEdge Installation in Windows”](#)

Contains information related to installation and post-installation tasks for OpenEdge in Windows. (The detailed procedures to complete the Installation Utility are presented only in the Windows online help.)

[Chapter 5, “Performing an OpenEdge Installation on UNIX or Linux”](#)

Contains information related to installation and post-installation tasks when installing OpenEdge on UNIX platforms. (The detailed procedures to complete the Installation Utility are presented only in the UNIX online help.)

[Chapter 6, “Administration Utilities”](#)

Provides step-by-step instructions to perform a variety of administrative tasks and describes how to manage Windows and UNIX platform-specific resources, respectively.

[Part 2, Configuration](#)

[Chapter 7, “Working in the OpenEdge Environment in Windows”](#)

Explains how the OpenEdge environment works on Windows.

[Chapter 8, “Working in the OpenEdge Environment on UNIX”](#)

Explains how the OpenEdge environment works on UNIX.

[Chapter 9, “Managing OpenEdge Key and Certificate Stores”](#)

Describes how to use OpenEdge utilities to manage key stores for OpenEdge servers and manage certificate stores for OpenEdge clients.

[Chapter 10, “Configuration”](#)

Introduces the Unified Broker framework, a common administrative architecture for installed OpenEdge server products, and highlights the framework’s elements, focusing on Unified Brokers.

[Chapter 11, “Starting and Running OpenEdge”](#)

Provides instructions to start and connect to an OpenEdge RDBMS in different modes. Also provides information about running OpenEdge clients and servers on a network.

[Part 3, OpenEdge Products and Components](#)

[Chapter 12, “OpenEdge Installation Products and Components in Windows”](#)

Identifies the components and subcomponents associated with each product that can be installed in Windows.

[Chapter 13, “OpenEdge Installation Products and Components on UNIX”](#)

Identifies the components and subcomponents associated with each product that can be installed on UNIX.

[Appendix A, “Preinstallation Checklist for Windows”](#)

Provides a planning tool to determine and record product installation choices in Windows **before** running the OpenEdge Release 11.3 Installation Utility.

[Appendix B, “Preinstallation Checklist for UNIX”](#)

Provides a planning tool to determine and record product installation choices on UNIX **before** running the OpenEdge Release 11.3 Installation Utility.

[Appendix C, “Command and Utility Reference”](#)

Describes commands and utilities whose primary syntax and functional documentation is in this manual.

[Appendix D, “OpenEdge National Language Support”](#)

Provides information about OpenEdge messages.

[Appendix E, “NameServer and NameServer Load Balancing Details”](#)

Presents additional detailed information about the NameServer load balancing feature.

[Appendix F, “Configuration Models”](#)

Provides information about different configuration models you can reference and details about running OpenEdge installations in a network environment.

[Appendix G, “AdminServer Authorization and Authentication”](#)

Provides additional information to use the AdminServer to determine the data logged in the AdminServer log and to set authentication to start servers administered by the AdminServer.

[Appendix H, “Third Party Acknowledgements”](#)

Using this manual

The main topics presented in this guide also work with or point to related installation or configuration details presented in other OpenEdge documentation.

For the latest documentation updates see the OpenEdge Product Documentation Overview page on PSDN:

<http://communities.progress.com/pcom/docs/DOC-16074>.

Installation planning and performing

Familiarize yourself with the installation information and tasks for your operating system by proceeding as follows:

- Read the chapters in [Part 1, Installation](#) in chronological order to help you plan and perform your installation.
- Reference the information provided in [Part 3, OpenEdge Products and Components](#) that provides preinstallation checklists, and product component and subcomponent details.
- Obtain a copy of the installation online help. Reference [Table 17](#) for detailed information about where you can locate a copy of the online help.

Configuration concepts

[Part 2, Configuration](#) presents general OpenEdge configuration concepts and arrangements. Reference details presented in these chapters:

- [Chapter 9, “Managing OpenEdge Key and Certificate Stores”](#)
- [Chapter 10, “Configuration”](#)

As needed, these chapters point to other product documentation for configuration details.

References to ABL compiler and run-time features

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

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

References to ABL data types

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



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

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

Typographical conventions

This manual uses the following typographical conventions:

Convention	Description
Bold	Bold typeface indicates commands or characters the user types, provides emphasis, or the names of user interface elements.
<i>Italic</i>	Italic typeface indicates the title of a document, or signifies new terms.
SMALL, BOLD CAPITAL LETTERS	Small, bold capital letters indicate OpenEdge key functions and generic keyboard keys; for example, GET and CTRL .
KEY1+KEY2	A plus sign between key names indicates a simultaneous key sequence: you press and hold down the first key while pressing the second key. For example, CTRL+X .
KEY1 KEY2	A space between key names indicates a sequential key sequence: you press and release the first key, then press another key. For example, ESCAPE H .
Syntax:	
Fixed width	A fixed-width font is used in syntax statements, code examples, system output, and filenames.
<i>Fixed-width italics</i>	Fixed-width italics indicate variables in syntax statements.

Convention	Description
<i>Fixed-width bold</i>	Fixed-width bold indicates variables with special emphasis.
UPPERCASE fixed width	Uppercase words are ABL keywords. Although these are always shown in uppercase, you can type them in either uppercase or lowercase in a procedure.
	This icon (three arrows) introduces a multi-step procedure.
	This icon (one arrow) introduces a single-step procedure.
Period (.) or colon (:))	All statements except DO, FOR, FUNCTION, PROCEDURE, and REPEAT end with a period. DO, FOR, FUNCTION, PROCEDURE, and REPEAT statements can end with either a period or a colon.
[]	Large brackets indicate the items within them are optional.
[]	Small brackets are part of ABL.
{ }	Large braces indicate the items within them are required. They are used to simplify complex syntax diagrams.
{ }	Small braces are part of ABL. For example, a called external procedure must use braces when referencing arguments passed by a calling procedure.
	A vertical bar indicates a choice.
...	Ellipses indicate repetition: you can choose one or more of the preceding items.

Examples of syntax descriptions

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

Syntax

```
ACCUM aggregate expression
```

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

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

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

Syntax

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

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

Syntax

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

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

Syntax

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

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

Syntax

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

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

Syntax

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

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

Syntax

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

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

Syntax

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


Long syntax descriptions split across lines

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

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

Syntax

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

Complex syntax descriptions with both required and optional elements

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

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

Syntax

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

Example procedures

This manual provides example code that illustrates syntax and concepts. You can access many of the example files, and details for installing them, from the following locations:

- A self-extracting Documentation and Samples file available on the OpenEdge download page of the Progress Software Download Center
- The OpenEdge Product Documentation Overview page on PSDN:

<http://communities.progress.com/pcom/docs/DOC-16074>

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**.

Part 1

Installation

[Chapter 1, Windows Installation Requirements](#)

[Chapter 2, UNIX Systems Installation Requirements](#)

[Chapter 3, OpenEdge Installation Prerequisites](#)

[Chapter 4, Performing an OpenEdge Installation in Windows](#)

[Chapter 5, Performing an OpenEdge Installation on UNIX or Linux](#)

[Chapter 6, Administration Utilities](#)

Windows Installation Requirements

This chapter details what you need to know before installing OpenEdge® Release 11.3 on the supported Windows platforms as described in the following sections:

- [System requirements](#)
- [Supported platforms](#)
- [Server compatibility](#)
- [Required third-party applications](#)
- [Licensing](#)

System requirements

This section describes the hardware and software requirements to run OpenEdge® Release 11.3 on the following supported platforms:

- Windows 2003
- Windows XP SP3
- Windows 7
- Windows 2008 R2

The specific support requirements for virtualization software, such as Citrix Presentation Server and VMware, is determined by, and dependent upon, the vendor's support for the underlying operating system.

To ensure that you have the most up-to-date information about system requirements, refer to *OpenEdge Release Notes* available with your installation package and *OpenEdge 11 Platform & Product Availability Guide* on the Progress Software Corporation (PSC) Web site

<http://www.progress.com/products/lifecycle/index.ssp>, as needed.

Ensuring you have the most up-to-date system requirements information

The system requirements provided in this chapter are current as of the publication date of this manual; however, requirements can change. To ensure that you have the most up-to-date information about system requirements, refer to *OpenEdge Release Notes* available with your installation medium and the *OpenEdge 11 Platform & Product Availability Guide* on the Progress Software Corporation Web site

<http://www.progress.com/products/lifecycle/index.ssp>.

Java considerations

Many OpenEdge products require the Java Run-time Environment (JRE), the Java Development Kit (JDK), or both of these components to use specific product functionality once the products are installed. The OpenEdge installation automatically installs the required JDK/JRE components in Windows.

Table 1 lists the JDK version required by component supported with this release.

Table 1: JDK version requirement by component (1 of 3)

Component	Version
OE Personal RDBMS	1.7_02
OE Personal RDBMS	1.7_02
OE Enterprise RDBMS	1.7_02
OE Workgroup RDBMS	1.7_02

Table 1: JDK version requirement by component (2 of 3)

Component	Version
OE DataServer for Oracle	1.7_02
OE DataServer for ODBC	1.7_02
OE DataServer MS SQL Svr	1.7_02
OE Development Server	1.7_02
OE Application Svr Basic	1.7_02
OE Application Svr Ent	1.7_02
OE Studio	1.7_02
OE Adap Sonic ESB	1.7_02 and 1.6 Runtime
OE SQL Client Access	1.7_02
Client Networking	1.7_02
Translation Manager	1.7_02
OE Replication	1.7_02
OE Repl Plus	1.7_02
4GL Development System	1.7_02
Visual Translator	1.7_02
Query/RESULTS	1.7_02
WebSpeed Workshop	1.7_02
WebSpeed Messenger	1.7_02
Mobile REST Mgt Agent	1.7_02 and 1.6 Runtime
NameServer	1.7_02
NameServer Load Balance	1.7_02
Progress Developer Studio for OpenEdge	1.7_02
OpenEdge Ultra Controls	1.7_02
AppServer IntAdap	1.7_02 and 1.6 Runtime
WSA	1.7_02 and 1.6 Runtime
OE BP Server	1.7_02
OE BP Modeler	1.7_02
SNMP Adapter	1.7_02

Table 1: JDK version requirement by component (3 of 3)

Component	Version
OpenEdge TDE	1.7_02
OE Multi-tenant Tables	1.7_02
OpenEdge Management	1.7_02

Note: Java 1.7 update 21 and higher and 1.6 update 45 and higher are not supported.

To determine what version of Java you currently have on your operating system, type `java -version` at the command line.

Note: On some platforms, multiple versions of Java may be available. You can change your `PATH` variable to reference a different version, if needed. In the case of the Open Client Toolkit, other Java versions (including versions from other vendors) can also be used.

For specific information about these components, see the *OpenEdge 10 Platform & Product Availability Guide* on the Progress Software Corporation Web site <http://www.progress.com/products/lifecycle/index.ssp>.

Windows system requirements

Table 2 lists the minimum requirements for running OpenEdge in Windows.

Table 2: Windows system requirements to run OpenEdge (1 of 3)

Component	Requirement
Development System: <ul style="list-style-type: none"> Progress Developer Studio for OpenEdge, including OpenEdge Development Server ABL client (using Client Networking or WebClient) Database Server Application Server (AppServer or WebSpeed Transaction Server) 	A Pentium PC (or compatible computer) with these characteristics: <ul style="list-style-type: none"> Clock speed — A minimum of 1GHz Memory — A minimum of 1GB
Deployment/Production system	A Pentium PC (or compatible computer) with these characteristics: <ul style="list-style-type: none"> Clock speed — A minimum of 433MHz Memory — A minimum of 256MB

Table 2: Windows system requirements to run OpenEdge (2 of 3)

Component	Requirement
OpenEdge database server	<p>A Pentium PC (or compatible computer) with these characteristics:</p> <ul style="list-style-type: none"> • Clock speed — A minimum of 1GHz • Memory — A minimum of 1GB
Hard disk drive Disk space	<p>Sufficient disk space to hold the OpenEdge products you want to install.</p> <p>Sufficient disk space on the drive where Windows is installed for Windows system files that OpenEdge copies there. Depending on which Windows system files are already installed on your system, OpenEdge can install up to 50MB in your Windows directory.</p>
Java	<p>See the “Java considerations” section on page 32 for detailed information about the Java requirements if you are installing any of these products: the OpenEdge® Enterprise RDBMS, OpenEdge® Workgroup RDBMS, or the OpenEdge® Personal RDBMS, or the components OpenEdge SQL Client Access.</p>
OpenEdge installation medium	<p>To access the OpenEdge product installation program, Progress Software Corporation supports Electronic Software Distribution (ESD) download. It facilitates downloading software images from the Progress Download Center available at http://www.progress.com/esd. This Web site requires a valid account that your company must establish with Progress Software Corporation to access OpenEdge products and updates.</p>
Network protocol	<p>Your system must support the TCP/IP and UDP protocols.</p>

Table 2: Windows system requirements to run OpenEdge (3 of 3)

Component	Requirement
Web server	<p>Your Web server must support one of the following interfaces:</p> <ul style="list-style-type: none"> • Microsoft Web server (IIS) or ISAPI compatible — For example, Microsoft Internet Information Server (IIS), Versions 5.x and later. The IIS Web server ships with Windows, but is not installed by default. This Web server supports an “in memory” messenger (ISAPI) and CGI messenger. • Sun Web server or NSAPI compatible — For example, a Sun Web server (formerly Netscape, iPlanet and SunOne Web Servers). This Web server supports an “in memory” messenger (NSAPI) and the CGI messenger. • CGI 1.1 — For example, Microsoft Internet Information Server (IIS), Versions 3.x and 4.x, or Apache. This Web server provides support for the Common Gateway Interface (CGI) messenger.
Web browser	<p>To use WebSpeed®, you must have a Web browser. Most contemporary browsers such as Mozilla, Opera, or Microsoft Internet Explorer, will work with WebSpeed.</p> <p>To run WebSpeed Workshop, Progress Software Corporation recommends you use Microsoft Internet Explorer, Version 6. The version of Internet Explorer you are using must support JScript Version 2.0.</p>

Supported platforms

This section describes the platforms and components supported in OpenEdge 11.3, and provides additional details about specific platforms and platform-related features. Refer to the hard-copy and online *OpenEdge Release Notes* for additional requirements.

Table 3 lists the platforms supported with this release and the minimum operating system level supported.

Table 3: Supported platforms

Platform	Build Platform	Certification Platform
Windows 32bit (INTEL)	Windows 2003 Server R2	Windows XP Professional SP3
		Windows 7
		Windows 2008 Server R2
		Windows 8 (32 and 64 bit)
		Windows 2012 Server
Windows 64bit (INTEL)	Windows 2008 Server R2	Windows 8 (64 bit)
		Windows 2012 Server

This is the most up-to-date information on supported platforms. For the most recent changes to the list of supported platforms, see *OpenEdge 11 Platform & Product Availability Guide* on the PSC Web site:

<http://www.progress.com/products/lifecycle/index.ssp>.

Supported products by platform

Table 4 lists the products supported on Windows 32-bit and Windows 64-bit. For more details about the Windows 64-bit functionality, see the “[Windows 64-bit](#)” section on page 234.

Table 4: Products supported by platform in Windows (1 of 2)

Product	Windows 32-bit	Windows 64-bit
OE Personal RDBMS	✓	✓
OE Workgroup RDBMS	✓	✓
OE Enterprise RDBMS	✓	✓
OE DataServer for Oracle	✓	✓
OE DataServer for ODBC	✓	✓
OE DataServer MS SQL Svr	✓	✓
OE Development Server	✓	✓
OE Application Svr Basic	✓	✓
OE Application Svr Ent	✓	✓
OE Studio	✓	✓
OE Adap Sonic ESB	✓	✓
OE SQL Client Access	✓	✓
Client Networking	✓	✓
Translation Manager	✓	✓
OpenEdge Replication (Workgroup and Enterprise)	✓	✓
OpenEdge Repl Plus (Workgroup and Enterprise)	✓	✓
4GL Development System	✓	✓
Visual Translator	✓	✓
Query/RESULTS	✓	✓
WebSpeed Workshop	✓	✓
WebSpeed Messenger	✓	✓

Table 4: Products supported by platform in Windows (2 of 2)

Product	Windows 32-bit	Windows 64-bit
NameServer	✓	✓
NameServer Load Balance	✓	✓
Progress Developer Studio for OpenEdge	✓	✓
OpenEdge Ultra Controls	✓	✓
AppServer IntAdap	✓	✓
WSA	✓	✓
OpenEdge Management	✓	✓
SMNP Adapter	✓	✓
OE BP Server	✓	✓
OE BP Modeler	✓	✓
OpenEdge Transparent Data Encryption	✓	✓
OpenEdge Multi-tenant Tables	✓	✓
WebClient	✓	—

Disk space requirements

[Table 5](#) lists the approximate disk space requirements for each OpenEdge product. This is the size of the product as installed in your installation directory. Additional space is consumed by third-party products that are installed in their own directories. See [Table 6](#) for details.

Table 5: Product disk space requirements in Windows (1 of 2)

Product	Average size Win 32-bit in MB	Average size Win 64-bit in MB
OE Personal RDBMS	871	1119
OE Workgroup RDBMS	841	1089
OE Enterprise RDBMS	841	1089
OE DataServer for Oracle	907	970
OE DataServer for ODBC	1022	1100
OE DataServer MS SQL Svr	1021	1099
OE Development Server	1207	1261

Table 5: Product disk space requirements in Windows (2 of 2)

Product	Average size Win 32-bit in MB	Average size Win 64-bit in MB
OE Application Svr Basic	738	843
OE Application Svr Ent	788	894
OE Studio	1574	1348
OE Adap Sonic ESB	299	305
OE SQL Client Access	260	367
Client Networking	567	704
Translation Manager	279	285
OE Replication	291	298
OE Repl Plus	291	298
4GL Development System	845	1188
Visual Translator	459	670
Query/RESULTS	464	608
WebSpeed Workshop	1182	1456
WebSpeed Messenger	146	146
Mobile REST Mgt Agent	16	16
NameServer	506	540
NameServer Load Balance	241	246
Progress Developer Studio for OpenEdge	2804	2900
OpenEdge Ultra Controls	2	2
AppServer IntAdap	142	143
WSA	112	112
OE BP Server	1443	1491
OE BP Modeler	669	680
SNMP Adapter	85	85
OpenEdge TDE	241	247
OE Multi-tenant Tables	241	247
OpenEdge Management	572	578

Note: Because products may contain common components, your actual disk space requirements, will not precisely equal the sum of size of all the products you installed.

Third party products installed by OpenEdge require additional disk space. [Table 6](#) details the additional disk space consumed.

Table 6: Third-party product disk space requirements in Windows

OpenEdge product	Disk space consumed by required third-party products ¹
All development products	500 MB
OpenEdge Ultra Controls	200 MB

1. Third party products are installed on your C: drive.

Server compatibility

If you run an OpenEdge multi-user system that includes older versions of OpenEdge (or Progress) products, make sure that your servers are compatible. The following sections detail OpenEdge RDBMS, supported Progress and OpenEdge DataServers, Progress® AppServer™, Transaction Server, AdminServer, and OpenEdge® Name Server™ compatibility.

OpenEdge clients

[Table 7](#) describes the clients supported by OpenEdge servers.

Table 7: OpenEdge clients

OpenEdge Server	Client
OpenEdge RDBMS Server	ABL client (character or GUI)
	WebSpeed Agent
	AppServer
OpenEdge DataServer server	ABL client (character or GUI)
	WebSpeed Agent
	AppServer
AppServer	ABL client
	Java Client
	WebSpeed Agent
	AppServer
	Web Services Adapter
	OpenEdge Adapter for Sonic ESB
	.NET client
OpenEdge SQL Database Server	OpenEdge SQL ODBC client
	OpenEdge SQL JDBC client

General connectivity and compatibility rules

Connectivity rules between clients and servers are as follows:

- Client products can connect to server products of the same release number and one major release backward.

For example, OpenEdge Release 11 clients can connect to both OpenEdge Release 11 and Release 10 Application and Database Servers.

- Client products can connect to AppServer products of one major release backward and one major release forward.

For example OpenEdge Release 10 clients can connect to OpenEdge Release 11 Application Servers.

Caution: Clients prior to OE10.2A03 and OE10.2B01 are implemented using the Progress V9 Application Server protocol, and appear to be Progress V9 Clients from a system perspective. Consequently these clients cannot interact with an OpenEdge Release 11 Application Server.

OpenEdge Release 10.2A clients must be at Release 10.2A03 or later, and OpenEdge Release 10.2B clients must be at Release 10.2B01 or later in order to connect to an OpenEdge Release 11 Application Server.

- Server products, when connecting to another server, behave like a client, and follow the client rules stated previously.
- Client to client compatibility is not applicable.
- Shared memory connections between clients and server require the release numbers to match exactly.
- Database migration/conversion is not required because there is no internal database major-version change this release.
- R-code compilation is not required because there is no r-code major-version change in 11.3. An OpenEdge Release 11.3 client can execute r-code generated with 11.0. However an OpenEdge Release 11.3 client can not execute r-code generated with release 10, 9, or earlier.

Note: An OpenEdge Release 11.1 client can execute r-code generated with 11.3. You must recompile the r-code generated with 11.3.

- R-code is portable across all platforms of the same user interface type.

OpenEdge SQL

OpenEdge SQL ODBC and JDBC clients can connect to a server of the current release, and one release forward and back. For example, 11.3 clients can connect to a 11.3, 11.2 or a 11.1 server; 11.1 clients can connect to a 11.1 or 11.2 server.

Deployment rules supported by a DataServer broker

If you are using the DataServer brokering technology for an n-tier deployment, the OpenEdge versions of the client and the DataServer broker must be at the same maintenance level. For example, an OpenEdge 11.3 client cannot connect to an OpenEdge 11.2 DataServer broker. Similarly, an OpenEdge 11.2 client cannot connect to an OpenEdge 11.3 broker.

Development rules related to schema holder compatibility

Normal OpenEdge RDBMS database rules apply if the Schema Holder does not contain newer features or functionality that are not supported by the lesser release being used by the client.

For example, OpenEdge 11.3 and 11.2 clients can connect to an OpenEdge 11.3-created schema holder being served in multi-user mode by an OpenEdge 11.3 RDBMS broker. However, if you update the 11.2 schema holder using OpenEdge 11.3 and enable new features, then the OpenEdge 11.2 client will no longer be supported, and you might experience runtime errors.

Apache Tomcat

Apache Tomcat installed with Progress Developer Studio for OpenEdge is the default Web Server 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, \$DLC/servers/tomcat.

Required third-party applications

The following sections describe the third-party applications that OpenEdge installs during the installation process if you do not currently have the minimum required versions of these third-party applications on your system:

- [Microsoft .NET Framework](#)
- [Infragistics NetAdvantage](#)
- [DataDirect ODBC branded drivers](#)

Microsoft .NET Framework

If Microsoft .NET Framework 4.0 is not currently installed on the system where you plan to install OpenEdge Release 11.3, and you are installing any of the development products listed below, the OpenEdge Installation Utility automatically launches the Microsoft .NET Framework installation once installation of the OpenEdge products is complete. As part of the installation process, you are required to explicitly accept the License Agreement for the Microsoft .NET Framework.

The development products that require the Microsoft .NET Framework 4.0 are:

- 4GL Development System
- OpenEdge Development Server
- OpenEdge Studio
- Progress Developer Studio for OpenEdge
- OpenEdge Ultra Controls

Notes: OpenEdge installs the English version of the Microsoft .NET Framework. If you require a different language version, you must install it **before** you install OpenEdge.

Frameworks in additional languages are available from the Progress Download Center at <http://www.progress.com/esd>.

If you plan a Silent installation that includes OpenEdge products that require Microsoft .NET Framework, you must verify that the .NET Framework software is available on the system you are installing on **before** you initiate the installation. Otherwise, the Silent installation process will terminate. The License Agreement **must** be accepted interactively.

If you are installing only deployment products, the core installation process gives you the option to install the .NET Framework if needed during the installation, by checking the appropriate check box. For Net Setup, the installation process gives you the option to install the .NET Framework if needed. The .NET Framework installation is saved in your OpenEdge installation directory during the core install for this purpose. For WebClient, the installer does not contain the .NET Framework installation. The WebClient Application Assembler provides the ability to embed and install the .NET Framework when your application is deployed.

Infragistics NetAdvantage

If you are installing the OpenEdge Ultra Controls in Windows, there is a dependency on both the Microsoft .NET Framework and the Infragistics NetAdvantage for .NET v2013 Vol 1 UI Controls.

The OpenEdge Ultra Controls can only be installed if one of the following development products is also being installed, or you are adding to an existing installation where the development product is installed:

- Progress Developer Studio for OpenEdge
- OpenEdge Studio
- 4GL Development System
- OpenEdge Development Server

The NetAdvantage installation process launches after the OpenEdge core installation completes. If the installation of the Microsoft .NET Framework is also required, it is installed before NetAdvantage. The .dll files containing the Net Advantage controls are installed into *OpenEdge-install-dir\bin\infragistics\winforms*.

The NetAdvantage files are inserted into your installation in the *OpenEdge-install-dir\bin\infragistics\winforms* subdirectory when the development products listed previously are installed, and when the following deployment products are installed:

- OpenEdge Personal RDBMS
- Client Networking

For a Web Client installation, the Web Client Application Assembler is responsible for installing the NetAdvantage files.

NetAdvantage has its own Help subsystem this is automatically installed with the product. To access online help for NetAdvantage, which is available on its site, press F1.

DataDirect ODBC branded drivers

In OpenEdge 11.3 Release, the 32-bit and the 64-bit DataDirect 7.0 drivers are installed with MS SQL DataServer. The driver branding process customizes both the 32-bit and the 64-bit DataDirect drivers to be used only with the OpenEdge DataServer products for MS SQL Server and ODBC.

Note: Any attempts to use the drivers with other ODBC applications result in errors.

If you are installing OpenEdge on Windows, you must have the following Microsoft Installers for this installation:

- **The MDAC 2.6 Installer from Microsoft** — Installs Microsoft Data Access Components (MDAC) in Windows. The OpenEdge Installation Utility contains the MDAC 2.6 Installer, which automatically launches during installation. The MDAC 2.6 Installer is located at *OpenEdge-install-dir\odbc\install\mdac_typ.exe*.

You can find more information about the MDAC at Microsoft's Web site at the following URL: <http://msdn.microsoft.com/en-us/library/ms810805.aspx>.

- **The DCOM98 Installer from Microsoft** — Installs the Distributed Component Object Model (DCOM). The OpenEdge Installation Utility installer contains the DCOM98 Installer, which automatically launches during installation. The DCOM98 Installer is located at *OpenEdge-install-dir\odbc\install\dcom98.exe*. You can find more information about DCOM98 at Microsoft's Web site by searching on DCOM98 at the following URL: <http://www.microsoft.com/>.

You must reboot your system after installing either the MDAC or the DCOM98 Installer.

DataDirect SQL ODBC drivers

The Installation Utility installs the DataDirect SQL ODBC driver files to the *installation-path\bin* directory.

Note: Both the 32bit ODBC drivers and the 64bit ODBC drivers can operate at the same time on the same machine.

Table 8 lists the SQL driver files installed for the OpenEdge database.

Table 8: Windows driver files for SQL

Database	Driver files
OpenEdge SQL	<p>pgodbc.lic</p> <p>pgicu26.dll</p> <p>pgssl26.dll</p> <p>pgoe26.dll¹</p> <p>pgoe26r.dll</p> <p>pgtrc26.dll</p>

1. Identifies the primary OpenEdge driver file. The pgoe26.dll must be registered as a file data source name (DSN).

Installing the DataDirect branded ODBC drivers

The Installation Utility installs the DataDirect branded ODBC driver files to the *OpenEdge-install-dir\odbc\bin* directory.

[Table 9](#) lists the DataDirect branded ODBC driver files installed with the OpenEdge DataServer for ODBC.

Table 9: Windows driver files for the OpenEdge DataServer for ODBC

Database	Driver files
All ODBC data sources	PGASE26.DLL PGASE26R.DLL PGDB226.DLL PGDB226R.DLL PGICU26.DLL PGMSSS26.DLL PGMSSS26R.DLL PGSSL26.dll IVPG.LIC DDPG.LIC
DB2 UDB Common Server (32-bit)	PGDB226.DLL PGDB226R.DLL IVPG.LIC
DB2 UDB Common Server (64-bit)	PGDB226.DLL PGDB226R.DLL DDPG.LIC
Microsoft SQL Server 2000 and later (32-bit)	PGMSSS26.DLL PGMSSS26R.DLL IVPG.LIC
Microsoft SQL Server 2000 and later (64-bit)	PGMSSS26.DLL PGMSSS26R.DLL DDPG.LIC
Sybase (32-bit)	PGASE26.DLL PGASE26R.DLL IVPG.LIC
Sybase (64-bit)	PGASE26.DLL PGASE26R.DLL DDPG.LIC

Note: Refer to the *odbceref.pdf* file installed in the *OpenEdge-install-dir\odbc\doc* subdirectory for information about how to configure data sources to connect to the different databases that OpenEdge supports.

Additional DataServer information for ODBC-related components

To use DataServer for ODBC:

- Progress Software Corporation recommends you use DataDirect Version 7.0.
- You must have the specific data-source software components and version numbers installed for the specific data sources you are using.

[Table 10](#) presents the specific data-source software requirements.

Table 10: Data-source components and version numbers

If you are using the DataServer for ODBC product from this data source ...	Then you must install ...
DB2	DB2/NT Version 9.1 or later
DB2/400	DB2/400 V5r3 or V5r4
Sybase	Sybase adaptive Server System 15 or later

Licensing

When installing OpenEdge, the installation utility prompts you to enter product information from your *License Addendum*. The installation utility records the license information in the OpenEdge configuration file, `progress.cfg`.

Note: For information about using a License Addendum File, see the [“Obtaining an Electronic License Addendum file”](#) section on page 72.

The following syntax shows how to use the Show Configuration (`SHOWCFG`) utility to display the product license information for each OpenEdge product installed on your system:

Syntax

```
OpenEdge-install-dir/bin/showcfg OpenEdge-install-dir/progress.cfg
```

For more information about licensing, see [Chapter 6, “Administration Utilities.”](#)

UNIX Systems Installation Requirements

This chapter describes the requirements for installing OpenEdge Release 11.3 on a machine running a UNIX or Linux operating system. This chapter also identifies the supported platforms on which you can install and run OpenEdge, as outlined in the following sections:

- [UNIX system requirements](#)
- [Supported platforms](#)
- [Licensing](#)

UNIX system requirements

This section describes the hardware and software requirements for running OpenEdge Release 11.3 on UNIX and Linux, as described in the following sections:

- [Requirements for using Java](#)
- [Requirements for running OpenEdge applications](#)

The system requirements provided in this chapter are current as of the publication date of this manual; however, requirements can change. To ensure that you have the most up-to-date information about system requirements, refer to *OpenEdge Release Notes* available with your installation package and *OpenEdge 10 Platform & Product Availability Guide* available on the Progress Software Corporation (PSC) Web site <http://www.progress.com/products/lifecycle/index.ssp>.

Requirements for using Java

Many OpenEdge products require the Java Runtime Environment (JRE), the Java Development Kit (JDK), or both of these components to use specific product functionality once the products are installed.

OpenEdge supports Java version 6.0. OpenEdge components that rely on Java use 32bit JVM on 32bit platforms and 64bit JVM on 64bit platforms. This support enables stored procedures and triggers for 64-bit platforms.

JDK component

The JDK contains the software and tools that developers need to compile, debug, and run applets and applications written using the Java programming language. The JDK software and documentation, typically included with each new release of an operating system, are available for download at the vendor's Web site. You need a JDK if you intend to develop Java stored procedures and triggers with the database, or if you intend to create Java proxies with the Progress® Open Client Toolkit.

[Table 11](#) lists the JDK version required by component supported with this release.

Table 11: JDK version requirement by component (1 of 2)

Component	Version
OE Personal RDBMS	1.7_02
OE Personal RDBMS	1.7_02
OE Enterprise RDBMS	1.7_02
OE Workgroup RDBMS	1.7_02
OE DataServer for Oracle	1.7_02
OE DataServer for ODBC	1.7_02
OE DataServer MS SQL Svr	1.7_02
OE Development Server	1.7_02

Table 11: JDK version requirement by component (2 of 2)

Component	Version
OE Application Svr Basic	1.7_02
OE Application Svr Ent	1.7_02
OE Studio	1.7_02
OE Adap Sonic ESB	1.7_02 and 1.6 Runtime
OE SQL Client Access	1.7_02
Client Networking	1.7_02
Translation Manager	1.7_02
OE Replication	1.7_02
OE Repl Plus	1.7_02
4GL Development System	1.7_02
Visual Translator	1.7_02
Query/RESULTS	1.7_02
WebSpeed Workshop	1.7_02
WebSpeed Messenger	1.7_02
Mobile REST Mgt Agent	1.7_02 and 1.6 Runtime
NameServer	1.7_02
NameServer Load Balance	1.7_02
Progress Developer Studio for OpenEdge	1.7_02
OpenEdge Ultra Controls	1.7_02
AppServer IntAdap	1.7_02 and 1.6 Runtime
WSA	1.7_02 and 1.6 Runtime
OE BP Server	1.7_02
OE BP Modeler	1.7_02
SNMP Adapter	1.7_02
OpenEdge TDE	1.7_02
OE Multi-tenant Tables	1.7_02
OpenEdge Management	1.7_02

Notes: Java 1.7 update 21 and higher and 1.6 update 45 and higher are not supported. For details about the Release 11.3 supported platforms and specific Java requirements needed to support an OpenEdge installation on each platform, see the [“Supported platforms with Java components”](#) section on page 57.

Java versions

To determine what version of Java you currently have on your operating system, type `java -version` at the command line.

If Java software is not supplied with your installation package, you must verify that it is correctly installed on your system, according to the previous criteria.



To ensure that the correct Java version is properly installed and recognized by the OpenEdge installation:

1. Install the certified JDK to be used with OpenEdge Release 11.3 before you install OpenEdge.
2. Verify that the JDK is located in the `$PATH` environment variable to ensure that the OpenEdge installation can tailor the `java_env` file.

The `$PATH` environment variable must point to the correct Java installation **before** you run the `proinst` utility. Otherwise, the system default Java executable's version is referenced from the `PATH`; the system default is not necessarily the correct Java version for the OpenEdge installation.

3. Verify that the JDK is located in `$JAVAHOME/bin` environment variable so that the Installation Utility can find it. (The `JAVAHOME` `PATH` is the Java installation directory.)

OpenEdge products that require the JRE

The JRE consists of the Java Virtual Machine, the Java Core Classes, and the supporting files. The JRE is the run time part of the JDK and does not include a compiler, a debugger, or development tools. You must have the JRE if you intend to use one of the following:

- OpenEdge Management or OpenEdge Explorer
- WebSpeed Transaction Server
- OpenEdge® Application Server Basic
- OpenEdge® Application Server Enterprise
- Java application or applet
- OpenEdge® Adapter for SonicMQ®
- Web Services Adapter
- Secure AppServer Internet Adapter (AIA)

You must have the JRE to execute Java stored procedures and triggers from the database.

Supported platforms with Java components

As mentioned earlier in this chapter, the OpenEdge installation program also automatically installs the JDK when you install a product that requires the JDK on all the platforms.

Operating systems and JDK and JRE version details

Table 12 lists operating systems, specifies the versions of JDK and JRE each supports, and provides a URL for more information about the JDK associated with a platform.

Table 12: JRE/JDK requirements by platform

Platform	Entry level JDK/JRE required	URL location
Solaris SPARC (64-bit)	1.6.0_27 (JDK/JRE ships with OpenEdge)	http://java.sun.com
HP-UX (PA-RISC) (64-bit)	1.6.0.07	http://www.hp.com/java
HP-UX ITANIUM (IA 64)	1.6.0.07	http://www.hp.com/java
IBM AIX 6.1 (64-bit)	1.6.0_20071205_6 4	http://www.ibm.com/java
Linux Intel	1.6.0_26	http://java.sun.com

- To determine what version of Java you currently have on your operating system, type **java -version** at the command line.

Note: On some platforms, multiple versions of Java may be available.

You can change your `PATH` variable to reference a different version, if needed. In the case of the Open Client Toolkit, other Java versions (including versions from other vendors) can also be used.

Support for 64-bit JVM

OpenEdge components that rely on Java have 64-bit JVM support. This enhances support for stored procedures and triggers on 64-bit platforms by enabling the OpenEdge SQL server to load the 64-bit Java Virtual Machine (JVM). As a result, stored procedures and triggers can be created, compiled, and executed by the 64-bit JVM loaded by the OpenEdge SQL Server.

The distribution of 64-bit Java in JDK/JRE differs from platform to platform. For example, AIX64 and Linux64 have separate 64-bit JDK/JRE packages.

Requirements for running OpenEdge applications

[Table 13](#) lists the minimum requirements for running OpenEdge applications.

Table 13: Minimum requirements for running OpenEdge applications

Component	Requirement
Terminal	A character terminal attached to a host computer. Note: OpenEdge does not support spacetaking terminals unless the terminal has a firmware setup option to change it to nonspacetaking mode.
Libraries	Networking libraries must be installed on your machine. Multi-user OpenEdge configurations connect UNIX-to-UNIX through the OpenEdge-supported network protocol TCP/IP. You can also connect to a UNIX server from a Windows client through TCP/IP.
JDK	Installed JDK components. See Table 12 for the current versions of JDK releases.

Product and application dependencies

Additional requirements might exist, depending on the applications you plan to run and/or the OpenEdge products you plan to install. For example, you might need any or all of the following elements:

- **Web server** — To run WebSpeed (Web server types supported include Microsoft IIS Web server or ISAPI-compatible, Sun Web server or NSAPI-compatible, or CGI-compatible)
- **Java servlet engine** — To run Web Services
- **JRE/JVM** — To run Java applications
- **Apache Tomcat Web server** — To run OpenEdge development products (such as Progress Developer Studio for OpenEdge)

Note: Apache Tomcat installed with Progress Developer Studio for OpenEdge is the default Web Server where you can deploy the REST Web applications.

File descriptors

You must allocate sufficient file descriptors to handle all the WebSpeed Agents your configuration uses. Set the file descriptors to 1024 by entering the following command:

```
ulimit -n 1024
```

Supported platforms

This section describes the platforms and components supported in OpenEdge 11.3, and provides additional details about specific platforms and platform-related features. Refer to the hard-copy and online *OpenEdge Release Notes* for additional requirements.

Table 14 lists the platforms supported with this release and the minimum operating system level supported.

Table 14: Supported platforms

Platform	Build Platform	Certification Platform
HP-UX 64-bit (PA-RISC)	HP-UX 11i v3	
HP-UX 64-bit (ITANIUM)	HP-UX 11i v3	
IBM AIX 6.1 64-bit (POWER)	AIX 6.1	AIX 7.1
Linux 32-bit (INTEL)	RedHat Enterprise AS 5.0 Update 4	SuSE Enterprise Server 11
		Oracle Linux 5.4
		CentOS 5.4
		CentOS 6.1
		RedHat 6.1
Linux 64-bit (INTEL)	RedHat Enterprise AS 5.0 Update 4	SuSE Enterprise Server 11
		Oracle Linux 5.4
		CentOS 5.4
		CentOS 6.1
		RedHat 6.1
Solaris 64-bit (SPARC)	Solaris 10	Solaris 11

This is the most up-to-date information on supported platforms. For the most recent changes to the list of supported platforms, see *OpenEdge 10 Platform & Product Availability Guide* on the PSC Web site:

<http://www.progress.com/products/lifecycle/index.ssp>.

Supported products by platform

The section describes the products available on each of the supported 32-bit Linux and 64-bit UNIX and Linux platforms.

[Table 15](#) lists the supported components by platform for Unix and Linux platforms.

Table 15: Supported 32-bit and 64-bit products by platform (1 of 2)

Component	HP-UX Itanium 64-bit	HP-UX PA- RISC 64-bit	AIX 64-bit	Linux Intel 64-bit	Linux Intel 32-bit	Solaris 64-bit
OE Personal RDBMS	✓	✓	✓	✓	✓	✓
OE Workgroup RDBMS	✓	✓	✓	✓	✓	✓
OE Enterprise RDBMS	✓	✓	✓	✓	✓	✓
OE DataServer for Oracle	✓	✓	✓	✓	✓	✓
OE Development Server	✓	✓	✓	✓	✓	✓
OE Application Svr Basic	✓	✓	✓	✓	✓	✓
OE Application Svr Ent	✓	✓	✓	✓	✓	✓
OE Adap Sonic ESB	✓	—	✓	✓	✓	✓
OE SQL Client Access	✓	✓	✓	✓	✓	✓
Client Networking	✓	✓	✓	✓	✓	✓
OpenEdge Replication (Workgroup and Enterprise)	✓	✓	✓	✓	✓	✓
OpenEdge Repl Plus (Workgroup and Enterprise)	✓	✓	✓	✓	✓	✓
4GL Development System	✓	✓	✓	✓	✓	✓

Table 15: Supported 32-bit and 64-bit products by platform (2 of 2)

Component	HP-UX Itanium 64-bit	HP-UX PA- RISC 64-bit	AIX 64-bit	Linux Intel 64-bit	Linux Intel 32-bit	Solaris 64-bit
Query/ RESULTS	✓	✓	✓	✓	✓	✓
WebSpeed Messenger	✓	✓	✓	✓	✓	✓
NameServer	✓	✓	✓	✓	✓	✓
NameServer Load Balance	✓	✓	✓	✓	✓	✓
AppServer IntAdap	✓	✓	✓	✓	✓	✓
WSA	✓	✓	✓	✓	✓	✓
OpenEdge Management	✓	✓	✓	✓	✓	✓
SMNP Adapter	✓	✓	✓	✓	✓	✓
OpenEdge Transparent Data Encryption	✓	✓	✓	✓	✓	✓
OpenEdge Multi-tenant Tables	✓	✓	✓	✓	✓	✓

Disk space requirements

Table 16 lists the approximate disk space requirements for each OpenEdge product on Linux 32-bit and Unix 64-bit platforms. This is the size of the product as installed in your installation directory.

Table 16: Unix disk space requirements by product (1 of 2)

Product	Average size Linux 32-bit in MB	Average size Unix 64-bit in MB
OpenEdge Personal RDBMS	787	794
OpenEdge Workgroup RDBMS	787	794
OpenEdge Enterprise RDBMS	787	794
OpenEdge DataServer for Oracle	711	713
OpenEdge Development Server	1126	1126
OpenEdge Application Svr Basic	577	588
OpenEdge Application Svr Ent	627	639
OpenEdge Adap Sonic ESB	214	209
OpenEdge SQL Client Access	250	248
Client Networking	438	464
OpenEdge Replication	10	11
OpenEdge Repl Plus	10	11
4GL Development System	839	847
Query/RESULTS	400	406
WebSpeed Messenger	76	77
Mobile REST Mgt Agent	18	18
NameServer	355	351
NameServer Load Balance	135	129

Table 16: Unix disk space requirements by product (2 of 2)

Product	Average size Linux 32-bit in MB	Average size Unix 64-bit in MB
AppServer Int Adap	79	80
WSA	53	54
SNMP Adapter	4.4	4.5
OpenEdge TDE	135	129
OE Multi-tenant Tables	135	129
OpenEdge Management	487	483
OpenEdge BP Server	1433	1433
OpenEdge BP Modeler	559	548

Note: Because products may contain common components, your actual disk space requirements, will not precisely equal the sum of size of all the products you installed.

Licensing

When installing OpenEdge, the installation utility prompts you to enter product information from your *License Addendum*. The installation utility records the license information in the OpenEdge configuration file, `progress.cfg`.

Note: For information about using a License Addendum File, see the [“Obtaining an Electronic License Addendum file”](#) section on page 72.

The following syntax shows how to use the Show Configuration (`SHOWCFG`) utility to display the product license information for each OpenEdge product installed on your system:

Syntax

```
OpenEdge-install-dir/bin/showcfg OpenEdge-install-dir/progress.cfg
```

For more information about licensing, see [Chapter 6, “Administration Utilities.”](#)

OpenEdge Installation Prerequisites

This chapter presents prerequisite information and some preliminary tasks to perform before installing OpenEdge on any of the supported platforms, as described in the following sections:

- [Tasks overview](#)
- [Gathering information to plan your installation](#)
- [Determining your installation method](#)
- [Determining the type of installation](#)
- [Obtaining an Electronic License Addendum file](#)
- [Shared Network Installation utility](#)
- [Windows-specific installation considerations](#)
- [UNIX-specific installation considerations](#)
- [OpenEdge Replication](#)
- [WebSpeed configuration choices](#)
- [Accessing documentation and samples from the Progress Download Center](#)

Tasks overview

Complete the following preinstallation tasks before starting your OpenEdge installation:

- ☐ Gather product-related documents to make informed decisions about your OpenEdge installation before you begin. See the [“Gathering information to plan your installation”](#) section on page 69.
- ☐ Determine the installation method you plan to perform: online or silent. See the [“Determining your installation method”](#) section on page 70.
- ☐ Determine the type of installation you plan to perform: complete or custom. See the [“Determining the type of installation”](#) section on page 71.
- ☐ Review other required applications. See the [“Windows-specific installation considerations”](#) section on page 75, or the [“UNIX-specific installation considerations”](#) section on page 84.
- ☐ Save or upgrade an existing OpenEdge or Progress installation (if applicable). See the [“Saving an existing OpenEdge or Progress installation in Windows”](#) section on page 76 and the [“Upgrading an existing OpenEdge or Progress installation on UNIX platforms”](#) section on page 84.

Note: Tasks are considered common to all supported platforms unless otherwise specified.

Gathering information to plan your installation

Progress Software Corporation (PSC) provides documents to help you plan your installation. [Table 17](#) identifies and briefly defines these various documents.

Note: The OpenEdge 11.3 Installation Utility in Windows and on UNIX is available from Electronic Software Download (ESD).

Table 17: Preinstallation documentation resources

For this task ...	Go to ...	Which is available as ...
Determine and record product installation choices	Appendix A, "Preinstallation Checklist for Windows" or Appendix B, "Preinstallation Checklist for UNIX"	<ul style="list-style-type: none"> An appendix in this guide A PDF in ESD
Identify serial numbers and product control codes associated with products to install	<i>License Addendum</i>	<ul style="list-style-type: none"> A document downloadable from the Progress Download Center;¹ see the "Obtaining an Electronic License Addendum file" section on page 72 for more information
Identify product-related details that might not be in the OpenEdge 11.3 product documentation	<i>OpenEdge Release Notes</i>	<ul style="list-style-type: none"> A document downloadable from the Progress Download Center¹ An online document accessible from specific menu options in Windows and UNIX installation dialog boxes
Reference each product's components and subcomponents, as needed	Chapter 12, "OpenEdge Installation Products and Components in Windows" or Chapter 13, "OpenEdge Installation Products and Components on UNIX"	A chapter in this guide
Complete the checklist to prepare for the installation, and perform each procedure online during the installation process	Windows OpenEdge Installation online help and UNIX OpenEdge Installation online help	<ul style="list-style-type: none"> During either the Windows or UNIX installation process; choose Help or another specific Help control on an installation dialog box

1. The Progress Download Center is located at <http://www.progress.com/esd>. You must have a valid user name and password to download products from this site. Contact a Progress Customer Service Representative to set up your Download Center account.

Determining your installation method

You can install OpenEdge Release 11.3 products in Windows or on UNIX using the following methods:

- **Online, interactive installation** — This method prompts you to make installation choices online and record your input in dialog boxes. The dialog boxes appear programmatically as determined by the products you identify to install and the type of install you choose to perform. After you complete the Installation Utility, the Setup Utility initializes your choices, enabling you to use the products after the installation.

For details about loading your installation package and initiating either a Windows installation or a UNIX installation, see:

- [Chapter 4, “Performing an OpenEdge Installation in Windows”](#)
- [Chapter 5, “Performing an OpenEdge Installation on UNIX or Linux”](#)

Note: Online help is provided with each platform’s Installation Utility, and is accessible from **Help** or a specific **Help** control. The online help provides information and details the procedures required to complete each installation dialog box.

- **Silent Installation Utility** — The silent, or batch mode, installation does not prompt you to interactively enter your installation choices. A silent installation reads your installation values and settings as recorded in a response file. Using specific commands, you initiate your response file to run without user involvement. A silent installation supports either a complete or a custom installation.

For details about running silent or batch mode installations by creating a response file, see:

- The [“OpenEdge Silent installation overview”](#) section on page 123 for Windows.
- The [“OpenEdge Silent installation overview”](#) section on page 165 for UNIX.

Determining the type of installation

Your OpenEdge installation process depends on what products, components, and subcomponents you choose to install and the type of installation you plan to perform. [Table 18](#) summarizes the installation options.

Table 18: Installation options

Installation option	Purpose
Complete	Automatically installs all mandatory, recommended, and optional components and subcomponents of the OpenEdge products you are installing.
Custom	<p>Installs all mandatory product components and subcomponents, but allows you to selectively install the recommended and optional components and subcomponents on a product-by-product basis.</p> <p>This installation is recommended for more advanced users and provides the flexibility to distribute OpenEdge components on different machines, select product components to suit their business needs, and work around issues such as disk space limitations.</p> <p>Note: When customizing an install, Progress Software Corporation recommends that you consider removing only optional components and subcomponents. Removing recommended products might negatively affect a product's functionality.</p>

How product selection can affect your installation tasks

Some OpenEdge products that you install have additional installation dependencies, prompting you to perform additional set up or installation tasks as part of the OpenEdge installation process. The OpenEdge Installation Program automatically determines certain product dependencies and guides you through the completion of these tasks. The following list identifies ways the products you install affect your installation experience:

- If you are performing a Windows installation, OpenEdge 11.3 development products require Microsoft .NET Framework installed on the machine on which you are installing OpenEdge. If it is not already installed, you will be prompted to accept the software's installation at the conclusion of the Installation Program. For details, see the ["Required third-party applications"](#) section on page 46 and [Appendix A, "Preinstallation Checklist for Windows"](#).
- To complete an OpenEdge installation that includes Progress Dynamics®, the Installation Program automatically launches the Progress Dynamics Configuration Utility (DCU). For details about the DCU utility procedure, see the ["Completing the DCU wizard"](#) section on page 102.

To review a complete list of all OpenEdge products, and the components and subcomponents that comprise each, see [Chapter 12, "OpenEdge Installation Products and Components in Windows"](#) and [Chapter 13, "OpenEdge Installation Products and Components on UNIX"](#).

Obtaining an Electronic License Addendum file

An electronic License Addendum file contains the serial numbers and control codes for the OpenEdge license you purchased. An electronic License Addendum file eases the installation process by reading your serial number and control code information from the file, rather than being manually entered. Your electronic License Addendum file is accessible from the Progress Software Download Center.

Note: The *License Addendum File* is an electronic version of the *License Addendum* that shipped with your OpenEdge software.



To obtain an electronic License Addendum file for your OpenEdge software:

1. Access the Progress Software Download Center at <http://www.progress.com/esd> and log in.

Note: You must be a registered user to download the License Addendum file. Follow the links to create a user account, or retrieve a forgotten password if necessary. Contact CustomerService@progress.com for additional assistance.

2. Navigate to the License Addendum page for your OpenEdge release and platform:
 - a. Click **Download Software**. This brings you to the **Software Product List**.
 - b. From the listed products, select the product for which you are obtaining an Electronic License Addendum, for example Progress® OpenEdge®(with OpenEdge Replication). This brings you to the **Software Product Information** page.
 - c. From the **Software Product Information** page listing a suite of products. Click the product you want to download. This brings you to a page of releases.
 - d. Locate the release and click on its name in the **Description** column. This brings you to the **OpenEdge Product Downloads** page.
 - e. Click **Accept** to accept the PSC End User License Agreement. This brings you to the **Software Product Download** page.
3. On the **Software Product Download** page, click the **License** tab.

4. Position your mouse over the License Addendum area. Save the License Addendum by selecting the **File -> Save As** (or **Save Page As**) menu and changing the **Save As Type** to "**Web Page, HTML only**".
5. If you are saving multiple License Addendum files for future electronic installations, be sure to give each an unique name.

Notes: These instructions are verified for the following browsers: FireFox 2.0.0.16 and Internet Explorer 7 and above.

If you require assistance using the Download Center, you can use the Download Center help.

Once you save the License Addendum File locally, you can access it from the Serial Numbers and Control Codes dialog during the OpenEdge Installation.

Caution: Do not edit the License Addendum file. Opening the License Addendum file with software such as Microsoft Word can change the format of the file, making it unreadable by the installation process.

Shared Network Installation utility

In Windows, you can provide multiple-client access to a single copy of OpenEdge that is installed on a network-accessible drive (server). To initiate this shared installation arrangement, you must choose to install the Shared Network Installation component (NetSetup) during the installation process. Then, using the appropriate NetSetup deployment option, you can install the NetSetup regardless of whether you are performing a Complete or a Custom installation.

For more information about sharing an OpenEdge installation between a server and a client, see [Chapter 4, “Performing an OpenEdge Installation in Windows.”](#)

Windows-specific installation considerations

This section identifies the following Windows installation considerations:

- [OpenEdge working directory reminder](#)
- [Read-only .dll and .ocx files](#)
- [Required software to run OpenEdge products or components](#)
- [Saving an existing OpenEdge or Progress installation in Windows](#)
- [Reviewing the Windows installation directory structure](#)
- [Integrating OpenEdge with Windows Explorer](#)

OpenEdge working directory reminder

A *working directory* is a directory that contains your OpenEdge database and application files. The Installation Utility prompts you to create a working directory from which to run OpenEdge.

Caution: **Never** run OpenEdge products from the directory in which you installed them. If you do, you could damage the OpenEdge software files.

Read-only .dll and .ocx files

Before you install any OpenEdge products in Windows, check the directory `C:\Winnt\system32` or `C:\Windows\system32` to see whether any of the .dll or .ocx files have the read-only bit set. If any .dll or .ocx files in this directory are read-only, you must reset them before installing OpenEdge. If you try to install OpenEdge with one or more related .dll or .ocx files set to read-only, OpenEdge generates a dialog box informing you that you must reset the .dll bit or .ocx bit and reinstall OpenEdge.

Required software to run OpenEdge products or components

Some OpenEdge products and/or components depend on the presence of other software or software elements to run as designed. These elements might be required either before you perform an OpenEdge installation or concurrent with the OpenEdge install you perform.

Microsoft Internet Explorer

If you are installing a product that contains OpenEdge Management or OpenEdge Explorer or ProxyGen, you must have Microsoft Internet Explorer (MS IE) Version 4.01 or later installed on your system to use the graphical administrative tools. If you do not have MS IE Version 4.01 or later, you receive a warning message during the installation that tells you to install MS IE Version 4.01 or later. You can obtain information about acquiring or upgrading to MS IE Version 4.01 or later from the Microsoft Web site. You can continue with the installation after viewing this message, but neither ProxyGen nor the OpenEdge Management or OpenEdge Explorer graphical administrative tools will be functional.

Open Client Toolkit component

If you plan to install a product that contains ProxyGen, you might need to install and configure additional tools to allow the Open Client Proxy Generator (ProxyGen) to build proxies. For more information, see the chapter on configuration and deployment in [OpenEdge Development: Open Client Introduction and Programming](#).

OpenEdge SQL

The installation program does not automatically install the JDK component when you install any of these products:

- OpenEdge Enterprise RDBMS
- OpenEdge Workgroup RDBMS
- OpenEdge Workgroup RDBMS with OpenEdge SQL Client Access

If you intend to develop Java stored procedures and Java triggers for your database, you must install an OpenEdge development product such as Progress Developer Studio for OpenEdge. For information on writing Java stored procedures and triggers, see [OpenEdge Data Management: SQL Development](#) and [OpenEdge Data Management: SQL Reference](#).

OpenEdge SQL ODBC and JDBC Clients

The OpenEdge SQL ODBC and JDBC Clients are components of the OpenEdge Personal RDBMS, Workgroup RDBMS, and Enterprise RDBMS products. You can download them from the Progress Software Corporation Web site at <http://www.progress.com/esd>, using the OpenEdge SQL Client Access product.

Saving an existing OpenEdge or Progress installation in Windows

If you have an existing OpenEdge or Progress installation, you might want to save certain pieces of it to simplify configuring your new installation. If you want to continue using any templates, customized procedure or code files, or a `progress.ini` file, you must copy them to another location before you begin the new installation.

Progress Software Corporation strongly recommends that you thoroughly examine and review your existing installation before you make any changes. The tasks to plan and save a current installation will vary, depending on several factors.

Resources to help you plan and save your current installation

To help plan and implement the tasks required to save your current installation, consult the following online resources for OpenEdge database documentation and appropriate white papers available on the:

- PSC Web site at <http://www.progress.com>
- Progress Knowledge Center at <http://www.progress.com/en/support/index.html>
- Progress Software Developers Network (PSDN) at <http://communities.progress.com/pcom/docs/DOC-16074>

In addition to the online resources, you can contact Progress Technical Support for help with saving an existing OpenEdge or Progress installation.

Caution: Do not install different versions of OpenEdge into the same `OpenEdge-install-dir` directory.



To save an existing installation:

1. Copy any templates that you want to continue using to another location before installing OpenEdge.
2. Copy your current `progress.ini` file to a directory other than where you are installing OpenEdge if it contains information that you want to continue using.
3. Copy any customized procedure or code files in the directory where you are installing OpenEdge into a different directory.

For more information about saving previous versions of your `progress.ini`, customized procedures, or code files, see the [“Performing an OpenEdge Installation in Windows”](#) section on page 97.

4. If you have OpenEdge installed, and the `PROMSGS` environment variable is set on the **Environment** tab of the **System** settings in the Windows Control Panel, you must remove the `PROMSGS` environment variable before installing OpenEdge. If `PROMSGS` points to an old or nonexistent `PROMSGS` file, the InstallShield utility will not write all the necessary data to the Windows registry.

Caution: You must perform [Step 4](#) as described if your current installation meets the criteria defined. Otherwise, you will have unpredictable and undesirable results.

5. Truncate the before-image (`.bi`) file using the PROUTIL TRUNCATE BI utility and back up your existing database using the PROBKUP utility. For more information about these utilities, see [OpenEdge Data Management: Database Administration](#).

OpenEdge requires that your databases be converted to a multi-volume structure. If you were using single-volume databases with Progress Version 8 or Version 9, you must convert your OpenEdge databases to a multi-volume structure before converting the databases to OpenEdge. You must truncate your BI file before you convert it. If you plan to replace your current Progress Version files with OpenEdge 10, complete this step before you perform the installation to avoid erasing your current Progress Version files.

Caution: This conversion task involves many steps and requires that you plan each of them. To plan your steps, see the [“Resources to help you plan and save your current installation”](#) section on page 77.

Existing JavaSoft (InstallShield) JDK

If the required version of the Java Soft (InstallShield) JDK has been installed on your system before the OpenEdge Release 11.3 installation and you want to use this pre-existing JDK utility, you can. However, first you must complete the OpenEdge installation and then, as a postinstallation task, you must edit files tailored by the install to ensure that they point to this pre-existing JDK. Contact Progress Technical Support for assistance to perform this task.

OpenEdge automatic save of properties files

OpenEdge automatically makes copies of your `ubroker.properties`, `conmgr.properties`, and `proxygen.preferences` files and places them in a work directory. The new installation automatically upgrades the files in the `install-path\properties` directory. However, after you have finished your new installation, you must replace the newly installed versions of these files with these copies. When you start the AdminServer, your older files will be updated to match the current standards for these files. For information about the procedure to uninstall an existing OpenEdge product and instantiating the properties files, see [Chapter 5, “Performing an OpenEdge Installation on UNIX or Linux.”](#)

When you uninstall an existing OpenEdge product, the process copies the three files in the `install-path\properties` directory, to `%TEMP%\ubroker.properties`, `conmgr.properties`, and `proxygen.preferences`. After installing a new OpenEdge product, you can manually copy back the files from `%TEMP%`.

Reviewing the Windows installation directory structure

The OpenEdge installation `PATH` contains configuration files and several subdirectories. The installation `PATH` directory contains the OpenEdge executables, several procedure (`.p`) files, and other related files and subdirectories. The default OpenEdge installation `PATH` is `C:\Progress\OpenEdge`. However, during the OpenEdge Installation, you can choose a different location into which to install. Apart from this path, during installation.

References to the Windows installation directory in this guide

Throughout this guide, the installation `PATH` is referred to as one of the following:

- **DLC** — The `DLC` variable in Windows, `%DLC%`, is automatically set to your OpenEdge installation `PATH`. Historically, it has been a convenient way to refer to the location in which you have installed OpenEdge.

Note that the `%DLC%` variable is set in the various command scripts and in the registry; the variable is not, and should not, be set at the system level. For information about the `%DLC%` environment variable, see [Chapter 7, “Working in the OpenEdge Environment in Windows.”](#)

- **OpenEdge-install-dir** — A more explicit reference to the directory location to which your OpenEdge installation `PATH` occurs. The Windows environment variable `DLC` is also used to create this location; the use of the phrase `OpenEdge-install-dir` is intended to be more self-explanatory than is the reference `%DLC%`.

[Table 19](#) describes a directory tree of the OpenEdge subdirectories.

Table 19: OpenEdge-install-dir (%DLC%) directory structure (1 of 3)

Directory name	Description
auditing	Contains object (<code>.x</code>), development, and environment (ADM2) files for the Audit Policy Maintenance product
bin	Contains the executable files for OpenEdge, such as PRODB. It also contains batch files and system executables
certs	Contains the public keys of the Certificate Authorities (CAs) used by OpenEdge clients to perform server-side certificate validation when communicating with secure Web servers using HTTPS
dotnet	Contains support files to develop and deploy the .NET client
esbadapter	Contains the configuration and support code for the OpenEdge Adapter for Sonic ESB
gui	Contains object (<code>.x</code>), development, and environment (ADE) files for the OpenEdge graphical tools—these tools are compiled to run in graphical mode in Windows; they cannot run in a character environment
help	Includes the help and other necessary files for OpenEdge

Table 19: OpenEdge-install-dir (%DLC%) directory structure (2 of 3)

Directory name	Description
include	Contains C and C++ header files
install	Contains Java tailoring classes that only the Installation Utility uses. Also contains the automatically generated <code>response.ini</code> file used in an OpenEdge silent installation
java	Includes the Java files and executables necessary for running OpenEdge products
javahelp	Contains .jar files for the OpenEdge Application Debugger
jdk	Contains the Java Development Kit files and executables necessary for running OpenEdge products
jms	Contains files to support client deployment of java messaging
jre	Contains the Java Runtime Environment files and executables necessary for running OpenEdge products
keys	Contains encrypted RSA Private Key and Certificate file information
lib	Contains shared objects necessary for running OpenEdge executables
licenses	Contains license and copyright information related to HTTP Client, Open SSL toolkit, Perl, and w3c IPR software notice
netsetup	Contains files for the Shared Network Installation Utility
odbc	Includes files to support ODBC
oebuild	Includes files that the OEBUILD utility uses when creating custom executables
oeide	Contains the Eclipse environment, the plug-ins that comprise the Progress Developer Studio for OpenEdge product, and other related files
ora	Contains files to support the DataServer for ORACLE
perl	Contains files to support the use of the Perl scripting language
proedit	Contains files to support the advanced editing features
prohelp	Includes the online help and other necessary files for OpenEdge
prokey32	Contains files for international keyboard support for the 32-bit Windows Character Client
prolang	Contains the national language support directories
properties	Contains property files that manage the configuration of OpenEdge services, such as WebSpeed, the NameServer, and the AppServer
scripts	Contains files related to the Failover Cluster component

Table 19: OpenEdge-install-dir (%DLC%) directory structure (3 of 3)

Directory name	Description
servlets	Identifies the default location of the AppServer Internet Adapter (AIA) and Web Services Adapter (WSA) servlet containers—these containers include web definitions ¹
sonic	Contains files that support the Sonic client and container
sports	Includes the schema triggers and supplier information for each sample database
sports2000trgs	Includes the schema triggers for the Sports2000 database
src	Contains source files for OpenEdge ADE tools, such as the WebSpeed, Data Dictionary, Procedure Editor, and Sample Applications
templates	Contains files related to the Failover Cluster component
toolkit	Includes files that help in deploying and encrypting your applications
tty	Includes object (.x) files for character-mode OpenEdge
ubqmanager	Includes files used by the AppServer exclusively. Do not modify these files
wcadd	Contains Web Client images that include, among other files, the <code>setup.exe</code> to install the Web Client
webinstall	Contains several WebSpeed-related files, including samples, scripts, and help files
webspeed	Supports WebSpeed Workshop files such as samples, scripts, and help, that reside on the Web server

1. Refer to your OpenEdge product documentation for details about configuring WSA and AIA.

Reviewing the Eclipse installation directory structure

Progress Developer Studio for OpenEdge provides a common location to install an instance of the Eclipse framework for all Eclipse-based Progress products. This eliminates the overhead of installing separate instances of the Eclipse framework for each Eclipse-based product. The default `PATH` for the Eclipse framework and Progress Eclipse-based products is `C:\Progress\OpenEdge\oeide\eclipse`.

Eclipse folder contains files which support the Eclipse framework and Progress Eclipse-based products.

Integrating OpenEdge with Windows Explorer

In Windows, Microsoft allows applications to integrate certain features with Windows Explorer. Among those features, OpenEdge supports defined icons, shortcut menus, and property sheets for several of its file types. You can now easily perform an action

on a file or view detailed information about a file from Windows Explorer. This section presents:

- [OpenEdge file types](#)
- [Icons](#)
- [Shortcut menus](#)
- [Properties](#)

OpenEdge file types

OpenEdge supports defined icons and shortcut menus for the following file types:

- Procedure source code file (.p)
- Window procedure source code file (.w)
- Include file (.i)
- Parameter file (.pf)
- Configuration file (.cfg)
- Database file (.db)

OpenEdge also supports specific property information for these file types:

- Compiled procedure code file (.r)
- Database file (.db)

This information is stored in the registry, separate from your OpenEdge settings.

If another application has already registered a file extension that OpenEdge uses, the Installation Utility asks if you want to overwrite the information for that file extension. If you choose no, OpenEdge does not display the icon, shortcut menu options, or properties information for that file type. If you choose yes, OpenEdge replaces the icon, shortcut menu options, or properties associated with the file extension with OpenEdge-specific information.

The shell integration DLL uses the `DLC` and, optionally, the `PROMSGS` environment variables to locate the `PROMSGS` file. The DLL searches these registry locations for the following variables:

- `HKEY_CURRENT_USER\SOFTWARE\PSC\PROGRESS\11.3\Startup`
- `HKEY_LOCAL_MACHINE\SOFTWARE\PSC\PROGRESS\11.3\Startup`

The Installation Utility writes the proper values to the above registry locations. However, if after the installation you move OpenEdge to another location (or move or rename the `PROMSGS` file), you must edit the variables in the registry so that the shell integration DLL can find `PROMSGS`.

Icons

OpenEdge associates each of the OpenEdge-supported file types, except for compiled procedure code files (.r), with a unique icon that is displayed in Windows Explorer. You

can execute the default action on a file by double-clicking on its icon. To perform other actions, you can right-click on the file and choose one of the options from the shortcut menu. To change the default setting, see the “[Shortcut menus](#)” section on page 83.

Shortcut menus

A shortcut menu allows you to perform an action on a file by eliminating several steps to accomplish the task. OpenEdge enhances this feature by adding context-specific options for each file type. For example, to edit the Sports database from Windows Explorer, right-click the `sports.db` icon and choose **Edit in Data Dictionary** (single-user) from the shortcut menu. If you do not use the shortcut menu, this same action requires several more steps.

To view the shortcut menu for a specific file, right-click the file. A shortcut menu appears with context-specific options.

To add to or change your default shortcut menu options, choose **View→Options→File Types** from Windows Explorer. In the **Registered file types** list, choose the OpenEdge file type you want to modify and click **Edit**.

The command line for each shortcut menu option includes a full `PATH` to the OpenEdge executable. If you move this executable to another location, you must modify the `PATH`.

Properties

By default, Microsoft provides general information about a file in its properties sheet. OpenEdge adds an extra page containing specific information for compiled procedure code (`.r`) and database (`.db`) file types. To view a file's properties, right-click on the file and choose **Properties** from the shortcut menu.

UNIX-specific installation considerations

This section discusses the following UNIX-specific considerations:

- [JDK and JRE considerations](#)
- [Upgrading an existing OpenEdge or Progress installation on UNIX platforms](#)
- [Reviewing the UNIX system installation directory structure](#)

JDK and JRE considerations

On all the UNIX and Linux platforms OpenEdge automatically installs a JDK or JRE for you. For more information on the JRE and JDK requirements, see [Chapter 2, “UNIX Systems Installation Requirements.”](#)

Upgrading an existing OpenEdge or Progress installation on UNIX platforms

If you have OpenEdge or Progress installed, you can upgrade to the latest OpenEdge release.



To upgrade to the latest release of OpenEdge:

1. Make sure that the `ULIMIT` is set to at least 8MB and at least 128 file descriptors. For specific instructions on setting the `ULIMIT` on your system, consult the man page by typing `man ulimit` at the command prompt.
2. Truncate the before-image (.bi) file of any existing database using the `PROUTIL TRUNCATE BI` utility. Back up your OpenEdge database using the `PROBKUP` utility. For more information about the `PROUTIL TRUNCATE BI` and `PROBKUP` utilities, see [OpenEdge Data Management: Database Administration](#).
3. Make copies of your `ubroker.properties` and `conmgr.properties` files to another directory. The new installation automatically upgrades the files in the `OpenEdge-install-dir/properties` directory. (However, note that the full `PATH` to `DLC` is not updated automatically; you must edit it manually.) After your new installation is complete, replace the newly installed versions of these files with your copies. When you start the AdminServer, your older files will be updated to match the current standards for these files.
4. Make sure you are installing the software into a directory other than the directory from which you are running the Installation Utility.

Reviewing the UNIX system installation directory structure

The OpenEdge installation `PATH` contains configuration files and several subdirectories. The installation `PATH` directory contains the OpenEdge executables, several procedure (`.p`) files, and other related files and subdirectories. During installation, the `$DLC` environment variable is automatically set to your OpenEdge installation `PATH`.

References to the UNIX or Linux installation directory

Throughout this book, the installation `PATH` is referred to as either of the following:

- **DLC** — The `DLC` variable on UNIX or Linux, `$DLC`, is automatically set to your OpenEdge installation `PATH`. Historically, it has been a convenient way to refer to the location in which you have installed OpenEdge.

Note: Note that the `$DLC` variable is set in the various command scripts; the variable is not and should not be set at the system level. For information about the `$DLC` environment variable, see [Chapter 8, “Working in the OpenEdge Environment on UNIX.”](#)

- **OpenEdge-install-dir** — A more explicit phrase referring to the directory location to which your OpenEdge installation `PATH` occurs. The Unix environment variable `DLC` is also used to create this location; the use of the phrase `OpenEdge-install-dir` is intended to be more self-explanatory than is the reference `$DLC`.

[Table 20](#) describes a directory tree of the OpenEdge subdirectories.

Table 20: OpenEdge-install-dir (\$DLC) directory structure (1 of 3)

Directory name	Description
<code>bin</code>	Contains the executable files for OpenEdge, such as <code>PRODB</code> . It also contains batch files and system executables
<code>certs</code>	Contains the public keys of the Certificate Authorities (CAs) used by OpenEdge clients to perform server-side certificate validation when communicating with secure Web servers using HTTPS
<code>esbadapter</code>	Contains the configuration and support code for the OpenEdge Adapter for Sonic ESB
<code>include</code>	Contains the header files required for ESQ
<code>install</code>	Contains Java tailoring classes that only the Installation Utility uses; it also contains the uninstall script to remove an OpenEdge Release 11.3 installation
<code>java</code>	Includes the Java files and executables necessary for running OpenEdge products
<code>javahelp</code>	Contains <code>.jar</code> files for the OpenEdge Application Debugger

Table 20: OpenEdge-install-dir (\$DLC) directory structure (2 of 3)

Directory name	Description
jdk	Contains the Java Development Kit files and executables necessary for running OpenEdge products
jms	Contains files to support client deployment of Java messaging
jre	Contains the Java Run-time Environment files and executables necessary for running OpenEdge products
keys	Contains encrypted RSA Private Key and Certificate file information
lib	Contains shared objects necessary for running OpenEdge executables
licenses	Contains license and copyright information related to HTTP Client, OpenSSL toolkit, Perl, and w3c IPR software notice
odbc	Includes files to support ODBC
oebuild	Includes files that the OpenEdge <code>MAKE</code> utility uses when creating custom executables
ora	Contains files to support the DataServer for ORACLE
perl	Contains files to support the use of the Perl scripting language
proedit	Contains files to support the advanced editing features
prohelp	Includes the online help and other necessary files for OpenEdge
prolang	Contains the national language support directories
properties	Contains property files that manage the configuration of OpenEdge services, such as WebSpeed, the NameServer, and the AppServer
scripts	Contains files related to the Failover Cluster component
servlets	Identifies the default location of the AppServer Internet Adapter (AIA) and Web Services Adapter (WSA) servlet containers—these containers include Web definitions ¹
sonic	Contains files that support the Sonic client and container
sports	Includes the schema triggers and supplier information for each sample database
sports2000trgs	Includes the schema triggers for the Sports2000 database
src	Contains source files for OpenEdge ADE tools, such as the Data Dictionary, Procedure Editor, and Sample Applications
templates	Contains files related to the Failover Cluster component
toolkit	Includes files that help in deploying and encrypting your applications

Table 20: OpenEdge-install-dir (\$DLC) directory structure (3 of 3)

Directory name	Description
tty	Includes object (.r) files and r-code procedure libraries (.pl) for character-mode OpenEdge
ubqmanager	Includes files used by the AppServer exclusively. Do not modify these files
auditing	Contains object (.r), development, and environment (ADM2) files for the Audit Policy Maintenance product
oeide	Contains the Eclipse environment, the plug-ins that comprise the Progress Developer Studio for OpenEdge product, and other related files

1. Refer to your OpenEdge product documentation for details about configuring AIA and WSA.

OpenEdge Replication

OpenEdge Replication provides database replication so that customers have access to their database with minimal interruption.

OpenEdge Replication requires an OpenEdge Enterprise database or OpenEdge Release Workgroup database license for each local or remote OpenEdge database used in replication. You can replicate only from Enterprise to Enterprise database and Workgroup to Workgroup database.

For more information about OpenEdge Replication, see [OpenEdge Replication: User Guide](#).

How you install OpenEdge Replication depends on whether you are:

- [Installing OpenEdge Replication for the first time](#)
- [Upgrading an existing version of OpenEdge Replication](#)

Note: When you install OpenEdge Replication, the AdminServer is started before OpenEdge Replication is added to the AdminServer property files. This means that OpenEdge Replication will not be enabled until the next time the AdminServer is started. To enable OpenEdge Replication, simply stop the AdminServer and restart it.

Installing OpenEdge Replication for the first time

There are two primary requirements to consider before you install OpenEdge Replication for the first time:

- You should have a comprehensive backup plan in place for your database before you begin the installation.
- You must decide where to install the OpenEdge Replication components.

Upgrading an existing version of OpenEdge Replication

If you are upgrading an existing version of OpenEdge Replication, be sure to back up the following files:

- Source database
- Target database
- `database.repl.recovery` files

Because OpenEdge upgrades the `database.repl.recovery` files, once you upgrade to OpenEdge 11.3 you cannot return to versions of OpenEdge Replication prior to Release 11.3 using your OpenEdge 11.3 files. If you need to return to a previous version of OpenEdge Replication on the source machine, you must back up your source database, start after-imaging, and enable and configure OpenEdge Replication.

On the target machine, you must set up your structure file, restore your backup from the source, and enable and configure OpenEdge Replication. You can then run OpenEdge Replication by simply starting your source and target databases with an OpenEdge Replication qualifier.

OpenEdge Management or OpenEdge Explorer

OpenEdge Management is a browser-based management tool that you can use to monitor databases, files, networks, OpenEdge components, and system resources in an OpenEdge environment.

OpenEdge Explorer is also browser-based and allows you to set configuration properties for various OpenEdge resources, as well as to start and stop them, view their status, and view their log file data.

Installing OpenEdge Management or Explorer for the first time

There are several factors to consider before you install OpenEdge Management or OpenEdge Explorer for the first time. You should analyze what you need to configure or monitor before you begin the installation and you must decide where to install the OpenEdge Management or OpenEdge Explorer components.



To prepare to install OpenEdge Management or OpenEdge Explorer:

1. Determine the names and locations of the resources that you need to monitor and the properties you want to configure. You can configure properties for resources associated with local and remote AdminServers. With OpenEdge Management, you can also monitor certain resources running under a local or remote AdminServer.
2. (In OpenEdge Management only) Determine whether to save monitoring information to the OpenEdge Management Trend Database and, when saving the monitoring information, decide where to locate the database.

The OpenEdge Management Trend Database stores the monitoring information that OpenEdge Management collects for databases, system resources, file resource, network resources, the AppServer, WebSpeed Transaction Server, and the NameServer. During configuration, you can choose whether to save monitoring information locally, remotely, or not at all. Before installation, you should decide if you want to save this data and where you want to save it.

OpenEdge Management automatically creates the OpenEdge Management Trend Database if you have an OpenEdge Enterprise RDBMS, an OpenEdge Workgroup RDBMS, or an OpenEdge Personal database installed on the same machine where you are installing OpenEdge Management.

If you decide to save monitoring information remotely, the remote machine must have both a database (Enterprise or Workgroup) and OpenEdge Management installed. In other words, you cannot just copy a trending database to a remote machine.

The local instance of OpenEdge Management needs to communicate with a remote instance of OpenEdge Management to use the remote trending database.

3. (In OpenEdge Management only) Determine how monitoring might affect system performance.

The more resources you monitor, the more system resources OpenEdge Management uses. If you plan to monitor a large number of database servers and network services in your configuration, you might want to consider configuring additional OpenEdge Management instances, both locally and remotely.

4. Determine where to install OpenEdge Management or OpenEdge Explorer.

Based on the decisions you made in Steps 1 through 4, you can install OpenEdge Management or OpenEdge Explorer either locally or on a separate or standalone machine.

See [OpenEdge Management and OpenEdge Explorer: Configuration](#) for more information.

System requirements

Most of the system requirements for OpenEdge Management or OpenEdge Explorer are the same as those for OpenEdge.

Product support

To use all of OpenEdge Explorer's features, you must have a database or server/broker installation.

To use all of OpenEdge Management's features, you must install products that support the following functionality:

- The AdminServer
- A Workgroup or Enterprise database, to allow trending of OpenEdge Management data
- A client networking license, to allow OpenEdge Management to run standard jobs and reports

Browser support

A Web browser is required to run the OpenEdge Management or OpenEdge Explorer graphical user interface known as the management console. Although you might find other browsers that you can use with OpenEdge Management or OpenEdge Explorer, the following browsers are supported in Windows platforms:

- Firefox
- Firefox Opera
- Internet Explorer

On UNIX platforms, the following browsers are supported:

- Mozilla
- Firefox

Support for multiple Eclipse frameworks

Progress Developer Studio for OpenEdge is developed using the Eclipse development framework. When installing Progress Developer Studio for OpenEdge, the installation process provides the option to integrate the Progress Developer Studio for OpenEdge plug-in files into a previously installed Eclipse development environment. This is in addition to the default OpenEdge environment.

If you choose to integrate the Progress Developer Studio for OpenEdge plug-ins into an additional environment, the installation process prompts you to supply the path to the additional framework and verifies it is a valid destination. Validation consists of two checks:

- The specified framework directory contains the file `eclipse.exe`
- The framework version is 3.7.1 and 3.8.0.
- When you install 32-bit Windows installer for OpenEdge, you must use 32-bit Eclipse. When you install 64-bit Windows installer for OpenEdge, you must use 64-bit Eclipse.

If the validation fails, the installation issues a warning message. If the target is an unsupported version of the framework, the message contains both the supported versions and the version detected. If the target is not a valid framework directory, user is given a warning message and the option to choose another path.

Integration after installation

The Progress Developer Studio for OpenEdge plug-ins can be integrated into an additional Eclipse framework after the OpenEdge install has completed with the use of an integration script, `<openedge-install-dir>\oeide\integrateArchitect.bat`.

Integrate the OpenEdge plug-ins to an additional framework as shown:

```
proenv> cd <openedge-install-dir>\oeide
proenv> integrateArchitect.bat -install <path-to-target-eclipse>
```

If the target Eclipse framework is invalid (either not a valid Eclipse location or invalid Eclipse version) the script does nothing, and exits. If the script executes successfully, the script integrates the Progress Developer Studio for OpenEdge plug-ins into the specified eclipse location, and the location is stored for references by uninstall and service pack updates.

Service pack updates to plug-ins

Service pack updates to OpenEdge plug-ins will be propagated to both the OpenEdge install, and all locations where the plug-ins have been integrated.

Uninstall

When Progress Developer Studio for OpenEdge is uninstalled, all Progress Developer Studio for OpenEdge plug-ins integrated into other Eclipse environments are also

uninstalled. This includes frameworks identified during installation, or specified at a later time with the integration script.

WebSpeed configuration choices

WebSpeed is an OpenEdge component to develop and deploy Web browser-based online transaction processing (OLTP) business applications. WebSpeed products require a Web server product for which you must configure a Web server directory on a machine that you want these products to reside. If you are upgrading an existing WebSpeed installation, shut down your Web server and reboot your machine. For information about shutting down a Web server and examples of various WebSpeed configurations, see [OpenEdge Getting Started: WebSpeed Essentials](#).

Developing Web applications with WebSpeed

These OpenEdge 11.3 products support developing Web applications with WebSpeed:

- OpenEdge Studio (includes WebSpeed Workshop and Progress Dynamics®)
- WebSpeed Workshop
- Progress Developer Studio for OpenEdge (includes the WebSpeed Workshop)
- OpenEdge Development Server (includes WebSpeed Transaction Server)

Deploying Web applications with WebSpeed

These OpenEdge 11.3 products support deploying Web applications with WebSpeed:

- OpenEdge Application Server Basic (includes WebSpeed Transaction Server)
- OpenEdge Application Server Enterprise (includes WebSpeed Transaction Server)
- WebSpeed Messenger (which is a component of the Application Server products)

To choose the proper configuration to install, you need to consider your network resources and whether you want to create a development, a deployment, or a combined development and deployment configuration. You can distribute the components required to run WebSpeed in many ways. [OpenEdge Getting Started: WebSpeed Essentials](#) describes the WebSpeed components and how they work together.

Accessing product documentation and samples

You can download the OpenEdge Documentation and Samples of your OpenEdge Release 11.3 from one of the following locations:

- Progress Software Corporation Web site at:
<http://communities.progress.com/pcom/docs/DOC-16074>.

Note: Go to the Web site and click the appropriate Production Documentation link for the required release to download a self-extracting Documentation and Samples file.

- Progress Download Center at: <http://www.progress.com/esd>.

Note: To download OpenEdge Documentation, you require a valid account that your company obtained from Progress Software Corporation to access OpenEdge products and updates.

This section outlines the procedures to access the documentation and sample information in Windows and on UNIX from the Progress Download Center.

Accessing documentation and samples from the Progress Download Center



To obtain the Documentation and Samples for your OpenEdge software from the Progress Download Center:

1. Access the Progress Software Download Center at <http://www.progress.com/esd> and log in.

Note: You must be a registered user to download the OpenEdge Documentation and Samples. Follow the links to create a user account, or retrieve a forgotten password if necessary. Contact CustomerService@progress.com for additional assistance.

2. Navigate to the **Software Product Download** page for your OpenEdge release and platform:
 - a. Click **Download Software**. This brings you to the **Software Product List**.
 - b. From the listed products, select the product for which you are obtaining an Electronic License Addendum, for example Progress® OpenEdge®(with OpenEdge Replication). This brings you to the **Software Product Information** page.
 - c. From the **Software Product Information** page listing a suite of products. Click the product you want to download. This brings you to a page of releases.

- d. Locate the release and click on its name in the **Description** column. This brings you to the **OpenEdge Product Downloads** page.
 - e. Click **Accept** to accept the PSC End User License Agreement. This brings you to the **Software Product Download** page.
 3. On the **Software Product Download** page, locate the **Documentation** section, and Click the **HTTPS Download/Download Manager** button. You are prompted to start the download.
 4. Click **Yes** to save the zip file to your system. Extract the zip file and locate the document you require.

Notes: These instructions are verified for the following browsers: FireFox 2.0.0.16 and Internet Explorer 7 and above.

If you require assistance using the Download Center, you can use the Download Center help.

Performing an OpenEdge Installation in Windows

This chapter contains instructions for installing OpenEdge in Windows, as described in the following sections:

- [Installation overview](#)
- [Running the Progress Dynamics Configuration Utility](#)
- [Additional product installation activities](#)
- [OpenEdge Silent installation overview](#)
- [Performing postinstallation tasks](#)
- [Uninstalling OpenEdge in Windows](#)
- [Sharing an OpenEdge installation on a network overview](#)
- [Uninstalling the Shared Network Installation Utility](#)
- [Running the Silent installation option for the Shared Network Installation Utility](#)

Installation overview

After you have completed the tasks described in the “[Tasks overview](#)” section on page 68, you are ready to perform the OpenEdge installation in Windows.

Loading the installation media

To load the installation media, you must have Administrator privileges on the machine where you are installing OpenEdge. For more information, see your Windows documentation.



To initiate the Installation Utility to install OpenEdge products:

1. Obtain a copy of the completed *Preinstallation Checklist for Windows*. You should also have the other installation-related documents highlighted in [Table 17](#) of the “[Gathering information to plan your installation](#)” section on page 69 available for reference.

2. Close all other applications before beginning the installation process.

Other applications or tasks might interfere with the installation or use files that OpenEdge needs to complete the installation. Shut down any processes where the executable itself, or a file used by the executable, is located in the directory where you intend to install OpenEdge.

3. Launch the installation program from the installation medium you plan to use, as described in the following table:

For this installation medium ...	Do the following ...
Electronic Software Distribution (ESD) download	Navigate to the software image you intend to download from the Progress Software Download Center. ¹

1. The Progress Software Download Center is available at <http://www.progress.com/esd>. Access to Progress software products and updates at this Web site requires a valid account.

Performing the installation

Once you have loaded the installation program from your installation medium, you are ready to perform the online tasks required to install OpenEdge.

Refer to [Table 17](#) for the documents you should reference during installation to help you perform the online OpenEdge installation.

Also, refer to the online installation help system that contains a help topic for each installation dialog box. To access the online help while you are running the Installation Utility:

- Choose **Help** on an installation dialog box. The help topic associated with the dialog box appears and describes the step-by-step procedure required to complete the dialog box.
- Choose help topics that display in the help system's Table of Contents. Note that the help viewer in which you can read an individual help topic also displays the help system's Table of Contents in the left pane. Use the Table of Contents to navigate through all the online installation-related help topics. To display the Table of Contents, click **Show** on the Navigator bar.

Finishing the installation

If you saved information in a `progress.ini` file in a previous version of OpenEdge and you want to continue using it, you can add that information to the new `progress.ini` file. Perform the following procedure whether you installed using the online, interactive method or the silent method.



To add information about your `progress.ini` file:

1. Copy the information you want to save from your previous `progress.ini` file to the OpenEdge `progress.ini` file.
2. If you are copying information from a Version 9 installation, rename any `PROUIB` section to `PROAB`. The `PROUIB` section of the Version 9 `progress.ini` file is referred to as `PROAB` in OpenEdge.
3. Run `ini2reg` to update the information in the registry with the information you added from your previous `progress.ini` file.
4. Restart your system.

Using property information from a previous installation

If you want to continue using the property information (such as `ubroker.properties`, `conmgr.properties`, or `proxygen.preferences`) that OpenEdge automatically saved prior to the current installation, copy the saved property files from `%TEMP%\OpenEdge-install-dir\properties`. For information about the automatic save of the property information before the installation process occurs, see the [“OpenEdge automatic save of properties files”](#) section on page 78. For information on merging property files, see the [“Mergeprop utility overview”](#) section on page 291.

You will also need to perform any other postinstallation tasks as discussed in the [“Post Installation Considerations”](#) section on page 100 and the [“Performing postinstallation tasks”](#) section on page 137.

Post Installation Considerations

Note these points after you have performed the Installation Utility:

- If you installed a product that includes Progress Dynamics, you must run the Progress Dynamics Configuration Utility (DCU) as a post installation task. See the [“Running the Progress Dynamics Configuration Utility”](#) section on page 101.
- Detailed product information about the OpenEdge products you installed is available in the Release 11.3 product documentation set. Access the PDF-formatted documentation set from the Progress Software Corporation Web site at: <http://communities.progress.com/pcom/docs/D0C-16074>.

Running the Progress Dynamics Configuration Utility

To complete the Progress Dynamics® installation, OpenEdge provides the Progress Dynamics Configuration Utility (DCU). The DCU is an Advanced Business Language program.

The DCU completes the installation by building a new `icfdb` Repository database or by upgrading an existing one from a previous release. This section includes the following:

- [Before you begin](#)
- [Completing the DCU wizard](#)
- [Editing Progress Dynamics files](#)

Before you begin

Before you perform the procedures described in this section, note the following:

- The DCU completes the installation by building a new `icfdb` Repository database or by upgrading an existing one from a previous release. (Consult the *Release Notes* for the most specific details about upgrading to the latest Progress Dynamics release; this step is most important to users who are upgrading from earlier versions of the DCU.)
- The DCU does not require ABL to run. You can use the DCU to deploy Progress Dynamics to client sites that do not have the compiler installed.
- The DCU launches directly after the OpenEdge installation completes provided the following conditions are met:
 - You are installing Progress Dynamics as a component of either OpenEdge Studio or Progress Developer Studio for OpenEdge.

Note: To install Progress Dynamics as a component of Progress Developer Studio for OpenEdge, you must select the component on the **Configuring / Installing Components** dialog box. The Progress Dynamics component supports the AppBuilder functionality within Progress Developer Studio for OpenEdge. Therefore, you must select the **Progress Dynamics** option on this dialog box to enable it in Progress Developer Studio for OpenEdge. For more information, see [Appendix A, “Preinstallation Checklist for Windows.”](#)

- During the OpenEdge installation, you select the **Install/upgrade Dynamics repository** option on the **Progress Dynamics Options** dialog box.

After you choose **Finish** in the InstallShield's **Complete Setup Done** dialog box, an OpenEdge session starts up. Then, DCU wizard starts by displaying the **Progress Dynamics Configuration Utility - Welcome** dialog box.

The DCU performs its work in a progressive and re-entrant fashion. If for any reason the DCU does not complete its work, or if you quit the utility before it finalizes, you can rerun it (from the command line or a shortcut) to complete its work. For more information about the DCU, see [OpenEdge Development: Progress Dynamics Administration](#).

Note: The DCU does not remove any part of a Progress Dynamics installation.

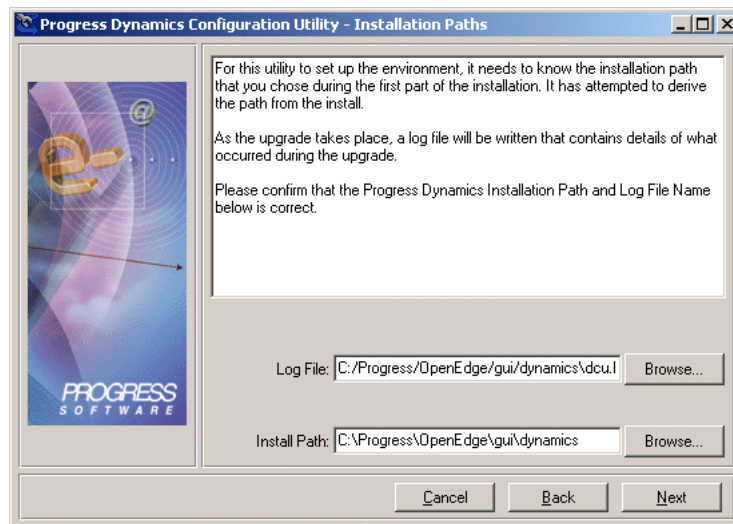
Completing the DCU wizard

This section presents the procedures to complete the Progress Dynamics installation using the Progress Dynamics Configuration Utility (DCU).



To complete the DCU wizard:

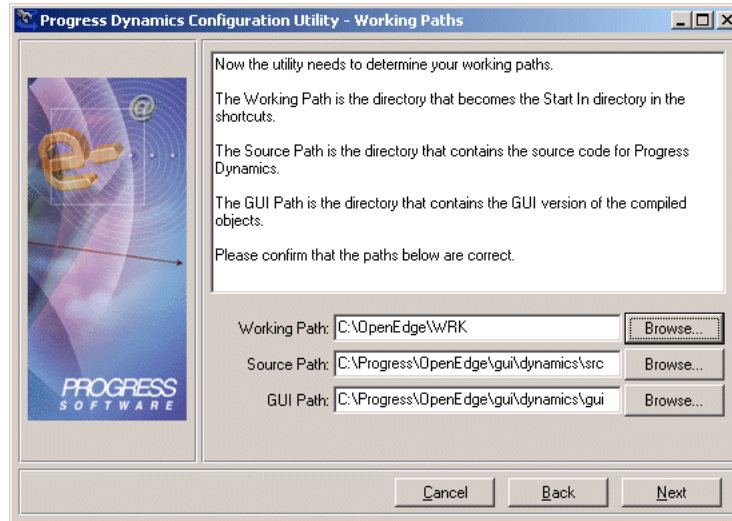
1. When the DCU automatically launches after the installation, choose **Next**. The **Installation Paths** dialog box appears:



2. Enter the path for the DCU log file. The **Log File** field contains the default pathname.
3. Enter the path of the directory where Progress Dynamics is installed.

The **Install Path** field contains a default entry based on the information you provided when you installed Progress Dynamics.

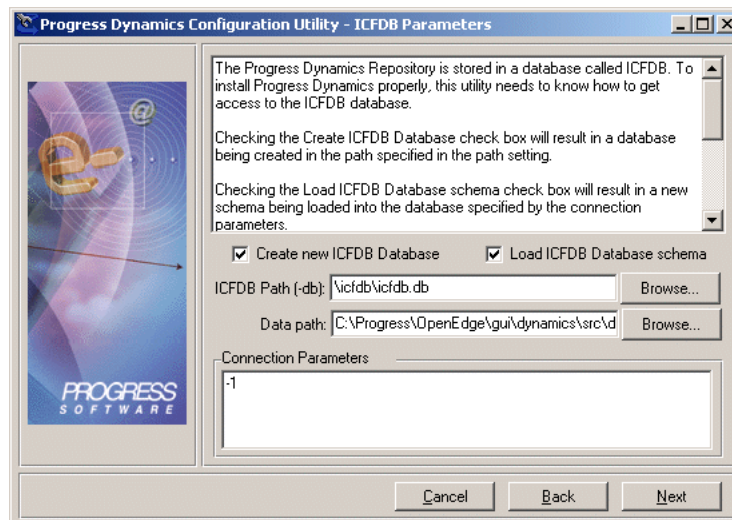
4. Choose **Next**. The **Working Paths** dialog box appears:



5. Enter the path information in the **Working Path**, **Source Path**, and **GUI Path** fields. The fields contain default entries based on the information you provided when you installed OpenEdge.

Note: The **Working Path** must **not** be under your OpenEdge installation directory. Otherwise, you can lose all of your work during future installs and upgrades of these products. Enter the path of your `icfdb` database and the database connection parameters.

6. After you enter the appropriate path information, choose **Next**. The **ICFDB Parameters** dialog box appears:



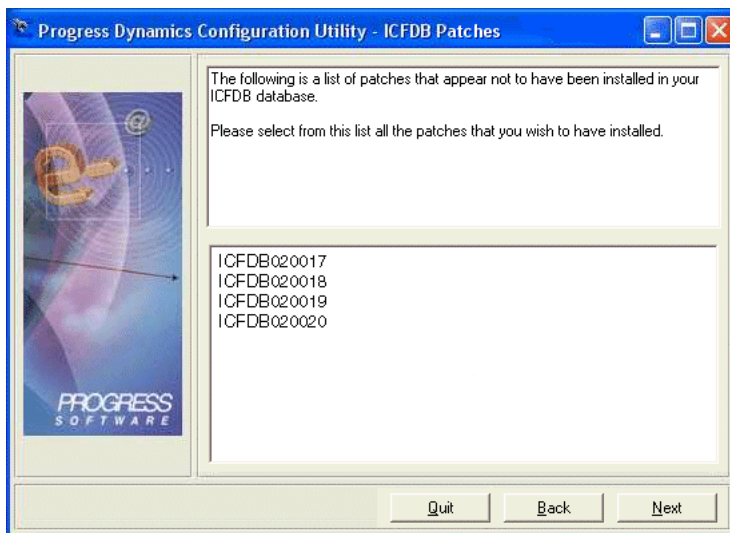
For more information about connection parameters, see [OpenEdge Deployment: Startup Command and Parameter Reference](#).

If you try to use the **Create new ICFDB Database** option to create a database on a remote machine, an error message appears. To load the `icfdb` schema to a remote database, you first need to create the database on the remote machine by starting a client and running the PRODB utility. For more information about the PRODB utility, see [OpenEdge Data Management: Database Administration](#).

Caution: If you are upgrading, you must remove the check from both the **Create new ICFDB Database** and the **Load ICFDB Database schema** options.

If you are upgrading, note the following:

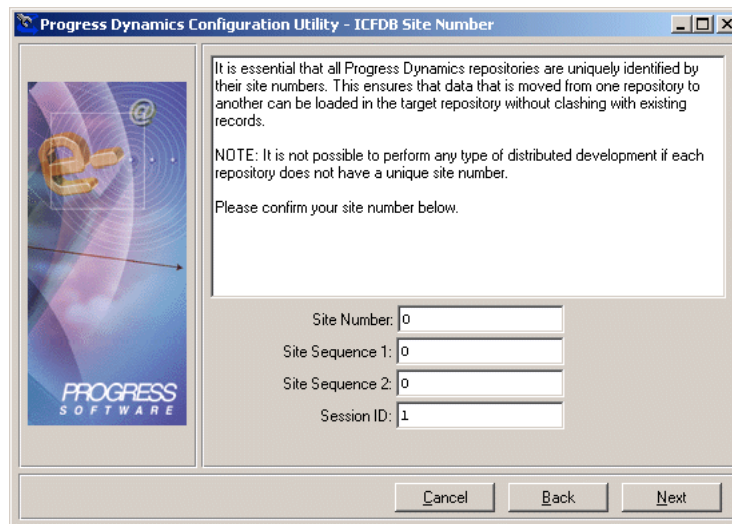
- If you check only the **Create new ICFDB Database** option, the DCU automatically loads the database schema as well. The DCU also deletes any database that already exists in the specified `icfdb` path and creates a new one.
 - If you check only the **Load ICFDB Database schema** option, the DCU treats any existing `icfdb` as if it was empty and loads the schema.
7. After you enter the appropriate information, choose **Next**. If you are upgrading, the **ICFDB Patches** dialog box appears:



If this is a new installation, the **ICFDB Patches** dialog box does not appear and you can proceed to [Step 10](#).

8. Review the patch information. The DCU lists which patches are needed to upgrade your database to the latest level. The list will vary depending on what patch level was added to the previous version of your `icfdb` database. The DCU applies the patches you select in the correct order.

9. After you review the patch information, choose **Next**. The **ICFDB Site Number** dialog box appears:

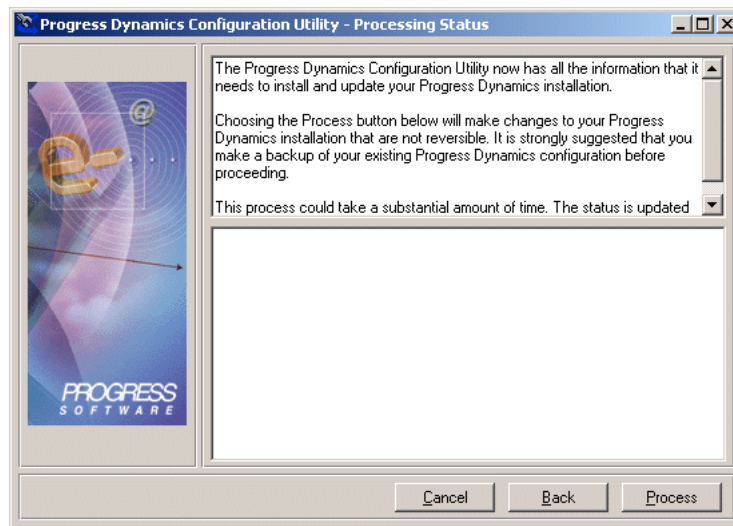


10. Enter appropriate values in the **Site Number**, **Site Sequence 1**, **Site Sequence 2**, and **Session ID** fields. For more information, see [OpenEdge Development: Progress Dynamics Administration](#).

If you are upgrading, the DCU transfers sequence values from the previous version of the Repository. Or you can obtain these values from the **Set Site Number** dialog box. You can access the **Set Site Number** dialog box from the Progress Dynamics AppBuilder **Build** menu.

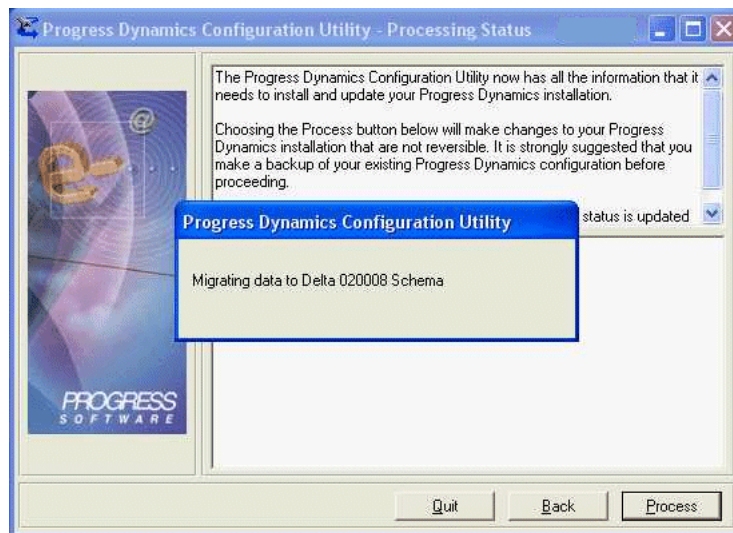
Note: Verify that the **Site Sequence 1** value is increased when you are upgrading. If the **Site Sequence 1** value in this installation is less than the value in the previous release, you will get error messages stating that an Object ID is already used.

11. After you enter the appropriate values, choose **Next**. The **Processing Status** dialog box appears:



Caution: If you are upgrading, this is the time to make a backup of your existing Progress Dynamics configuration because you might want to refer back to your prior configuration. The DCU processing that occurs next is irreversible. You will lose configuration information if you do not have a backup before running the DCU.

12. Choose **Process** to start the DCU. The **Processing Status** dialog box appears:



The appearance of this dialog box indicates which upgrade program is currently running.

13. Click **Finish** when the DCU completes processing.

Editing Progress Dynamics files

Typical of installations that use Progress Dynamics, you want to connect to the Progress Dynamics Repository database over a network. This type of configuration requires that you edit your `services` file to identify the service name and a port number for the `icfdb` Repository database.

The first time you install Progress Dynamics, assign a port number to the Repository Database server in the Windows `services` file. Subsequent OpenEdge uninstalls do not remove this entry. Therefore, you need to perform this task once.



To edit your `services` file:

1. In a text editor, open your `services` file. By default, the `services` file is in the `C:\WINNT\System32\drivers\etc\directory`. (In Windows XP Professional, the directory is `C:\WINDOWS\System32\drivers\etc\`.)
2. Assign port numbers, using the following format:

```
service_name    port_number/tcp
```

The `service_name` is the name you specify with the `-s` parameter when you start the database. The `port_number` is a unique four-digit number (one that is not already assigned to another service in the file). For example:

```
icfdb    8000/tcp
```

3. Add an additional line for each of your application databases, using a unique service name and a unique port number for each one.
4. Save and close the `services` file.

Caution: You cannot start two Repositories if both have the same service name (`icfdb` for example). If you need to run more than one Repository database, each version must have a different service name and a different port number.

Editing installed files

Table 21 lists files installed by the Progress Dynamics installer program that you can edit to modify configuration settings, add databases, or change paths.

Table 21: Progress Dynamics files that you can edit

Filename	Description	Location
icfconfig.xml	An XML file containing the instructions used by the Progress Dynamics Configuration File Manager to start up a specified Progress Dynamics session (for more information, see OpenEdge Development: Progress Dynamics Administration)	<i>OpenEdge-install-dir\gui\dynamics</i> Default: C:\Progress\OpenEdge\gui\dynamics
startdbs.bat	Starts database servers for the Progress Dynamics Repository by invoking <code>proserve</code>	<i>OpenEdge-install-dir\bin</i> Default: C:\Progress\OpenEdge\bin
stopdbs.bat	Stops database servers for the Progress Dynamics Repository by invoking <code>proshut</code>	
icf.ini	The Progress Dynamics initialization file (the Progress Dynamics version of the <code>progress.ini</code> file)	
icf.pf	The Progress Dynamics AppBuilder startup parameter file	<i>OpenEdge-install-dir</i> Default: C:\Progress\OpenEdge

If you are upgrading from an earlier version of Progress Dynamics, there are several tasks that you need to perform. To ensure that your existing applications run under the newer release, review and complete the tasks described in the following sections:

- [Editing the Progress Dynamics XML configuration file](#)
- [Starting a development session](#)
- [Stopping and restarting Progress Dynamics](#)
- [Updating session types](#)
- [Running the Entity Import tool](#)
- [Recompiling application code](#)
- [Setting up for Web development](#)

Editing the Progress Dynamics XML configuration file

The DCU upgrade process is not complete at this point. The DCU must run again to apply other upgrade procedures and to update data sets for the newly created `icfdb`. The DCU automatically runs again when you start an administrative session. But before you can log in and start an administrative session, you must have an XML configuration file that is compatible with the newer version of Progress Dynamics.

If you did not create a new `icfconfig.xml` file (or edit the default) for your application, you can skip this section. You can simply use the standard `OpenEdge-install-dir\gui\dynamics\icfconfig.xml` file that ships with Progress Dynamics to access the upgraded Repository.

However, if you modified or changed the name of the default XML configuration file (for example, you might have a customized XML configuration file in your current directory), you must edit the XML configuration file to make it compatible with your upgraded application. For example, you must add service entries for the new managers, and change the connection parameters for your `icfdb` database.

You should only edit the `icfconfig.xml` file for the session type that you use as your administrative session type. This allows you to connect to your Repository with administration privileges. Then you can use the Dynamics Administration tool's **Session** menu options to modify your other session types and regenerate the `icfconfig.xml` file.



To edit your `icfconfig.xml` file:

1. Create a backup copy of `icfconfig.xml` before you edit it.

Caution: If you make a mistake in the following steps, you might render your `icfconfig.xml` file unreadable and your Progress Dynamics session might not start. Therefore, creating a backup copy that you can revert to is extremely important.

2. Open your `icfconfig.xml` file in a simple text editor, such as Notepad.
3. Search for the string: `SessionType=ICFDev` where `ICFDev` is the name of the session type that you use for administration tasks. This string should occur inside a session node as follows:

```
<session SessionType=ICFDev>
```

Immediately following the `<session>` node is a `<properties>` node.

4. Scan down the file until you pass the end of the properties node, which is denoted by the end properties tag (`</properties>`).

Immediately following the `<properties>` node is a `<services>` node. The `<services>` node contains the list of services that should be connected when the session starts. Among them is a `<service>` node for each of the databases and AppServers that are connected to this session. Each `</service>` node is contained within a start `<service>` and end `</service>` tag.

5. Search for a `<cServiceName>` tag with the value `rvdb`.

If it exists, remove the entire `<service>` node for the `RVDB` service type from the file.

The `rvdb` database was used in Progress Dynamics Version 1.1A. It became obsolete in Version 2.0A. There should be no references to it in any of your configuration files.

6. Scan down until you find the first end managers (`</managers>`) tag.
7. Insert the following XML statements immediately before the line noted in [Step 6](#) of this procedure:

```
<manager>
<cManagerName>RIManager</cManagerName>
<cFileName>ry/app/ryrisrvrp.p</cFileName>
<cHandleName>RI</cHandleName>
<cSuperOf/>
</manager>
<manager>
<cManagerName>CustomizationManager</cManagerName>
<cFileName>ry/app/rycusrvrp.p</cFileName>
<cHandleName>NON</cHandleName>
<cSuperOf/>
</manager>
<manager>
<cManagerName>RepositoryDesignManager</cManagerName>
<cFileName>ry/app/rydessrvrp.p</cFileName>
<cHandleName>NON</cHandleName>
<cSuperOf/>
</manager>
```

8. Scan down until you find a `</service>` node that contains a `<cServiceName>` tag with the value `icfdb`.
9. Change the database connection parameters from the values for the earlier version to the appropriate values for the newer version.

The arguments for the `-db` and the `-S` parameters should be the `icfdb` that you upgraded through the DCU.

The bold text in the following example shows the changes to the `icfdb` services entry:

```
<service>
<cServiceType>Database</cServiceType>
<cServiceName>ICFDB</cServiceName>
<cPhysicalService>ICFDBn</cPhysicalService>
<cConnectParams>-db icfdbv21A -N TCP -H localhost
-S icfdbv21A</cConnectParams>
<lDefaultService></cConnectParams>
<lCanRunLocal></lCanRunLocal>
<iStartOrder></iStartOrder>
</service>
```

Since the upgrade simultaneously opens a large number of records, it is possible that you might get an error stating that the record lock table is too small. In that case, you must set the Lock Table Entries parameter (`-L`) to a very large value. (A value of 500,000 should be adequate.) See [OpenEdge Deployment: Startup Command and Parameter Reference](#) for more information.

10. Save the edited `icfconfig.xml` file.
11. Place the `icfconfig.xml` file in a directory that is included in your `PROPATH`.

Starting a development session

The DCU must run again to apply other upgrade procedures and to update data sets for the newly created `icfdb`. The DCU automatically runs again when you start a session that has administrative privileges. By default, you can run the development session (`ICFdev`) with administrative privileges. When the DCU ran the first time, it wrote information to the Repository, which it now uses to complete the upgrade.

To update data sets, Progress Dynamics applies Application Dynamic Object (ADO) files to the Repository. ADO files are XML documents that have a `.ado` filename extension.

After you complete the tasks described in the [“Editing the Progress Dynamics XML configuration file”](#) section on page 109, you should be able to start a session that connects to the `icfdb` that the DCU upgraded.



To start an administrative session:

1. Start the DB servers for the newer version of Progress Dynamics, if they are not already running.
2. Start an administrative session, logging in as **admin**.

Note: No password is required.

If you start a Progress Dynamics Development session from a desktop shortcut, check the properties to make sure that the value of `ICFSESSTYPE` is set to your administrative session type (usually `ICFDev`). Also, verify that the `-ini` and `-pf` parameters point to the Release 11.3 initialization and startup parameter files. For example:

```
C:\Progresss\OpenEdge\bin\prowin32.exe -p icfstart.p
-pf "C:\Progress\OpenEdge\icf.pf"
-ini "C:\Progress\OpenEdge\bin\icf.ini"
-icfparam ICFSESSTYPE=ICFDev
```

After you log in, the DCU starts, runs the upgrade programs, and applies ADOs. The DCU displays a status window that indicates its **progress**.

This phase of the upgrade involves running a large number of procedures, and it can be very time consuming. The actual duration depends on the size and complexity of your application.

Stopping and restarting Progress Dynamics

After the DCU loads all of the ADOs, you will be in a Dynamics AppBuilder session. Before you can continue, you must apply the changes you made up to this point by stopping and restarting Progress Dynamics. Log out of the current session. When you log in again, you must start an administrative session, as described in the [“Starting a development session”](#) section on page 111.

Updating session types

In the [“Editing the Progress Dynamics XML configuration file”](#) section on page 109, you updated the session that you use to administer your Repository. You must now update any other session types that apply to your application.

From the **Dynamics Administration Tool Session** menu, select the **Session Type Control** and modify your existing session types. After modifying your session types, regenerate the XML configuration file. (You can also edit the configuration file manually, but manual editing is more error prone.) See [OpenEdge Development: Progress Dynamics Administration](#) for more information about defining, modifying, and managing sessions.

Note: If you did **not** create a new `icfconfig.xml` file (or edit the default) for your application, you can skip this section. You can simply use the standard `icfconfig.xml` file that ships with Progress Dynamics to access the upgraded Repository.

Customizing Progress Dynamics session types

When you customize session types, you must add the appropriate managers. [Table 22](#) identifies the managers required for certain functionality.

Table 22: Managers for customized session types

Add the ...	For session types that ...
Referential Integrity Manager (RIManager) (<code>ry/app/ryrisrvrp.p</code>)	Make database connections
Customization Manager : the server side manager (<code>ry/app/rycussrvrp.p</code>); the client-side manager (<code>ry/app/rycusclntp.p</code>)	Make use of the customization facilities
	Include all development session types
	Include all server-side (AppServer and WebSpeed) session types
RepositoryDesignManager (<code>ry/app/rydessrvrp.p</code>) ¹	Design objects in the AppBuilder
RequestManager (<code>ry/app/ryreqsrvrp.p</code>)	Handle requests coming in from a Web browser
UserInterfaceManager (<code>ry/app/ryuimsrvrp.p</code>)	Handle the interactions with the user interface

1. The Repository Design Manager is only needed for development session types. It is not needed and should not be included in deployment sessions. It will only add unnecessary overhead when the application runs.

If you plan to deploy your application as a browser-based application on the Web, you must create an ICFWS session type. See [OpenEdge Development: Progress Dynamics Administration](#) and [OpenEdge Development: Progress Dynamics Web Development Guide](#) for more information.

The following example shows entries for all the new managers as they appear in the XML configuration file:

```
<manager>
  <cManagerName>RIManager</cManagerName>
  <cFileName>ry/app/ryrisrvrp.p</cFileName>
  <cHandleName>RI</cHandleName>
  <cSuperOf/>
</manager>
<manager>
  <cManagerName>CustomizationManager</cManagerName>
  <cFileName>ry/app/rycussrvrp.p</cFileName>
  <cHandleName>NON</cHandleName>
  <cSuperOf/>
</manager>
<manager>
  <cManagerName>RepositoryDesignManager</cManagerName>
  <cFileName>ry/app/rydessrvrp.p</cFileName>
  <cHandleName>NON</cHandleName>
  <cSuperOf/>
</manager>
<manager>
  <cManagerName>RequestManager</cManagerName>
  <cFileName>ry/app/ryreqsrvrp.p</cFileName>
  <cHandleName>NON</cHandleName>
  <cSuperOf/>
</manager>
<manager>
  <cManagerName>UserInterfaceManager</cManagerName>
  <cFileName>ry/app/ryuimsrvrp.p</cFileName>
  <cHandleName>NON</cHandleName>
  <cSuperOf/>
</manager>
```

Running the Entity Import tool

When the DCU runs after the initial installation, it displays an important note that you must run the Entity Import Tool with **Override all attributes from schema** selected. Your upgraded application database might not run correctly if you do not run the Entity Import Tool.

Start the Entity Import Tool from the main menu of the Administration Tool. Select **System→ Entity Import**.

In later versions of Progress Dynamics, DataFields are used extensively throughout the tools and at run time in Progress SmartDataObjects for attributes such as formats, data types, and labels. *DataFields* are a level of abstraction from the physical data storage. When you upgrade, you must run the Import Entity Tool to ensure that Progress SmartObjects are created for every entity, with DataField instances representing the fields that belong to the entity.

In addition, due to the increased and fundamental use of DataFields, it is imperative that you keep them up-to-date. Whenever you make schema changes to your application database, you must use the Entity Import tool to update your application's entities and DataFields. You must run the Entity Import tool against the central database for your organization. (If individual developers run the Entity Import on their own "satellite" databases, they might generate different Object IDs for the same DataField.)

By default, the Entity Import process does not overwrite any local changes you have made to attributes, such as labels. But if the value of your label matches the value of the old schema label, the Entity Import process updates the DataFields appropriately. The Entity import tool includes an **Override all attributes from schema** toggle box. If you select this option, the Entity Import process overwrites the local changes you made to the values of the entity attributes with the schema values.

If you do not update the DataFields, then at run time the SDOs use the old values in the DataFields, thereby regressing your schema changes at run time.

Note: The generateDataFields API in the Repository Design Manager supports the overriding of local attribute values with database metaschema values for certain attributes, such as Format, Label, and Help.

Recompiling application code

Any static application code (in product modules, for example) must be recompiled in OpenEdge. In particular, Progress Dynamics Version 1.1A code will not run in an OpenEdge environment. However, you should recompile the application code from any previous release.

Setting up for Web development

If you plan to use Progress Dynamics to create applications for the Web, you should:

- Test the Broker/Agent setup
- Test the Managers
- Test the StartUp page

For detailed information about setting up and creating Web applications with Progress Dynamics, see [OpenEdge Development: Progress Dynamics Web Development Guide](#).

Additional product installation activities

This section highlights the following additional product-related activities you might also want to perform:

- [Using an Electronic License Addendum file](#)
- [Installing additional products](#)
- [Installing additional components to previously installed products](#)
- [Viewing registry information](#)
- [Downloading executables for heterogeneous environments](#)
- [Configuring an Apache Tomcat Java Servlet Engine](#)

Using an Electronic License Addendum file

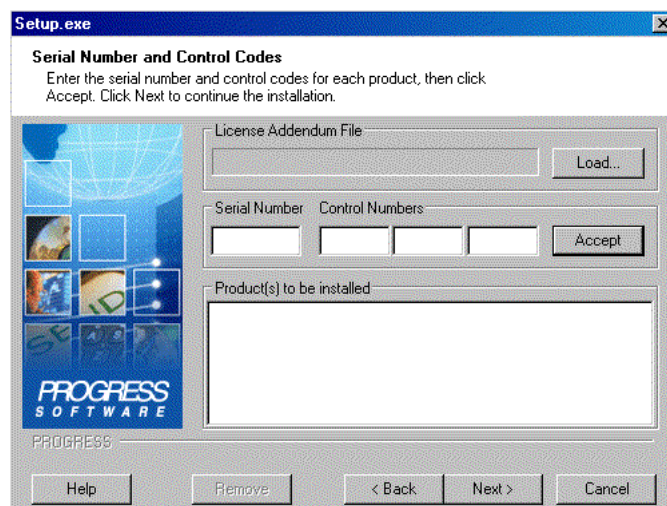
If you have obtained an Electronic License Addendum file, you can automatically enter serial numbers and product control codes. An Electronic License Addendum file contains the serial numbers and control codes for the OpenEdge license you purchased. For instructions on obtaining an Electronic License Addendum file, see the [“Obtaining an Electronic License Addendum file”](#) section on page 72.



To enter the serial number and control code for your product automatically:

1. In the **License Addendum File** field, enter the name and path of the License Addendum file.

Note: You can enter multiple License Addendum files containing additional serial numbers and control codes.



2. Click **Load.**

Once the OpenEdge install script validates the license addendum file, the Product(s) to be installed list is automatically populated.

Note: To remove products from the list, right click on the product you want to remove, and select **Delete**, or highlight the product you want to remove and click **Remove**.

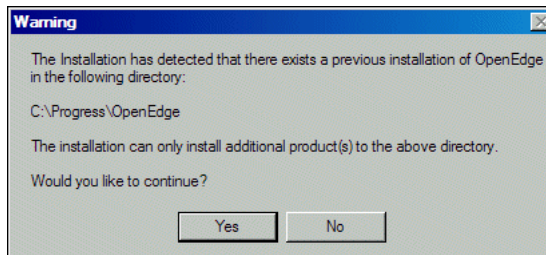
3. Once you have entered information for all the products you want to install, click **Next to continue the installation.**

Installing additional products

You can add other OpenEdge products to your current installation by following the steps outlined in the [“Installation overview”](#) section on page 98.

Note: You must shut down the AdminServer before you can successfully add additional products to a current installation.

When the installation process detects the existing version of OpenEdge, a **Warning** dialog box appears, notifying you of the existing version's location, as shown:



Note: When you add products to an existing installation, you can use the installation utility in batch mode regardless of the type of installation (complete or custom) that you are performing.

**To continue with the installation:**

1. Choose **Yes** to continue with the installation. The **Welcome** dialog box appears.
2. Choose **Next**. The **Serial Number and Control Codes** dialog box appears.
3. Enter the serial number and control numbers and choose **Accept** for each product you want to add to your current installation.

4. Choose **Next**. The **Progress License Agreement** dialog box appears.
5. Review the information and choose **Yes**. The **Choose Destination And Working Path Directories** dialog box appears. The install program deactivates (grays out) the **Browse** associated with the **Destination Directory** field and adds your OpenEdge products to directories automatically.
6. Accept the default directories and continue with the installation.

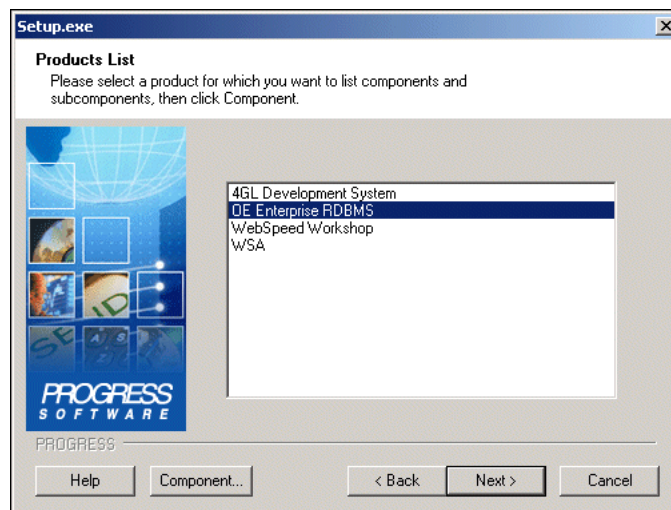
Installing additional components to previously installed products

You can add components and subcomponents to existing OpenEdge and later installations without entering any data other than the required components or subcomponents. In earlier Progress versions it is necessary to reinstall Progress and execute the “Custom Install” setup type; these steps are no longer necessary with this Add feature.



To add components or subcomponents to a previously installed product:

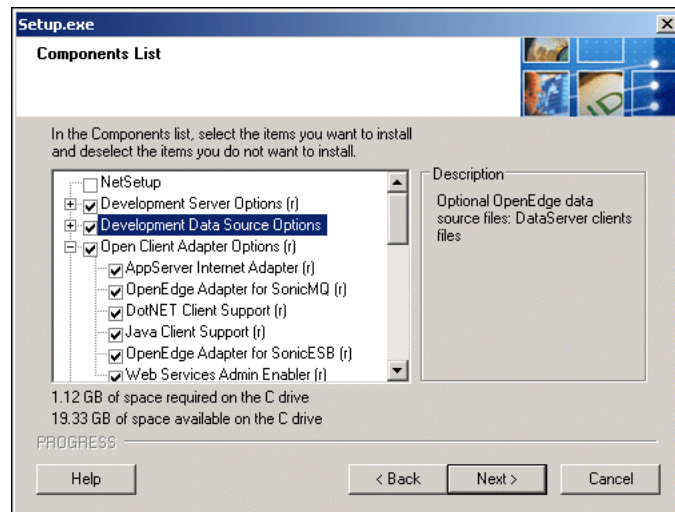
1. Choose **Start**→**All Programs**→**OpenEdge**→**Add Components**. The **Products List** dialog box appears:



All previously installed products appear on the **Products List**.

2. From the **Products List**, select the product to which you want to add components or subcomponents.

3. Choose **Component** to add components or subcomponents to the already installed product. The **Components List** dialog box appears:



Only components and subcomponents that have not been previously installed appear in the **Components List** dialog box.

4. Select the components and subcomponents you want to add.
5. Repeat [Step 2](#) through [Step 4](#), as needed.
6. Choose **Next**.
7. Choose **Finish** to update the existing install.

Note: If a system file in a newly added component or subcomponent is locked or busy during installation, a **Reboot** dialog box appears to prompt you to reboot your system.

Viewing registry information

Applications running in Windows rely on the registry for startup information, such as color, font, and key bindings.

Note: Proceed with caution when viewing registry information. Any change you make to the registry, accidentally or intentionally, could have an unexpected and potentially adverse affect on your application.

The OpenEdge installation adds the required `progress.ini` file information into the registry as entries under the following keys:

```
HKEY_CURRENT_USER\SOFTWARE\PSC\PROGRESS\11.3
HKEY_LOCAL_MACHINE\SOFTWARE\PSC\PROGRESS\11.3
```

Note: There is different `progress.ini` file information in each key.

The installation also automatically adds entries when you install an ODBC driver. For example, if you install the DataServer for ODBC, the following entries appear:

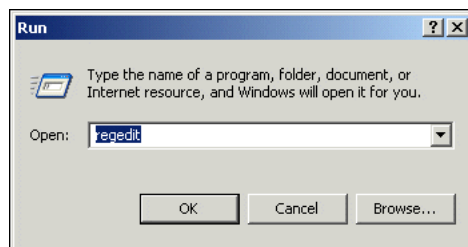
```
HKEY_CURRENT_USER\SOFTWARE\ODBC
HKEY_LOCAL_MACHINE\SOFTWARE\ODBC
HKEY_LOCAL_MACHINE\SOFTWARE\PSC\OE Personal RDBMS\11.3
HKEY_LOCAL_MACHINE\SOFTWARE\PSC\Progress ODBC\11.3
```

To add `progress.ini` file information into the registry, run the `ini2reg` utility. (The `ini2reg` updates the `HKEY_CURRENT_USERS` key.)



To review registry information:

1. On your desktop, choose **Start**→**Run**. The **Run** dialog box appears.
2. Type **regedit** (or **regedit32**) in the **Open** field:



3. Choose **OK**. A directory tree appears.
4. Double-click the `HKEY_CURRENT_USER\SOFTWARE\PSC\PROGRESS\11.3` key or the `HKEY_LOCAL_MACHINE\SOFTWARE\PSC\OpenEdge\11.3` key to view its contents.

Downloading executables for heterogeneous environments

The distributed architecture of OpenEdge allows you to optimize your hardware and network resources by installing components across networked machines, specifically when you are installing an OpenEdge Application Server component. Although some of these products' components must reside together on the same machine, you can, in some cases, distribute components to different machines, even if the machines run on different platforms. For example, you can install a WebSpeed Messenger or the NameServer in Windows and install a WebSpeed Transaction Server on UNIX.

The following list identifies executables that you can download for a platform other than Windows:

- WebSpeed Messenger
- NameServer
- Secure AppServer Internet Adapter (AIA)
- Web Services Adapter
- OpenEdge Adapter for Sonic ESB
- Promsgs (OpenEdge Messages)
- REST Adapter
- Business Process Modeler

The executables can be downloaded free of charge from Progress Software's Download Center at <http://www.progress.com/esd>.

Configuring an Apache Tomcat Java Servlet Engine

For OpenEdge development products (such as Progress Developer Studio for OpenEdge), Progress provides a script for configuring an Apache Tomcat Java Servlet Engine (JSE) and its Web server. When invoked, this script automatically configures an Apache Tomcat Web server (version 5.5 or later) and a JSE on your local machine.

Note: You do not need to perform these steps if you are using the Apache Tomcat installed with Progress Developer Studio for OpenEdge.



To configure a JSE and the Apache Tomcat Web server:

1. Download the latest Apache Tomcat Web server (version 5.5 or later) from <http://tomcat.apache.org> and extract it to your local drive.
2. Locate the batch file, `OE_TC.bat`, containing the installation script. This file is typically located in the `OpenEdge-install-dir\bin` directory.
3. Run the `OE_TC.bat` file and answer the questions prompted by the installation script.

Note: Once you install the JSE and Apache Tomcat Web server, you can test the connection. For more information see the [“Testing the configuration”](#) section on page 122.

Testing the configuration



To verify the configuration of the JSE and Apache Tomcat Web server:

1. Start the Apache Tomcat Web server. Locate the installation directory, and browse to `Tomcat-install-dir\bin`. Enter the following command:
`catalina.bat start`.

Note: Once you invoke the `catalina.bat start` command, a new command window appears, displaying information about the server startup procedure.

2. Open a browser window.
3. In the URL field of the browser, enter the default address and port number for the Web server. For example: **`http://localhost:8080`**.

Note: The default address and port number of your Web server varies.

4. Verify connectivity to the AIA servlet by entering the default address and port number for the Web server, followed by the path to the AIA servlet engine. For example: **`http://localhost:8080/aia/Aia`**.
5. Test the WSA servlet by entering the default address and port number for the Web server, followed by the path to the WSA servlet engine. For example: **`http://localhost:8080/wsa/wsa1`**.
6. Test WebSpeed connectivity by entering the default address and port number for the Web server, followed by the path to the WebSpeed administrator. For example: **`http://localhost:8080/cgi-bin/wspd.cgi.pl?WSMAdmin`**.

OpenEdge Silent installation overview

An interactive installation prompts you for input and records your values in a series of dialog boxes. The Installation Program immediately uses this data to setup your OpenEdge products.

In contrast, a Silent installation is a multi-step process:

- Data entered during the interactive installation process is recorded, typically in an `.ini` file. The OpenEdge installation automatically creates a `response.ini` file during the interactive installation process. Although you can create your own `.ini` file, the automatically-generated `response.ini` file is a reliable data input to perform a Silent installation.
- The installation data captured in an `.ini` file is read programmatically to install the products through a batch, or silent, mechanism at any time. Complete and custom installation support the Silent installation feature.

Note: If you plan to distribute a Silent installation that includes OpenEdge products that require Microsoft .NET Framework as part of the installation process, verify that the .NET Framework software is available on the system to which you are installing **before** you initiate the installation. Otherwise the Silent installation process will terminate.

The following sections describe the Silent installation steps in more detail:

- [Selecting a data input option for a Silent installation](#)
- [Understanding the response.ini file contents](#)
- [Running the Silent installation](#)
- [Checking the status of the Silent installation log file](#)

Selecting a data input option for a Silent installation

[Table 23](#) identifies and briefly describes the two types of data inputs you can use to perform a Silent installation.

Table 23: Data input options for a Silent installation

Data input options	Description
Automatically generated <code>response.ini</code> file	<p>An OpenEdge 11.3 interactive installation automatically creates a <code>response.ini</code> file that contains the installation values as you originally entered them in fields on the dialog boxes. It is stored in the <code>install</code> subdirectory in your installation directory, <code>OpenEdge-install-dir</code>. The file is immediately available for you to play back to start a Silent installation.</p> <p>See the “Understanding the response.ini file contents” section on page 124 for more information and an excerpt of the <code>response.ini</code> file.</p>
User-initiated programmatic method	<p>Provides Application Partners (APs) a streamlined approach to integrate the OpenEdge installer into an application installer. Using this method, an AP can access the automatically generated <code>response.ini</code> file to programmatically create an OpenEdge installation response file. When the AP's application is installed on a customer site, the OpenEdge installation information is read from the response file, enabling the customized install to be performed silently.</p> <p>For more information about this optional activity, see the “Creating a data input option” section on page 136.</p>

Note: You can choose to edit the response file. However, keep in mind that any modifications to the automatically- or programmatically-generated response file can be time consuming and error prone.

Understanding the response.ini file contents

The data captured in the `response.ini` file provides a detailed, reliable snapshot of the installation choices made during an interactive installation. As noted in [Table 23](#), the `response.ini` file is stored in your installation directory, `OpenEdge-install-dir`.

The `response.ini` file includes:

- A header version number and application details
- Section labels defined by brackets for easy referencing
- Each dialog box comment section identified with the label `DESCRIPTION` and the specific dialog box title
- Easy-to-read descriptions of the fields on each dialog box

- Only the values captured during the interactive install are stored in the response.ini file; there is no extraneous content
- Dialog boxes that appear in the same order as presented in the online installation
- A complete list of products installed

The initial response.ini file is created when you run the Silent installation; it is never overwritten. If you re-run the Silent installation to add products to an existing 11.3 installation, a new response.ini file is created and it is identified as response.ini.1. Any subsequent Silent installations will generate response.ini.2, response.ini.3, and so forth. These files will be saved to your installation directory.

Response.ini sample excerpt

The following example shows an excerpt from the automatically-generated response.ini file:

response.ini

(1 of 9)

```
[InstallShield Information]
Version=17.0.0.717

[Application]
Name=OpenEdge
Version=11.3
Company=Progress Software
File=Response File

;
; DESCRIPTION of Welcome Dialog
;
; Result - is used as the return code for this section. Only a value of 1 is
; acceptable.
;
[Welcome Dialog]
Result=1

;
; DESCRIPTION of Serial Number And Control Codes Dialog
;
; ProductCount - the number of products being installed.
; SerialNumber - the serial number of the product being installed.
; ControlNumber_1-0 - the first control code for the product being installed,
; where -0 indicates the first product.
; ControlNumber_2-0 - the second control code for the product being installed,
; where -0 indicates the first product.
; ControlNumber_3-0 - the third control code for the product being installed,
; where -0 indicates the first product.
; ControlNumber_1-1 - the first control code for the product being installed,
; where -1 indicates the second product.
; ControlNumber_2-1 - the second control code for the product being installed,
; where -1 indicates the second product.
; ControlNumber_3-1 - the third control code for the product being installed,
; where -1 indicates the second product.
; UseColorEditor - The only valid values are yes and no. This may be set to
; no when installing on operating system that allows bi-directional typing, such
; as Arabic or Hebrew.
; Result - is used as the return code for this section. Only a value of 1 is
; acceptable.
;
```

response.ini

(2 of 9)

```

[Serial Number And Control Codes Dialog]
ProductCount=11
SerialNumber-0=123456789
ControlNumber_1-0=Y8GR9
ControlNumber_2-0=YQHYC
ControlNumber_3-0=KMGYJ
SerialNumber-1=123456789
ControlNumber_1-1=Y9ESS
ControlNumber_2-1=3QHYM
ControlNumber_3-1=J4C?2
SerialNumber-2=123456789
ControlNumber_1-2=XZAS9
ControlNumber_2-2=3PHUC
ControlNumber_3-2=JMGY6
SerialNumber-3=123456789
ControlNumber_1-3=Y?ASS
ControlNumber_2-3=3QHYN
ControlNumber_3-3=KMC?6
SerialNumber-4=123456789
ControlNumber_1-4=X8ASS
ControlNumber_2-4=7QF3C
ControlNumber_3-4=J4CY2
SerialNumber-5=123456789
ControlNumber_1-5=XZGR9
ControlNumber_2-5=UQFYC
ControlNumber_3-5=3MGY2
SerialNumber-6=123456789
ControlNumber_1-6=Y8ES9
ControlNumber_2-6=7PH3D
ControlNumber_3-6=24G?3
SerialNumber-7=123456789
ControlNumber_1-7=Y9CR9
ControlNumber_2-7=YQH7D
ControlNumber_3-7=KMG?6
SerialNumber-8=123456789
ControlNumber_1-8=Y8GSS
ControlNumber_2-8=YQH7N
ControlNumber_3-8=2MGY7
SerialNumber-9=123456789
ControlNumber_1-9=Y8ASS
ControlNumber_2-9=UQHUC
ControlNumber_3-9=K4GY2
SerialNumber-10=123456789
ControlNumber_1-10=X8AR9
ControlNumber_2-10=UPFUM
ControlNumber_3-10=3MCY2
UseColorEditor=NULL
Result=1

;
; DESCRIPTION of TargetDirectory Dialog
;
; DestinationDirectory - identifies the directory in which you install your
OpenEdge product software.
; WorkDirectory - identifies the directory in which your applications,
databases, and log files will reside.
; Result - is used as the return code for this section. Only a value of 1 is
acceptable.
;
[TargetDirectory Dialog]
DestinationDirectory=C:\Progress\OpenEdge
WorkDirectory=C:\OpenEdge\WRK
Result=1

```

response.ini**(3 of 9)**

```

;
; DESCRIPTION of OEM TargetDirectory Dialog
;
; DestinationDirectory - identifies the directory in which you install your
OpenEdge product software.
; WorkDirectory - identifies the directory in which your applications,
databases, and log files will reside.
; Result - is used as the return code for this section. Only a value of 1 is
acceptable.
;
[OEM TargetDirectory Dialog]
DestinationDirectory=C:\Progress\oemgmt
WorkDirectory=C:\OpenEdge\wrk_oemgmt
Result=1

;
; DESCRIPTION of UserInstallationType Dialog
;
; InstallationType - identifies the type of product installation you plan to
perform. The valid values are complete and custom.
; - A Complete installation installs all mandatory, recommended, and optional
components and subcomponents of the OpenEdge products you are installing.
; - A Custom installation provides advanced users the opportunity to a
product-by-product basis.
; Result - is used as the return code for this section. Only a value of 1 is
acceptable.
;
[UserInstallationType Dialog]
InstallationType=complete
Result=1

;
; DESCRIPTION of Configuring / Installing Components Dialog
;
; ConfigureSonicESBAdapter - used to indicate whether or not you want to
manually configure the
;
; OpenEdge Adapter for Sonic ESB, or use default values.
; - a value of 0 indicates default values will be used.
; - a value of 1 indicates the SonicEsbProperties dialog will be used to
set values.
; ConfigureWebSpeedMessenger - used to indicate whether or not you want to
manually configure
;
; WebSpeed Messenger, or use default values.
; - a value of 0 indicates default values will be used.
; - a value of 1 indicates the WebServer Type dialog will be used to set
values.
; InstallingProgressDynamics - used to indicate whether or not you want to
install Progress Dynamics files.
;
; selectively install recommended and optional components and subcomponents on
- a value of 0 indicates Progress Dynamics files will NOT be installed.
; - a value of 1 indicates Progress Dynamics files WILL be installed.
; Result - is used as the return code for this section. Only a value of 1 is
acceptable.
;
[Configuring / Installing Components Dialog]
ConfigureSonicESBAdapter=1
ConfigureWebSpeedMessenger=1
InstallingProgressDynamics=0
Result=1

```

response.ini**(4 of 9)**

```

;
; DESCRIPTION of Select Program Folder Dialog
;
; ShortcutFolder - the program folder in which your OpenEdge program shortcuts
will appear.
; Result - is used as the return code for this section. Only a value of 1 is
acceptable.
;
[Select Program Folder Dialog]
ShortcutFolder=OpenEdge
Result=1
;
; DESCRIPTION of SonicEsbProperties Dialog
;
; ContainerName - identifies the Sonic ESB Container Name.
; DomainName - identifies the Sonic ESB Domain Name.
; ConnectionUrl - identifies the Connection URL to the Sonic ESB.
; UserName - identifies the User Name used to connect to the Sonic ESB.
; Password - identifies the Password used to validate the User Name.
; EsbHomeDirectory - identifies the directory where the Sonic ESB is
installed.
; Result - is used as the return code for this section. Only a value of 1 is
acceptable.
;
;
[SonicEsbProperties Dialog]
ContainerName=nbhydptripath7Container
DomainName=Domain1
ConnectionUrl=tcp://localhost:2506
UserName=Administrator
Password=Administrator
EsbHomeDirectory=unspecified
Result=0
;
; DESCRIPTION of WebServer Type Dialog
;
; ONLY one of the following Web Server types can be chosen by assigning a
value of 1 to it. Assign a value of 0 to the remaining two Web Server types.
; ISAPI - This Microsoft Web Server (IIS) supports an "in memory" messenger
(ISAPI) and the CGI messenger..
; NSAPI - This Web server supports an "in memory" messenger and the CGI
(Common Gateway Interface) messenger.
; CGIIP - This Web server supports the CGI messenger that all Web servers
support.
; ScriptDirectory - the directory in which you want the Web server scripts
stored.
; DocRootDirectory - the root directory to which WebSpeed will add files
needed for WebSpeed development.
; CopyStaticHtmlFiles - determines whether or not the install program is to
copy WebSpeed Workshop html files to the Document Root directory.
; - a value of 0 indicates NO Static Html Files are to be copied.
; - a value of 1 indicates Static Html Files are to be copied
; CreateVirtualDirectory - a value of 1 enables an alias to be created that
points to the HTML files for WebSpeed in the OpenEdge install directory and a
value of 0 disables this functionality.
; - a value of 0 indicates NO the Virtual Directory is to be created.
; - a value of 1 indicates the Virtual Directory is to be created.
; Result - is used as the return code for this section. Only a value of 1 is
acceptable.
;

```


response.ini

(5 of 9)

```

[WebServer Type Dialog]
DocRootDirectory=C:\Progress\OpenEdge\root
ISAPI=1
NSAPI=0
CGIIP=0
CopyStaticHtmlFiles=1
CreateVirtualDirectory=0
Result=1

ScriptDirectory=C:\Progress\OpenEdge\script

;
; DESCRIPTION of LanguageChoice Dialog
;
; DefaultLanguage - identifies the language in which PROMSGS appears by
default.
; DefaultCountry - identifies the country of the language in which PROMSGS
appears by default.
; -Valid values are:
;   Australia
;   Austria
;   Belgium
;   Brazil
;   Canada
;   Czech_Republic
;   Finland
;   France
;   Germany
;   Italy
;   Netherlands
;   Poland
;   Portugal
;   Spain
;   Sweden
;   Switzerland
;   United_Kingdom
;   United_States
; Languages - used to choose additional languages.
; -Valid values are:
;   Czech
;   Dutch
;   English - American
;   English - International
;   French
;   German
;   Italian
;   Polish
;   Portuguese
;   Portuguese - Brazilian
;   Spanish
;   Spanish - Latin
;   Swedish
; Result - is used as the return code for this section. Only a value of 1 is
acceptable.
;
[LanguageChoice Dialog]
DefaultLanguage=English - American
DefaultCountry=United_States
Languages=,,,,,,,,,English - American
Result=1

```

response.ini**(6 of 9)**

```

;
; DESCRIPTION of International Settings Dialog
;

; CharacterSet - identifies the -cpinternal and -cpstream values included in
the startup.pf file.
; Collation - identifies the -cpcoll value included in the startup.pf file.
; Case - identifies the -cpcase value included in the startup.pf file.
; DateFormat - identifies the -d value included in the startup.pf file.
; NumberFormat - identifies the number format value.
; NumberSeparator - identifies the -numsep value included in the startup.pf
file.
; NumberDecimalPoint - identifies the -numdec value included in the startup.pf
file.
; Result - is used as the return code for this section. Only a value of 1 is
acceptable.
;
[International Settings Dialog]
CharacterSet=ISO8859-1
Collation=Basic
Case=Basic
DateFormat=mdy
NumberFormat=1,234.56      (comma, period)
NumberSeparator=44
NumberDecimalPoint=46
Result=1

;
; DESCRIPTION of Web Services Adapter Settings Dialog
;

; WsaSampleServletUrl - this value defines the location for the sample Web
Services Adapter's Java Servlet.
; DisableWsaSecurity - indicates whether or not you want to disable the default
WSA security setting.
; - a value of 0 indicates you do NOT want to disable the default WSA security
setting.
; - a value of 1 indicates you want to disable the default WSA security
setting.
; Result - is used as the return code for this section. Only a value of 1 is
acceptable.
;
[Web Services Adapter Settings Dialog]
WsaSampleServletUrl=http://NBHYDPTRIPATH7:80/wsa/wsai
DisableWsaSecurity=1
Result=1

;
; DESCRIPTION of AdminServer Authorization Options Dialog
;

; GroupList - is a colon separated list of group names authorized to start
the Admin Server.
; RequireUsernameAndPassword - is used to require a Username and Password
before allowing the OpenEdge Admin Server to be started.
; EnableGroupChecking - is used to enable the group authorization checking.
; Result - is used as the return code for this section. Only a value of 1 is
acceptable.
;
[AdminServer Authorization Options Dialog]
GroupList=PSCAdmin
RequireUsernameAndPassword=0
EnableGroupChecking=0
Result=1

```

response.ini

(7 of 9)

```

[PscDlgGetDevEnvironments]
Sonic_Env=0
Other_Env=0
;
; DESCRIPTION of Summary Dialog
;
; Result - is used as the return code for this section. Only a value of 1 is
acceptable.
;
[Summary Dialog]
Result=1
[Installed Products]
ProductCount=11
Product 157=Progress Dev Studio OE

Product 111=OE Application Svr Basic
Product 42=OE Ultra Controls .NET
Product 108=OE DataServer MS SQL Svr

[Product 157]
__Component_NetSetup=1
__SubComponent_Client-Side Security (r)=1
__Component_Development Data Source Options=1
__SubComponent_Oracle Client=1
__SubComponent_Data Direct ODBC Driver Support=1
__Component_Other Options=1
__SubComponent_Results (Graphical)=1
__Component_Open Client Adapter Options (r)=1
__SubComponent_AppServer Internet Adapter (r)=1
__SubComponent_OpenEdge Adapter for SonicMQ (r)=1
__SubComponent_DotNET Client Support (r)=1
__SubComponent_Java Client Support (r)=1
__SubComponent_OpenEdge Adapter for SonicESB (r)=1
__SubComponent_Web Services Admin Enabler (r)=1
__SubComponent_Web Services Schema (r)=1
__SubComponent_DotNET Messages (r)=1
__Component_Studio Admin and Configuration (r)=1
__SubComponent_Toolkit (r)=1
__SubComponent_Name Server (r)=1
__SubComponent_Character Administration (r)=1
__Component_Application Server Options (r)=1
__SubComponent_WebSpeed Messenger (r)=1
__SubComponent_ADM GUI Runtime (r)=1
__SubComponent_AppServer - Dev (r)=1
__SubComponent_Transaction Server - Dev (r)=1
__SubComponent_Character Client - WebSpeed (r)=1
__SubComponent_Procedure Editor - CHAR Recommen (r)=1
__Component_Client-Side Web Services (r)=1
__SubComponent_Web Services Basic (r)=1
__SubComponent_WSDL Analyzer (r)=1
__Component_OEBuild Utility (r)=1
__Component_Progress Developer Studio AppBuilder (r)=1
__SubComponent_Compile Tool - CHAR=1
__SubComponent_Compile Tool GUI (r)=1
__SubComponent_WebSpeed Workshop Dev (r)=1
__Component_Progress Developer Studio Development (r)=1
__SubComponent_Character Client - RT=1
__SubComponent_Character Image=1
__SubComponent_Application Debugger (r)=1
__SubComponent_WebClient Assembler Utilities (r)=1
__SubComponent_Progress Developer Studio (r)=1
__SubComponent_Progress Dynamics - Progress Developer Studio for OE=1
__SubComponent_Progress Dynamics RT - Progress Developer Studio for OE=1
__SubComponent_XSD-4GL (r)=1

```

response.ini

(8 of 9)

```

[Product 111]
__SubComponent_Client-Side Security (r)=1
__Component_Open Client Adapter Options Basic (r)=1
__SubComponent_AppServer Internet Adapter (r)=1
__SubComponent_OpenEdge Adapter for SonicMQ (r)=1
__SubComponent_Java Client Support (r)=1
__SubComponent_DotNET Messages (r)=1
__Component_Server Data Source Options=1
__SubComponent_Oracle Client=1
__SubComponent_Data Direct ODBC Driver Support=1
__Component_Basic Server Options (r)=1
__SubComponent_WebSpeed Messenger (r)=1
__SubComponent_AppServer - Basic (r)=1
__SubComponent_Transaction Server - Basic (r)=1
__SubComponent_Name Server (r)=1
__SubComponent_ADM GUI Runtime (r)=1
__SubComponent_Character Client - WebSpeed (r)=1
__Component_OEBuild Utility (r)=1
__Component_Client-Side Web Services Deploy (r)=1
__SubComponent_Web Services Basic (r)=1
__SubComponent_Web Services Schema (r)=1

[Product 108]
__Component_NetSetup=1
__Component_Character Database Admin Tools -=1
__Component_Character Base Tools - Optional=1
__SubComponent_ADM Runtime - CHAR=1
__Component_Character Image - Dev - Optional=1
__Component_Character Runtime Client-Dev-Opt=1
__SubComponent_Client-Side Security (r)=1
__Component_Open Client Adapter Options Basic (r)=1
__SubComponent_AppServer Internet Adapter (r)=1
__SubComponent_OpenEdge Adapter for SonicMQ (r)=1
__SubComponent_Java Client Support (r)=1
__SubComponent_DotNET Messages (r)=1
__Component_OEBuild Utility (r)=1

[Product 8]
__SubComponent_Client-Side Security (r)=1
__SubComponent_German=1
__SubComponent_Spanish=1
__SubComponent_French=1
__SubComponent_Dutch=1
__SubComponent_Portuguese=1
__SubComponent_Swedish=1
__SubComponent_Czech=1
__SubComponent_Polish=1
__SubComponent_Italian=1
__SubComponent_Portuguese - Brazilian=1
__SubComponent_Spanish - Latin=1
__SubComponent_English - International=1
__SubComponent_English - American=1

[Product 311]
__SubComponent_Client-Side Security (r)=1

[Product 21]
__SubComponent_Client-Side Security (r)=1

[Product 343]
__SubComponent_Web Services Admin Enabler (r)=1
__SubComponent_Client-Side Security (r)=1

```

response.ini

(9 of 9)

```
[Product 105]
__Component_NetSetup=1
__SubComponent_Character Administration Tools=1
__SubComponent_Compile Tool - CHAR=1
__SubComponent_Oracle Client=1
__SubComponent_Character Client - 4GL=1
__SubComponent_Compile Tool - GUI=1
__SubComponent_Web Services Schema (r)=1
__SubComponent_ADM Runtime - CHAR=1
__SubComponent_ADM Runtime - GUI=1
__SubComponent_Character Image=1
__SubComponent_Procedure Editor - CHAR Optional=1
__SubComponent_Client-Side Security (r)=1
__Component_OpenEdge SQL ODBC Clients=1
__Component_OpenEdge SQL JDBC Clients=1
__Component_OpenEdge ESQ/C Clients=1
__Component_Open Client Adapter Options Basic (r)=1
__SubComponent_AppServer Internet Adapter (r)=1
__SubComponent_OpenEdge Adapter for SonicMQ (r)=1
__SubComponent_Java Client Support (r)=1
__SubComponent_DotNET Messages (r)=1
__Component_OEBuild Utility (r)=1
__Component_Failover Clusters (r)=1
__Component_Client-Side Web Services Deploy (r)=1
__SubComponent_Web Services Basic (r)=1
```

Running the Silent installation

The command you use to initiate, or play back, the response file is the same regardless of which data input option you choose. The OpenEdge Silent utility runs without intervention after you enter the command to start the process.

The syntax for running the OpenEdge Installation utility in silent mode is:

Syntax

```
<path-to-install-media>\setup.exe -psc_s [-notify]
-psc_f1=<path>\<response-file-name> [-psc_f2=<path>\<logfile-name>]
```

Note: Do not leave a space between command line entries and options. Also, neither command line entries nor options are case sensitive.

<path-to-install-media>\setup.exe

Runs an OpenEdge installation. The <path-to-install-media> indicates that you can run the installation from the installation executable downloaded with your product from the Progress Download Center, <http://www.progress.com/esd>. The setup.exe identifies the specific OpenEdge installation executable.

-psc_s

Indicates a Silent installation.

`-notify`

Indicates that the installation dialog boxes that display will contain details about the current installation phase and percent complete. This element is supported for backward compatibility only.

The preferred method is to set up your application installation program to poll the log file for status of the installation process. You can programmatically set up a query, checking the Runtime Status and Result Code values in the log file. See the [“Checking the status of the Silent installation log file”](#) section on page 134 to review the contents of a sample `oesetup.log` file that contains Runtime Status and Result Code details.

`-psc_f1=<path><response-file-name>`

Specifies the pathname and filename of the file. By default, the install will look for the response file `oesetup.ini` in the same directory as `setup.exe` is located.

`-psc_f2=<path><logfile-name>`

Indicates that an installation log file will be created, and specifies the pathname and filename of the installation log file. If no filename is specified, the OpenEdge Installation Utility provides the default log filename `oesetup.log`.

If you do not specify a value for `<path>`, the Installation Utility writes this file to the Windows directory.

Example

The following example shows a typical Silent installation command:

```
\\cd-server\OpenEdge\setup.exe -psc_s  
-psc_f1=C:\SilentInstalls\oesetup.ini -psc_f2C:\SilentInstalls\  
oesetup.log
```

Checking the status of the Silent installation log file

The Silent installation process automatically generates a log file, in which all messages—error and successful installation—are reported.

A log file, `oesetup.log`, is automatically generated for the OpenEdge install. The data captured in this log is useful when performing a Silent installation. It contains status information that you can query at run time. The `oesetup.log` file can also be used to debug a Silent installation.

An initial `oesetup.log` file is created when you install. Any time you re-run the Silent installation, subsequent, `oesetup.log` files are automatically created and saved as `oesetup.log.1`, `oesetup.log.2`, `oesetup.log.3`, and so forth. In all cases, by default the file is located in `C:\Windows`.

An excerpt from an `oesetup.log` file follows:

oesetup.log

```
OPENEDGE INSTALL UTILITY LOG <VERSION 11.3ALPHA> (Fri Sep 23 14:36:45 2011)
[Application]
Name=OpenEdge
Version=11.3
Company=PSC

[ResponseResult]
ResultCode=0
ResultDescription=The install completed successfully.

[DetectPreviousInstallSettings]
Information=[09-23-2011 14:36:45] Unable to locate file /etc/progress.

[CompletedEvents]
Event1=[09-23-2011 14:37:53] The Setup Utility is extracting archives

Event2=[09-23-2011 14:39:53] The Setup Utility has extracted archives
Event3=[09-23-2011 14:40:06] The Setup Utility has tailored files.
Event4=[09-23-2011 14:40:09] The Setup Utility has merged delta files.
[RuntimeStatus]
Progress=99

[UpdateUnixRegistry]
File=[09-23-2011 14:39:55] /etc/progress has been created successfully.

[FilesTailored]
File1=[09-23-2011 14:39:55] /usr/dlc/bin/proaiw has been tailored
successfully.
File2=[09-23-2011 14:39:55] /usr/dlc/bin/proapw has been tailored
successfully.
[DeltaFilesMerged]
File1=[09-23-2011 14:40:08] /usr/dlc/properties/delta/fathom.delta has been
merged successfully.
File2=[09-23-2011 14:40:09] /usr/dlc/properties/delta/replication.delta has
been merged successfully.

[TailoringExtensions]
Extension1=[09-23-2011 14:40:13] /usr/dlc/bin/rptlr.dll has been executed
successfully.
Extension2=[09-23-2011 14:40:13] /usr/dlc/bin/prodbgtlr.dll has been executed
successfully.
Extension3=[09-23-2011 14:40:13] /usr/dlc/bin/Fathomtlr.dll has been executed
successfully.
[TailoringClasses]
Start=[09-23-2011 14:40:13]
Finish=[09-23-2011 14:40:44]
INFORMATION=[09-23-2011 14:42:32] The Sonic Domain Manager is not running and
the OpenEdge SonicESB Adapter automatic configuration will not occur.

When the Sonic Domain Manager is available, the OpenEdge SonicESB Adapter
should be configured by running: /usr/dlc/sonic/ESB8.0/config/offline.sh.
```

Optional data input activities

The following optional activities are also supported when you are performing a Silent installation. Create the response file using the automatic method described in the [“Understanding the response.ini file contents”](#) section on page 124. Keep in mind that creating the response file manually or editing the response file is the more time-consuming and potentially error-prone approach.

Creating a data input option

You can choose to record a separate response file any time you perform an interactive installation. If you do not specify a filename for the response file that you create, the install provides the filename `oe-response.ini` and stores it in

`C:\Windows\oe-response.ini`. The format and structure of any data input option is identical to that which is presented in the automatically-generated `response.ini` file. See the [“Response.ini sample excerpt”](#) section on page 125 to review an excerpt of the file’s content.

The syntax for initiating the user-defined response file is:

Syntax

```
<path-to-install-media>\setup.exe  
-psc_r [-psc_fl=<path>\response-file-name]
```

Note: Do not leave a space between command line entries and options. Command line entries nor options are case sensitive.

`<path-to-install-media>\setup.exe`

Runs an OpenEdge installation. The `<path-to-install-media>` indicates that you can run the installation from the installation executable downloaded with your product from the Progress Download Center, <http://www.progress.com/esd>. The `setup.exe` identifies the specific OpenEdge installation executable.

`-psc_r`

Indicates that the install is in record mode.

`-psc_fl=<path><response-file-name>`

Indicates that the response file will be created, and specifies the pathname and the filename of the file. By default, the install will look for the response file `oesetup.ini` in the same directory the `setup.exe` is located.

Manually modifying data input option

You can edit any response file, whether it is automatically generated or one you create. Although all sections of the response file are required, you do not need to add each of these required sections in the order presented. The installer only retrieves the specific data it needs regardless of where the information is located in the response file.

Performing postinstallation tasks

Before you run OpenEdge, there are some postinstallation tasks you might need to complete, depending on your application needs and goals:

- **Completing the Progress Dynamics Configuration Utility (DCU)** — The DCU wizard guides you through the setup steps to install Progress Dynamics. For the procedures to complete the DCU, see the [“Completing the DCU wizard”](#) section on page 102.
- **Completing third-party software installations** — If you installed products that require the Microsoft .NET Framework, and you agreed to OpenEdge installing the framework, the installation of the Microsoft .NET Framework automatically launches. See the [“Required third-party applications”](#) section on page 46.
- **Set environment variables** — For more information on setting environment variables (including SQL), see [Chapter 7, “Working in the OpenEdge Environment in Windows.”](#)
- **Create customized executables** — To create customized product executables, see the information on building ABL executables in [OpenEdge Deployment: Managing ABL Applications](#). Creating executables might be required for certain product configurations.
- **Re-apply properties file details (if needed)** — See the [“OpenEdge automatic save of properties files”](#) section on page 78 for details.
- **Validate and populate group names created for the AdminServer security option** — The tasks used to verify that your groups were created, and to identify the specific members of these groups, are completed outside of the OpenEdge installation. The installation process only allows you to create the groups that you specify.

Note: For detailed information about how to verify that groups have been created, and how to access and set up group members for each group in Windows, refer to your operating system-specific documentation. The criteria you use to set up users within each group is determined by your company.

- **Edit files to point to a previously installed JDK** — If the required version of the Java Soft (InstallShield) JDK was installed on your system prior to the OpenEdge Version 11.3 installation and you choose to use this pre-existing JDK utility, as a postinstallation task, you must edit files tailored by the install to ensure that they point to this pre-existing JDK. Contact Progress Technical Support for assistance to perform this task.

Uninstalling OpenEdge in Windows

When you delete files from the OpenEdge directory tree, you only partially remove an OpenEdge installation. By contrast, the Uninstall utility or the Remove Program utility removes all OpenEdge files as well as the configuration information from the registry. This prevents conflict with subsequent OpenEdge installations. If for some reason the Uninstall utility cannot completely uninstall OpenEdge, you must manually remove the installation.

If you want to upgrade or remove an installation, choose one of the following:

- Run the Uninstall utility from the OpenEdge program group, as described in the “[Using the Uninstall or Add/Remove Programs utility](#)” section on page 138.
- Run the Add/Remove Programs utility from the Microsoft Windows Control Panel, as described in the “[Using the Uninstall or Add/Remove Programs utility](#)” section on page 138.
- Manually remove the installation, as described in the “[Manually removing OpenEdge](#)” section on page 140. For additional details, see the Progress Software Company Knowledge Center at <http://progress.atgnow.com/esprogress>.

Notes: The OpenEdge uninstall does not remove the Microsoft .NET framework.

If OpenEdge installed the Infragistics NetAdvantage product, OpenEdge uninstalls it during the execution of the OpenEdge uninstall.

If the Infragistics product was installed independent of OpenEdge, OpenEdge does not uninstall it.

Using the Uninstall or Add/Remove Programs utility

You can run the Uninstall utility (or use the Add/Remove Programs utility located in the Windows Control Panel) to automatically remove OpenEdge from your system. Running the Uninstall or the Remove Program utility removes configuration information from the registry and prevents conflict with subsequent OpenEdge installations.

Caution: When uninstalling, do not delete any of the following Microsoft system files: asycfilt.dll, comctl32.ocx, ctl3d32.dll, mfc70.dll, mfc71.dll, mfcans32.dll, mscomctl.ocx, msvci70.dll, msvc71.dll, msvcr.70.dll, msvcr71.dll, oc30.dll, oleaut32.dll, olepro32.dll, pdh.dll, psapi.dll, stdole2.tlb. These system files are common to other applications, and deleting them might adversely affect the operation of the other applications that use them. To avoid deleting these system files while running the Uninstall utility, answer **No** to the prompts at the end of the uninstall process.

If you have installed the OpenEdge Ultra Controls for .NET, do not uninstall the Infragistics NetAdvantage with Add/Remove Programs.

**To run the Uninstall utility:**

1. Log in under the same domain and user name you used when you installed OpenEdge.
2. Make sure that OpenEdge is not running in an open DOS window (or that the current directory is not any OpenEdge-related directory).
3. Stop all OpenEdge processes, including your Web server (for example, you might be using a Microsoft Web server or ISAPI-compatible, or Sun Web server or NSAPI-compatible Web server), and close all OpenEdge help files. Use the Task Manager to ensure that you stop all processes and close all help files.
4. Using the OpenEdge Management or OpenEdge Explorer, shut down all OpenEdge and WebSpeed services (brokers, NameServers, and database servers).
5. If you have installed OpenEdge Management, stop the OpenEdge Management Trend Database.

You can use OpenEdge Management or OpenEdge Explorer or the following command:

```
dbman -stop FathomTrendDatabase
```

The AdminServer must be running in order to stop the OpenEdge Management Trend Database.

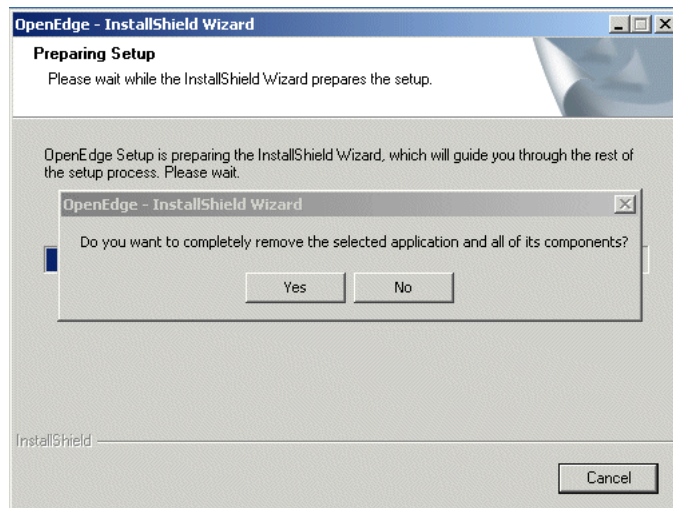
If you receive a warning during the uninstall that the `fathom.db` is in use, the OpenEdge Management Trend Database has not been stopped.

6. From the Windows desktop, select **Start**→**Settings**→**Control Panel**→**Administrative Tools**→**Services**.

Highlight the **AdminService for OpenEdge 11.3**, and choose **Stop**. Change the startup from **Automatic** to **Manual**, and choose **OK**.

Note: If these same services will be required for a new installation, be sure to note any configuration settings, agent parameters, etc.

7. Do one of the following tasks to uninstall:
 - a. Choose the **Uninstall** icon in the OpenEdge Program Group.
 - b. From the desktop, select **Start**→ **Control Panel**→ **Add or Remove Programs**. Select **OpenEdge** and choose **Change/Remove**. The **InstallShield Wizard** appears:



8. Choose **Yes**.
9. Choose **OK** when the deletion process is complete. If the uninstall was successful, you have finished. However, if the uninstall failed or terminated abnormally during the process, you must manually remove the OpenEdge `Uninstall` folder. Refer to the [“Manually removing OpenEdge”](#) section on page 140 for the procedure to complete.

Manually removing OpenEdge

If you attempted to perform the uninstall procedure outlined in the [“Using the Uninstall or Add/Remove Programs utility”](#) section on page 138 and it failed, you must manually remove the OpenEdge `Uninstall` folder before reinstalling. The uninstall file records the initial installation and appends additional installations to the file.

This section provides guidelines to manually uninstalling the OpenEdge folder located at `C:\Program Files\InstallShield Installation Information\{CFD926DB-10C8-4CB6-A6B3-49FD8F98262F}` and performs other steps related to this task.



To manually uninstall OpenEdge:

1. Log in under the same domain and user name you used when you installed OpenEdge.
2. Make sure that OpenEdge is not running in an open DOS window (or that the current directory is not any OpenEdge-related directory).
3. Stop all OpenEdge processes and close all OpenEdge help files. You can use the Task Manager to ensure that you stop all processes and close all help files.

4. Using the OpenEdge Management or OpenEdge Explorer, shut down all OpenEdge services (brokers, NameServers, and databases).
5. Shut down the AdminServer by following these steps:
 - a. From the desktop, select **Start**→ **Control Panel**→ **Administrative Tools**→ **Services**.
 - b. Highlight the **AdminService for OpenEdge 11.3**, and select **Stop**.
 - c. When the service stops, choose **Action**→ **Properties**. The **AdminService** dialog box appears.
 - d. Change the **Startup type** from **Automatic** to **Manual**, and choose **OK**. (This step is necessary if you reboot your machine before completing the uninstall so that the AdminServer does not start up automatically.)

Note: If these same services will be required for a new installation, be sure to note any configuration settings, agent parameters, and so forth.

6. Shut down some services, as needed. Consider the following situations:
 - If you are using a Sun Web server (or NSAPI-compatible Web server) that uses the `wsnsa.dll`, you are not required to shut down a Windows service. You only have to shut down the Web server and the WebSpeed Transaction Server.
 - If you are using the Microsoft IIS Web server to use the WebSpeed Messenger that uses the `wsisa.dll`, you must shut down the IIS Admin Service.
7. Remove the `C:\Program Files\InstallShield Installation Information\{CFD926DB-10C8-4CB6-A6B3-49FD8F98262F}` directory.
8. Run `regedit.exe` (or `regedt32.exe`) to edit the Windows registry as follows:
 - a. Remove the 11.3 keys that appear under the `HKEY_CURRENT_USER` location. If there is only one release number identified under `PSC`, delete the `PSC` key, as shown:

`HKEY_CURRENT_USER\SOFTWARE\PSC\Progress\release number`

- b. Remove the 11.3 keys that appear under the `HKEY_LOCAL_MACHINE` location. Check the release number identified under each product subfolder. If only one release number is identified as installed for all products, delete the `PSC` key, as shown:

`HKEY_LOCAL_MACHINE\SOFTWARE\PSC\product name(s)\release number(s)`

Note: If you have installed 32-bit version of OpenEdge on a 64-bit version of Windows, the keys are located under

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\PSC\product  
name(s)\release number(s).
```

- c. Remove the 11.3 key that appears under the following HKEY_LOCAL_MACHINE location:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\
Uninstall
```

9. If you have installed an OpenEdge product for which drivers have also been installed, run `regedit.exe` (or `regedt32.exe`) to edit the Windows registry as follows:

- a. Go to HKEY_LOCAL_MACHINE\SOFTWARE\ODBC\ODBCINST.INI.
- b. Remove the following key(s) (and the values it contains), as needed:

```
DataDirect 4.20 32-BIT OpenEdge SQL v11.3
OpenEdge 11.3 Informix Driver
OpenEdge 11.3 Informix Wire Protocol Driver
OpenEdge 11.3 SQL Server Wire Protocol Driver
OpenEdge 11.3 Sybase Wire Protocol Driver
```

10. Depending on the products you have installed, the following files might have been registered during the install and should be unregistered:

dzocx32.ocx	pstimer.ocx
dzstat32.ocx	prox.dll
duzocx32.ocx	sstree.ocx
cscomb32.ocx	cihttp.ocx
csspin32.ocx	cslist32.ocx

Note: The registered version of some of these files might not be under the OpenEdge installation directory. Check the Windows `system`, `system32`, and `SysWOW64` directories for these files.

The following example shows how you can unregister one of these OCX files. In the example it is located in the Windows `system32` directory:

```
OpenEdge-install-dir\bin\regsvr32.exe/u c:\windows\system32\cscomb32.ocx
```

11. Delete the OpenEdge program directory, including all of its subdirectories. The default OpenEdge directory is `C:\Progress\OpenEdge`.
12. Delete the OpenEdge folder/group from the Windows **Start** menu.

13. Shut down your Web server and delete the `cgiiip.exe` and `wsisa.dll` files from the Web server `cgi-bin/scripts` directory.

Note: If you are uninstalling WebSpeed and using the Sun Web server that uses the `wsnsa.dll`, you must return the `obj.conf` file to its pre-WebSpeed state. If you are upgrading WebSpeed to the same directory, you need not modify the `obj.conf` file. However, if you intend to change the directory location, then you must modify the `obj.conf` file to reflect the correct location.

14. Depending on the installation options you chose (that is, Web server type, WebSpeed virtual directory, or having static HTML files copied to your Web server document root directory), you might need to perform either one, or both of the following steps:
 - a. Delete the WebSpeed directory from your Web server document root directory. For example, on MSIIIS the default is:
`\InetPub\wwwroot\webspeed.`
 - b. Delete any virtual directories defined for WebSpeed in your Web server.
 - c. Reboot your machine and install OpenEdge.

Sharing an OpenEdge installation on a network overview

You can use OpenEdge networking functionality and the Shared Network Installation Utility (NetSetup) to install a single copy of the OpenEdge Installation Program on a network-accessible drive (server) and enable multiple clients to access it.

Note: Citrix MetaFrame does not support a shared network installation.

Primary tasks

The primary tasks to share an OpenEdge installation on a network are:

- Perform an OpenEdge installation on a network server machine, using the OpenEdge Installation Program.

During the installation process, the Shared Network Installation (NetSetup) Utility—the component that allows each client machine to install the required software to access the network server machine—is installed on the server. In a Complete installation, the NetSetup component is automatically installed with all OpenEdge products. In a Custom installation, you must select the NetSetup component as an optional component. The NetSetup Utility also supports a Silent installation option.

- Use the NetSetup Utility to update each client machine, enabling it to access the network server's installation copy.

The NetSetup Utility ensures that all the system files, icons, and registry entries needed to launch the OpenEdge products locally are set up on each client machine. The NetSetup Utility is comprised of one dialog box, the **Destination and Work Paths** dialog, that you run on the client.

The details to address the tasks previously listed and other related activities are described in the following sections:

- [Networking overview](#)
- [Determining a shared network to clients connection](#)
- [Setting up the shared network](#)
- [Running the Shared Network Installation Utility to set up a client connection](#)
- [Reviewing local intranet security settings](#)
- [Uninstalling the Shared Network Installation Utility](#)
- [Running the Silent installation option for the Shared Network Installation Utility](#)

Networking overview

This section provides some background information about the basic networking hardware needed to run OpenEdge in a network-to-client configuration.

A network typically consists of the following hardware elements:

- **Application workstation** — A computer on your network that executes the OpenEdge Client or single-user software. This allows one or more users to access the database server machine.
- **Database server machine** — The OpenEdge database server machine is a computer on your network that executes the OpenEdge Server software. This software allows the database server machine to manage one or more OpenEdge databases.
- **Network file server machine** — The network file server is a computer that manages file sharing and system security, coordinates station-to-station communications, and controls any attached peripherals, such as printers, disk drives, and modems.

You can install an OpenEdge client on a single node, or you can install it on a network file server. For more information about networking with OpenEdge, see the [“Working with Unified Brokers”](#) section on page 276.

Note: Progress Software Corporation (PSC) does not support installing one copy of the OpenEdge Application server products for multiple machines because there is only one set of configuration files; conflicts will occur.

Determining a shared network to clients connection

You can use the following connection types to share OpenEdge installed on a network server with multiple client machines:

- Mapped drive
- Uniform Naming conventions (UNC) pathnames

Setting up the shared network

This section describes setting up the OpenEdge products on your shared network server.

Prerequisites

Before you set up a shared network installation on your network server, perform the following tasks:

- Uninstall any existing OpenEdge or Progress product that is installed on client machines to which you will be installing. See the [“Uninstalling OpenEdge in Windows”](#) section on page 138.
- Review the OpenEdge installation tasks. See the [“OpenEdge Installation Prerequisites”](#) section on page 67 and the [“Installation overview”](#) section on page 98.
- Determine the destination location of your OpenEdge installation on the network. You will be prompted to enter this information during the network installation, and when you use the NetSetup utility to install the connecting clients.



To perform a shared network installation on your network server:

1. Run the OpenEdge installation.
2. When the **Choose Destination And Working Path Directories** dialog box appears, make a note of the location you type in the **Destination Directory** field. You will need this directory path information when you install on each client machine.
3. Complete the OpenEdge installation on the server.

Running the Shared Network Installation Utility to set up a client connection

The Shared Network Installation Utility (NetSetup) updates each client machine with all system files, icons, and registry entries needed to launch OpenEdge locally. Each client can then share the networked copy of OpenEdge.

The client machine in a NetSetup installation uses the OpenEdge installer located on the network server. The installer software enables you to locally launch NetSetup.

Note: Uninstall any existing 11.3 OpenEdge product that is currently installed on a client machine to which you are installing. For information on uninstalling, see the [“Uninstalling OpenEdge in Windows”](#) section on page 138.



To run NetSetup on your client machine:

1. Choose **Start**→**Run**. The **Run** dialog box appears.
2. In the **Open** field, type one of the following supported connection options to connect the client machine to the shared network server:

- a. To identify a mapped drive connection, type:

```
drive:\destination_path\netsetup\setup.exe
```

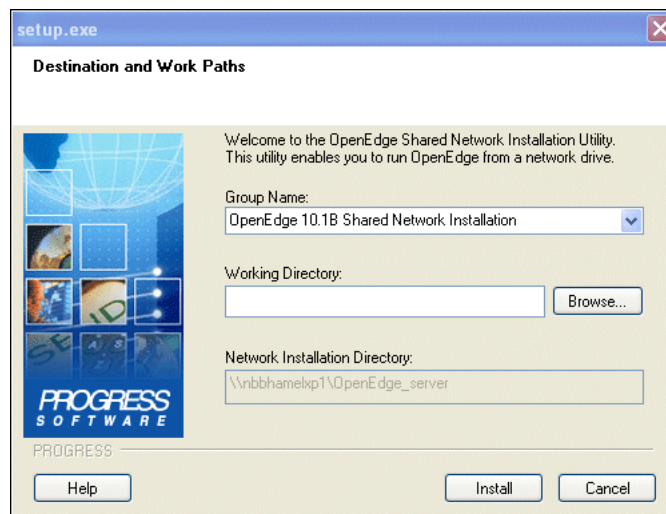
The *destination path* is the same path where OpenEdge is installed on the server machine.

- b. To identify the UNC pathname connection, type:

```
\\servername\sharename\destination_path\netsetup\setup.exe
```

The *destination path* is the same path where OpenEdge is installed on the server machine.

3. Choose **OK**. The **Destination and Work Paths** dialog box appears:



4. Accept or change the **program group** name that appears in the **Group Name** field. The **Group Name** value identifies the menu option label that appears on the client machine. When you select this name from **All Programs**, you can access the OpenEdge installation that resides on the network server.

Note: If the group name does not already exist, the NetSetup utility adds the group name to **All Programs**.

5. Type the absolute path or browse to find the file to identify as the client-based working directory in the **Working Directory** field. The **Working Directory** is a local folder in which OpenEdge places the files you create on the client.

6. Review the pathname information that appears in the **Network Installation Directory** field. This pathname identifies where OpenEdge is installed on the network server.

Note: The Network Installation Directory field always appears grayed out, confirming that the information that appears in this field cannot be changed. The pathname that appears in this field identifies two pieces of information: where OpenEdge is installed on the network server and the type of connection that you are using to share the network installation (that is, mapped drive or UNC pathname).

7. Choose **Install**. Select the Group Name you defined from **All Programs** to access the OpenEdge installation from the network serve.

Note: If you change the original installation on the network, and the installation includes additional shortcuts supported by the NetSetup Utility, you must uninstall and reinstall the NetSetup Utility on the client to ensure that the shortcuts are available on the client machine.

Shortcuts

Table 24 shows all of the OpenEdge product-specific shortcuts that can be potentially available on a shared network server and clients.

Table 24: Available Network Server and Client Shortcuts

AppBuilder	Desktop
Application Compiler	Help
Audit Policy Maintenance	Proenv
Character Client	OpenEdge Management or OpenEdge Explorer Tool
Client	Proxy Generator
Config	Release Notes
Data Administration	RESULTS
Data Dictionary	Translation Manager
Debugger	Visual Translator

Keep in mind that the specific shortcuts available to a shared network server and its client machines will vary, depending on the actual OpenEdge products installed on the network.

Reviewing local intranet security settings

The .NET Framework includes a system of Code Access Security (CAS) that tries to prevent untrusted code from performing privileged operations. You might need to alter the CAS settings on your local machine to use certain OpenEdge features. In particular, the ProxyGen tool and any OpenEdge application using the GUI for .NET features, including the GUI client, Batch clients, Application Servers, etc., need higher permissions than the default settings for network shares.

The NetSetup Utility includes the option to automatically make the necessary changes. You should consult your IT administrator on the security implications before choosing this option. For more information, see PSDN for a white paper on deploying OpenEdge GUI for .NET applications.

Uninstalling the Shared Network Installation Utility

You can use an uninstall utility to uninstall the NetSetup Utility from a client machine that is currently connected to an OpenEdge shared network installation. All the products that you previously installed for this OpenEdge release are removed. This procedure must be done for each client machine you intend to uninstall.



To uninstall the client machine from a Shared Network installation:

1. From the desktop, choose **Start**→ **Control Panel**→ **Add Or Remove Programs**.
2. From the list of installed programs, select the **OpenEdge 11.3 Shared Network Installation**. Choose **Change/Remove**. A confirmation dialog box appears.

Note: Remove client files first, then uninstall the server to ensure that the shared network installation is properly uninstalled.

3. Choose **Yes** to confirm that you want to delete the OpenEdge 11.3 Shared Network Installation from your client machine. The **Remove Programs From Your Computer** dialog box appears.
4. Choose **OK** to exit the Uninstall utility from the client machine.

Note: When the usage count on a shared system file reaches 0, a **Shared File warning** dialog box appears; follow the instructions in the dialog box.

Running the Silent installation option for the Shared Network Installation Utility

NetSetup supports a Silent installation process. The process is comparable to performing a Silent installation of the OpenEdge installation. Data entered during a NetSetup installation is recorded and played back at a later date to initiate the installation silently.

To perform a NetSetup Silent installation, however, you must create your own response file. Unlike the OpenEdge installation, `response.ini` is created by default during a NetSetup installation.

The information in this section describes:

- [Creating a user-defined response file](#)
- [Executing NetSetup with the Silent installation option](#)

Creating a user-defined response file

Before you can run the NetSetup utility in Silent installation mode, you must create the user-response file. This file records the values that the NetSetup utility needs to successfully complete the Silent installation process. This section describes how to create a response file using the interactive method.

To create this file, you must perform an initial interactive installation, providing the required values.



To create the user-defined response file using the interactive installation mode:

1. Enter the following command on the command line:

```
drive:\destination_path\netsetup\setup.exe -r
        [-flC:\<path-to-file>\response-file]
```

`setup.exe`

The command to run the NetSetup program interactively.

`-r`

Directs the install to create the response file using the interactive method. The response file is an editable text file. If you do not specify the response filename with the `-fl` parameter, the file is named `setup.ini`.

Note: The `-r` parameter is the recommended method to ensure that you create a complete response file.

`-fl<path>\<response-file-name>`

Specifies the name of the response file. By default, the install will look for the file `setup.ini` in the same directory as `setup.exe` is located.

2. Press ENTER. NetSetup runs interactively.

When you type values through the keyboard, NetSetup simultaneously creates the response file. Values specific to your installation are read and stored in the response file.

The following example shows the typical contents of a sample response file, `setup.iss`:

setup.iss

```
[Silent]
Version=v7.00
File=Response File
[File Transfer]
OverwrittenReadOnly=NoToAll
[{874D5CE4-F913-4D5B-A6D4-CC129785B5C8}-DlgOrder]
Dlg0={874D5CE4-F913-4D5B-A6D4-CC129785B5C8}-DLG_SHARED_INSTALL-0
Count=2
Dlg1={874D5CE4-F913-4D5B-A6D4-CC129785B5C8}-MessageBox-0
[{874D5CE4-F913-4D5B-A6D4-CC129785B5C8}-DLG_SHARED_INSTALL-0]
ProgramFolder=OpenEdge 11.3 Shared Network Installation
WorkingDir=C:\OpenEdge\NetinstWrk
[Application]
Name=OpenEdge Shared Network Install Utility
Version=11.3
Company=PSC
Lang=0009
[{874D5CE4-F913-4D5B-A6D4-CC129785B5C8}-MessageBox-0]
Result=1
```

The values entered for the **Program Folder** and **WorkingDir** during the interactive installation are recorded in the response file. The shortcut information identified in the **Program Folder** and the user's work files identified in the **WorkingDir** are read during a Silent installation.

Note: The **OverwrittenReadOnly** option ensures that the **Read Only File** dialog box is suppressed during a Silent installation.

Executing NetSetup with the Silent installation option

Once the response file exists, the installation process using the silent mode can be initiated.

To initiate NetSetup with the Silent installation option, enter the following command on the command line to run NetSetup in silence:

```
drive:\destination_path\netsetup\setup.exe -psclog[C:\<path-to-file>] -s
[-flC:\<path-to-file>\response-file]
```

drive:\destination_path\netsetup

The path to where the NetSetup utility resides on the server in the OpenEdge product file structure.

setup.exe

The command to run the NetSetup program.

-psclog[C:\<path-to-file>]

The required parameter to run NetSetup in Silent installation mode. This parameter is also optionally used to identify a path to a log file that contains information about the status of the silent installation.

The log file created by the installation program is called PscNetSetupMsg.log.

-s

The required parameter to run an installation without requiring user interaction. This parameter is executed with the setup.exe to run a silent installation.

-fl<path>\<response-file-name>

Specifies the name of the response file. By default, the install will look for the file setup.iss in the same directory as setup.exe is located.

The following example shows the typical contents of the PscNetSetupMsg.log file:

```
[Progress NetSetup Messages]
Type=INFORMATION
Date=6-23-2005
Time=10:02:46
File=setup.rul
Line=987
Message=Setup is complete. You may run the installed program.
=====
Type=INFORMATION
Date=6-23-2005
Time=10:02:46
File=setup.rul
Line=377
Message=Completed Successfully.
=====
```

Performing an OpenEdge Installation on UNIX or Linux

This chapter contains instructions for installing OpenEdge on UNIX or Linux, as outlined in the following sections:

- [Installation overview](#)
- [Additional product installation activities](#)
- [OpenEdge Silent installation overview](#)
- [Performing postinstallation tasks](#)
- [Performing a rolling upgrade of OpenEdge Management](#)
- [Uninstalling OpenEdge on UNIX and Linux operating systems](#)

Installation overview

After you have addressed all the topics presented in the “[Tasks overview](#)” section on page 68, you are ready to install OpenEdge on either a UNIX or a Linux platform.

Starting the installation process



To initiate the Installation Utility to install OpenEdge products:

1. Obtain a copy of the completed *Preinstallation Checklist for UNIX or Linux*. You might also want to have the other installation-related documents highlighted in [Table 17](#) available for reference.

Note: When you install a client networking license, the ADM2 directory is not installed in the `OpenEdge-install-dir/GUI` directory. This r-code is considered part of your application and should be deployed as a module of your application.

2. Close all other applications before beginning the installation process.

Other applications or tasks might interfere with the installation or they might use files that OpenEdge needs to complete the installation. Shut down any processes where the executable itself, or a file used by the executable, is located in the directory where you intend to install OpenEdge.

3. Log in as root. If you do not know the root password for your machine, check with your system administrator.
4. To install OpenEdge product installation program, Progress Software Corporation supports Electronic Software Distribution (ESD) download. Navigate to the software image you intend to download from the Progress Software Download Center.

Note: Electronic Software Distribution (ESD) download supports downloading software images from the Progress Download Center available at <http://www.progress.com/esd>. This Web site requires a valid account that your company must establish with Progress Software Corporation to access OpenEdge products and updates.

5. Download the software image and save it to your PC.
6. To start the installation, enter the following install command in the command prompt and press **Enter**:

```
./path/proinst
```

Note: `Path` refers to the directory where you have saved the software image. You cannot run `proinst` if you are in the directory where you have saved the software image or the intended installation directory.

If the JVM is ...	Then ...	Next ...
Found to be installed on your platform	The Welcome dialog box appears.	Proceed with the installation
Not found to be installed on your platform	The Installation Utility searches your \$PATH for it	If the JVM is then found in the \$PATH, the Welcome dialog box appears
	The JVM has not been detected Warning message appears	The installation continues, but you can only install products that do not require a JVM
	The JVM version does not match the version supported by OpenEdge Warning message appears	You can choose to continue with the installation whether or not you have the supported JVM version on your system, however, Progress Software Corporation recommends that you install the supported JVM version to ensure full functionality ¹

1. If you are performing a batch installation, you can add an entry to the `.ini` file to allow batch installs to override this warning. See the ["OpenEdge Silent Installation overview"](#) section on page 165 for more information.

The **Welcome** dialog box appears:

```

+-----+
+ Welcome +
+-----+
+
+ WELCOME TO THE OPENEDGE INSTALLATION UTILITY
+
+ Ensure that you have your completed "Preinstallation Checklist
+ for Unix" handy to consult. This information will facilitate your
+ installation and ensure your choices are accurately recorded.
+
+ Copyright (c) 1984-2007 Progress Software Corp.
+ All Rights Reserved.
+
+ [Enter=OK]
+-----+

```

8. Proceed to the “Performing the installation” section on page 157.

Performing the installation

Once you have loaded the installation program from your installation medium and the **Welcome** dialog box appears, you are ready to perform the online tasks required to install OpenEdge.

Refer to [Table 17](#) for the documents you should reference during the installation to help you perform the online OpenEdge installation.

Navigating though the Installation Utility

The Installation Utility is designed to programmatically present the dialog boxes for which you need to enter data, according to the products you are installing and the type of installation you choose to perform. Record your input on each dialog box and advance to the next dialog box at your own pace. The specific controls you use to advance to the next dialog box or return to a previous dialog box are identified on each dialog box. Highlight a menu option using the **SPACEBAR** key, the **TAB** key, the **CURSOR** keys, or the accelerator keys that are highlighted in each selection on the dialog box.

You can generally use the **Cancel** control to toggle back to a previous dialog box to review and/or update your choices to date. **Cancel** also allows you to quit the installation at any time before you commit to your selections. You are also given the option to not install any installation files at this time, and you can begin the installation process again at a later time.

Some dialog boxes also have unique buttons that allow you to complete a procedure or reset default values.

Accessing online help topics

Refer to the online installation help which contains a help topic for each installation dialog box. Access the online help according to the method identified on each dialog box; generally, you will enter the control-key sequence, or highlight the **Menu** option and press **ENTER**. Scroll in the help to view all of the dialog box explanatory and procedural details; press **ESC-ESC** to exit the help topic and return to the Installation Utility.

Installation-related messages

During the installation, additional questions, messages, or information related to certain dialog boxes might appear. Follow the instructions as presented.

Committing your installation choices

Once you are satisfied with all your selections, you can review a comprehensive list of your installation choices and commit them to be installed from the **Summary** dialog box.

Finishing the installation

Now that you are ready to understand, review, and make changes to the user environment to run OpenEdge, note the following:

- If you have installed an SQL product, you must set your environment variables. For more information on setting environment variables, either to support an SQL product or to customize the variables to your own preferences, see [Chapter 8, “Working in the OpenEdge Environment on UNIX.”](#)
- To create customized product executables, see the information on building ABL executables in [OpenEdge Deployment: Managing ABL Applications](#); creating executables might be required for certain product configurations.
- Address any other postinstallation tasks discussed in the [“Performing postinstallation tasks”](#) section on page 175.

Additional product installation activities

This section highlights the following additional product-related activities you might want to perform:

- [Using an electronic license addendum file](#)
- [Installing additional products](#)
- [Adding components to previously installed products](#)
- [Downloading executables for heterogeneous environments](#)

Using an electronic license addendum file

If you have obtained an electronic license addendum file, you can enter the information in the **Product Configuration Data** page. An electronic License Addendum file contains the serial numbers and control codes for the OpenEdge license you purchased. For instructions on obtaining an electronic license addendum file, see the [“Obtaining an Electronic License Addendum file”](#) section on page 72.



To enter the serial number and control code for your product automatically:

1. In the **Product Configuration Data** page, press **CTRL+A** to display the License Addendum File dialog box:

```

+-----+
|               Product Configuration Data               |
+-----+
| Company Name: _____ [Enter=Additional]           |
| Serial Number: _____ [Ctrl-E=Done]               |
| Control Number: _____ [Ctrl-T=Quit]               |
|                                     [Ctrl-N=Release Notes]|
|                                     [Ctrl-V=View]        |
|                                     [TAB=Next Field]     |
|                                     [Ctrl-P=Help]        |
|                                     [Ctrl-A=Addendum File]|
+-----+

```

2. In the **License Addendum File** dialog box, enter the name and path of the License Addendum file in the **Enter Path** field:

```

+-----+
|               License Addendum File                   |
+-----+
| Enter Path: █_____ |
|                                     [Enter=OK] [Ctrl-N=Cancel]|
+-----+
|                                     [Ctrl-V=View]        |
|                                     [TAB=Next Field]     |
|                                     [Ctrl-P=Help]        |
|                                     [Ctrl-A=Addendum File]|
+-----+

```

3. Press **Enter**.

Once the license addendum file is validated, the **Entered Product List** is automatically populated.

Note: Press **CTRL+V** to view the **Entered Product List**. You cannot remove loaded products during a UNIX or Linux installation (unlike Windows.)

Installing additional products

Once OpenEdge is successfully installed, you can choose to add additional products to your current installation.

To initiate this process, you must re-load your installation media. First perform the steps outlined in the [“Starting the installation process”](#) section on page 156, and then perform the steps outlined in the procedure that follows.

Note: When you add products to an existing installation, you can use the Installation Utility in batch mode as long as you are performing a Complete installation of the products you are adding. For more information about a batch installation, see the [“OpenEdge Silent installation overview”](#) section on page 165.



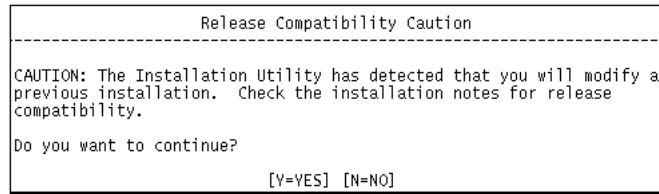
To install additional products when the Welcome dialog box appears:

1. Press **RETURN** to continue. The **Serial & Control Numbers** dialog box appears.
2. Enter only the control numbers for the products you are adding to the list of previously installed products.
3. When you are done, press **CTRL+E**. The **Done Configuration Data Confirmation** dialog box appears.
4. Press **Y** to continue (or press **N** to add more products). The **Type Device and Destination** dialog box appears.
5. Choose **Select the Destination Pathname**, and type the path of the initial installation.
6. Press **RETURN**. The **Destination Pathname Exists** dialog box appears:

```
-----
|                                     |
|               Destination Pathname Exists               |
|-----|-----|
| Select an alternate destination path |
| Erase the current destination path  |
| Install the OpenEdge products in the pre-existing destination path |
| Help                               |
|-----|-----|
```


7. Choose **Install the OpenEdge products in the pre-existing destination path** and press **RETURN** to continue with the installation.

If you install products that will affect previously installed products, you might see the following caution message:



8. Choose **Yes** to continue with the installation.

The installation program adds your OpenEdge products to your directories automatically.

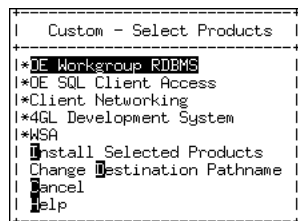
Adding components to previously installed products

You can add components and subcomponents to existing OpenEdge installations without having to enter any data other than the required components or subcomponents.



To add components or subcomponents using the Add feature:

1. At the command line, type the shell script `destination-path/proaddcomp` to run the add feature. The **Select Products** dialog box appears:



All previously installed products appear on this list. The **Select Products** dialog box allows you to select and deselect OpenEdge products for which you want to add components or subcomponents.

2. Select or deselect a product by highlighting the product and pressing **RETURN**. An asterisk (*) indicates that a product is selected. If you want to select the first product on the list, you must first press **RETURN** to deselect the product and then press **RETURN** again to select it. When you select a product the **Select Components** dialog box appears:

```

+-----+
| Custom - Select Products |
+-----+
|*DE Workgroup RDBMS      |
|*DE SQL+                 |
|*Client Custom - Select Components |
|*4GL De+                 |
|*WSA | Oracle DataServer Client - Optio |
| Install OpenEdge SQL ODBC Clients |
| Change OpenEdge SQL JDBC Clients |
| Cancel OpenEdge ESQL/C Clients |
| Help |*Open Client Adapter Options Basic (r) |
|-----|*DEBuild Utility (r) |
|         | Client-Side Web Services Deploy (r) |
|         | Previous Menu |
|         | Cancel |
|         | Help |
+-----+
| Disk Space Required for Products: 541,597,696 bytes |
| Disk Space Required for Installation: 542,949,888 bytes |
| Disk Space Remaining After Installation: 4,294,967,295 bytes |
| Destination pathname: /usr1/bhamel/101b |
+-----+

```

The **Select Components** dialog box lists only those components that have not been previously installed. Select or deselect a component to install by highlighting the component and pressing **RETURN**. An asterisk (*) indicates that a component is selected.

3. If the selected component has subcomponents the **Select Subcomponents** dialog box appears:

```

+-----+
| Custom - Select Products |
+-----+
|*DE Workgroup RDBMS      |
|*DE SQL+                 |
|*Client Custom - Select Components |
|*4GL De+                 |
|*WSA | Oracle DataServer Client - Optio |
| Install Ope+             |
| Change Ope| Custom - Select Subcomponents |
| Cancel Ope+             |
| Help |*OpenAppServer Internet Adapter (r) | |
|-----|*DEB|*OpenEdge Adapter for SonicMQ (r) |
|         |*Cli|*Java Client Support (r) |
|         | Pre| Previous Menu |
|         | Can| Cancel |
|         | Hel| Help |
+-----+
| Disk Space Required for Products: 541,597,696 bytes |
| Disk Space Required for Installation: 542,949,888 bytes |
| Disk Space Remaining After Installation: 4,294,967,295 bytes |
| Destination pathname: /usr1/bhamel/101b |
+-----+

```

The **Select Subcomponents** dialog box lists the subcomponents for the component you selected. The symbol (r) indicates that a subcomponent is recommended and will be installed automatically unless you deselect it. Mandatory components are not displayed.

Select or deselect a subcomponent by highlighting the component and pressing **RETURN**. An asterisk (*) indicates that a subcomponent is selected.

Choose **Previous Menu** and press **RETURN** when you have selected all the subcomponents you want to add.

4. Choose **Install Selected Products** from the **Select Products** dialog box and press **RETURN**. The **Done Selecting Products Confirmation** dialog box appears:

```

+-----+
|           Done Selecting Products Confirmation           |
+-----+
| Are you sure that you are done selecting products to install? |
|                                     [Y=YES] [N=NO]          |
+-----+

```

5. Type **Y** to continue with the installation or **N** to select additional components or subcomponents. The **Copy Scripts?** dialog box appears:

```

+-----+
|           Copy Scripts?           |
+-----+
| Copy the scripts to /usr/bin? |
|                                     [Y=YES] [N=NO]          |
+-----+

```

OpenEdge products consist of several scripts and program modules. When you install a product, the scripts are placed in the installation directory you specify.

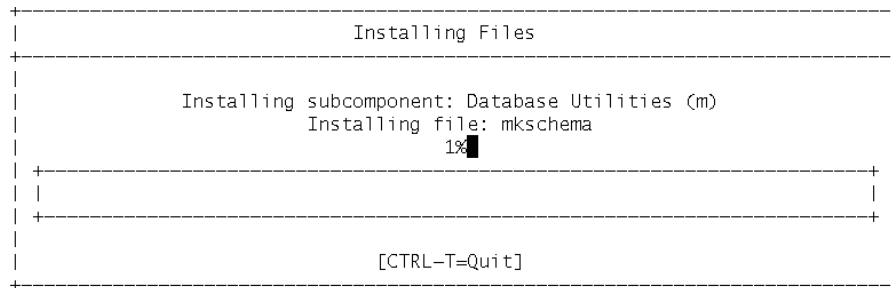
6. Choose one of the following:
 - To allow all users on your system to run the product, you should answer **Yes** when prompted to copy the scripts to `/usr/bin`. Type **Y** to instruct the Installation Utility to place OpenEdge scripts in `/usr/bin` **and** in the destination pathname you specified earlier.

Caution: Answering **Y** might cause the OpenEdge Installation Utility to overwrite existing executables in this directory.

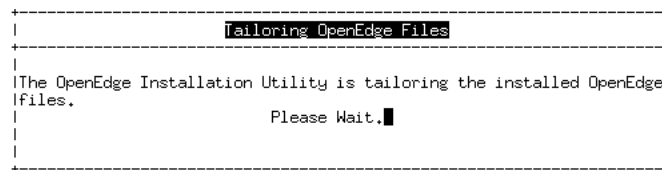
- Type **N** to instruct the Installation Utility to place OpenEdge scripts in the destination pathname you specified earlier.

Note: If you are maintaining two versions of OpenEdge on the same machine, answer **N** to this question.

While OpenEdge decompresses the files, the **Installing Files** dialog box appears:



While OpenEdge tailors the files, the **Tailoring OpenEdge Files** dialog box appears:



7. Press **RETURN**. When the installation is complete, OpenEdge returns you to the UNIX system prompt.

Downloading executables for heterogeneous environments

The distributed architecture of OpenEdge allows you to optimize your hardware and network resources by installing components across networked machines, specifically when you are installing products such as the WebSpeed Transaction Server and the AppServer. Although some of these products' components must reside together on the same machine, in some cases, you can distribute components to different machines, even if the machines have different hardware or operating systems. For example, you can install a WebSpeed Messenger or the NameServer on a UNIX platform and install a WebSpeed Broker and agents in Windows.

If you need either the WebSpeed Messenger executable or the NameServer executable for a platform other than UNIX, you can download the executables free of charge from <http://www.progress.com/esd>.

OpenEdge Silent installation overview

An interactive installation prompts you for input and records your values in a series of dialog boxes. The installation program immediately uses this data to install your OpenEdge products.

In contrast, a Silent installation is a two-step process:

- Data entered during the interactive installation process is recorded, typically in an `.ini` file. An OpenEdge installation automatically creates a `response.ini` file during the interactive installation. Although you can create your own `.ini` file, the automatically-generated `response.ini` file is a more reliable data input to perform a Silent installation.

Note: This section focuses primarily on using the `response.ini` file because this data input does not require you to perform any additional file-related tasks. Optional response file-related activities, such as editing a response file, are discussed later in this section.

- The installation data captured in an `.ini` file is read programmatically to install the products through a batch, or silent, mechanism at any time. Complete and custom installation support the Silent installation feature.

The main tasks to perform a Silent installation are:

- Selecting which `.ini` file to use to capture your installation values
- Entering the command to start the Silent installation
- Checking the status of the installation log

Data input options for a Silent installation

[Table 25](#) identifies and briefly describes the two types of data inputs you can use to perform a Silent installation.

Table 25: Data input options for a Silent installation

Data input options	Description
Automatically generated <code>response.ini</code> file	<p>An OpenEdge 11.3 interactive installation automatically creates a <code>response.ini</code> file that contains the installation values you originally entered in fields on the dialog boxes. It is stored in your <code>install</code> subdirectory in your installation directory, <code>OpenEdge-install-dir</code>. The file is immediately available for you to play back to start a Silent installation.</p> <p>See the “Understanding the response.ini file contents” section on page 166 for more information and an excerpt of the <code>response.ini</code> file.</p>
User-initiated programmatic method	<p>Provides Application Partners (APs) a streamlined approach to integrate the OpenEdge installer into an application installer. Using this method, an AP can access the automatically generated <code>response.ini</code> file to programmatically create an OpenEdge installation response file. When the AP's application is installed on a customer site, the OpenEdge installation information is read from the response file, enabling the customized install to be performed silently.</p> <p>For more information about this method, see the “Creating data input option” section on page 174.</p>

Note: You can choose to edit the response file. However, keep in mind that any modifications to the automatically- or programmatically-generated response file can be time consuming and error prone.

Understanding the response.ini file contents

The data captured in the automatically-generated `response.ini` file provides a detailed, reliable snapshot of the installation choices made during an interactive installation.

The `response.ini` file includes:

- A header version number and application details
- Section labels defined by brackets for easy referencing
- Each dialog box comment section identified with the label `DESCRIPTION` and the specific dialog box title
- Easy-to-read descriptions of the fields of each dialog box

- Only the values captured during the interactive install are stored in the `response.ini` file; there is no extraneous content
- Dialog boxes that appear in the same order as in the online installation
- A complete list of products installed

The original `response.ini` file is initially created when you run the Silent installation; it is never overwritten. If you re-run the Silent installation to add products to an existing 11.3 installation, a new unique `response.ini` file is created. It is identified as `response.ini.1`, `response.ini.2`, `response.ini.3`, and so forth. These files will be saved to your installation directory.

Response.ini sample excerpt

The following example shows an excerpt from the automatically-generated `response.ini` file:

response.ini

(1 of 5)

```
;
; DESCRIPTION of Configuration Count
;
; ProductCount - the number of products being installed.
;

[Configuration Count]

NumberOfConfigurations=3


[Product Configuration 1]
name=PSC
serial=123456789
version=11.3ALPHA
control=Z9ASH UPSUM K4HY4
prodname=OE Enterprise RDBMS


[Product Configuration 2]
name=PSC
serial=123456789
version=11.3ALPHA
control=Y8CR? YQQUC KMH?M
prodname=Progress Dev Studio OE


[Product Configuration 3]
name=PSC
serial=123456789
version=11.3ALPHA
control=Y9CSH 7QS3C 34DY8
prodname=OpenEdge Replication
;
```

response.ini

(2 of 5)

```

; DESCRIPTION of Configuration Count
;
; ProductCount - the number of products being installed.
;

;
; DESCRIPTION of OpenEdge Explorer
;
; enable - indicates whether or not you want to enable the OpenEdge Explorer
functionality.
; - a value of false indicates you do NOT want to enable the OpenEdge Explorer
functionality.
; - a value of true indicates you want to enable the OpenEdge Explorer
functionality.
;

[OpenEdge Explorer]

;
; enable - indicates whether or not you want to enable the OpenEdge Explorer
functionality.
; - a value of false indicates you do NOT want to enable the OpenEdge Explorer
functionality.
;
; - a value of true indicates you want to enable the OpenEdge Explorer
functionality.
;

[OpenEdge Explorer]
enable=true

;
; DESCRIPTION of Java
;
; jdkHomePath - the root directory where the Java Development Kit is installed
; jvmAllowUnsupported - this setting allows an install to continue when the
detected java version is different than what was used for testing
;

[Java]
jdkHomePath=/jvms/Linux/jdk1.6.0_25
jvmAllowUnsupported=yes

;
;

DESCRIPTION of Type and Destination
;
;

path - identifies the directory in which you install your OpenEdge product
software.
; workpath - identifies the directory in which your applications, databases,
and log files will reside.
; oem_path - identifies the directory in which you install your Management
product software.
; oem_workpath - identifies the directory in which your Management
applications, databases, and log files will reside.

```


response.ini

(3 of 5)

```

;
[Type and Destination]
type=COMPLETE
path=/usr/dlc
workpath=/usr/wrk
oem_path=/usr/oemgmt
oem_workpath=/usr/wrk_oemgmt

;
; DESCRIPTION of Server Engine
;
; UseSqlServerEngine - valid values are 0 and 1.
;   0 - indicates that the SQL Database Engine is to not be installed.
;   1 - indicates that the SQL Database Engine is to be installed.
;

[Server Engine]
UseSqlServerEngine=1

;
;

DESCRIPTION of SonicEsbAdapter;
;

esbdomain - identifies the Sonic ESB Domain Name.
;   esburl - identifies the Connection URL to the Sonic ESB.
;   esbusername - identifies the User Name used to connect to the Sonic ESB.
;   esbpassword - identifies the Password used to validate the User Name.
;   esbpath - identifies the directory where the Sonic ESB is installed.
;   esbcontainername - identifies the Sonic ESB Container Name.
;

[SonicEsbAdapter]
esbcontainername=vmhydoerh5lx-02Container
esbdomain=Domain1
esburl=tcp://localhost:2506
esbusername=Administrator
esbpassword=Administrator
esbpath=empty

;
;

;DESCRIPTION of Language Default
;
;
DefaultLanguage - identifies the language in which PROMSGS appears by default.
;   -Valid values are:
;       Czech
;       Dutch
;       English - American
;       English - International
;       French
;       German
;       Italian
;       Polish
;       Portuguese
;       Portuguese - Brazilian
;       Spanish
;       Spanish - Latin
;       Swedish
;

```

response.ini**(4 of 5)**

```

[Language Default]

DefaultLanguage=English - International


[Language Choice]
lang1=English - International

;
;
DESCRIPTION of International Settings
;
;

;  cinternal - identifies the -cinternal and -cpstream values included in
the startup.pf file.
;  cpcollation - identifies the -cpcoll value included in the startup.pf file.
;  cpcase - identifies the -cpcase value included in the startup.pf file.
;  dateformat - identifies the -d value included in the startup.pf file.
;  numsep - identifies the -numsep value included in the startup.pf file.
;  numdec - identifies the -numdec value included in the startup.pf file.

[International Settings]

cinternal=ISO8859-1
cpcollation=Basic
cpcase=Basic
dateformat=mdy
numsep=44
numdec=46

[Installed Products]

ProductCount=3

Product 105=OE Enterprise RDBMS

Product 157=Progress Dev Studio OE
Product 21=OpenEdge Replication
[Product 105]
__SubComponent Compile Tool - CHAR=1
__SubComponent Oracle Client=1
__SubComponent ADM Runtime - CHAR=1
__SubComponent Client-Side Security (r)=1
__Component OpenEdge SQL ODBC Clients=1
__Component OpenEdge SQL JDBC Clients=1
__Component OpenEdge ESQ/C Clients=1
__Component Open Client Adapter Options Basic (r)=1
__SubComponent AppServer Internet Adapter (r)=1
__SubComponent OpenEdge Adapter for SonicMQ (r)=1
__SubComponent Java Client Support (r)=1
__Component OEBuild Utility (r)=1
__Component Client-Side Web Services Deploy (r)=1
__SubComponent Web Services Basic (r)=1
__SubComponent Web Services Schema (r)=1

```

response.ini

(5 of 5)

```
;[Product 157]
__Component_Oracle DataServer Client - Optio=1
__Component_SQL Database Server - Optional=1
__SubComponent_Client-Side Security (r)=1
__Component_OpenEdge SQL ODBC Clients=1
__Component_OpenEdge SQL JDBC Clients=1
__Component_OpenEdge ESQL/C Clients=1
__Component_Open Client Adapter Options (r)=1
__SubComponent_AppServer Internet Adapter (r)=1
__SubComponent_OpenEdge Adapter for SonicMQ (r)=1
__SubComponent_Java Client Support (r)=1
__SubComponent_OpenEdge Adapter for SonicESB (r)=1
__SubComponent_Web Services Admin Enabler (r)=1
__SubComponent_Web Services Schema (r)=1
__Component_Client-Side Web Services (r)=1
__SubComponent_Web Services Basic (r)=1
__SubComponent_WSDL Analyzer (r)=1
__Component_Application Debugger (r)=1
__SubComponent_Application Debugger (r)=1
__Component_OEBuild Utility (r)=1
__Component_Progress Developer Studio Development (r)=1
__SubComponent_Progress Developer Studio (r)=1
__Component_4GL utilities (r)=1
__SubComponent_XSD-4GL (r)=1

[Product 21]
__SubComponent_Client-Side Security (r)=1
```

Running the Silent installation

The command you use to initiate, or play back, the response file is the same regardless of the data input you use. Once you enter the command to start the process, the OpenEdge Silent installation utility runs without your intervention.

The syntax for running the OpenEdge Silent Installation utility in batch mode follows:

Syntax

```
proinst -b <path>/<install-ini-name> -l <path>/<logfile-name> [-n]
```

proinst

The command to initiate an OpenEdge installation.

`-b<path>/<install-ini-file>`

Identifies that a batch installation will be performed, and specifies the pathname and filename of the `.ini` file that you select to run the Silent installation. You can use the `response.ini` file, the `install.ini` file, or another `.ini` file that you create and name.

`-l <path>/<logfile-name>`

Identifies that a log file will be created, and specifies the pathname and filename of the installation log file in which the installation events will be logged. If no filename is specified, the OpenEdge Installation Utility uses the default log filename of `install.log`.

If no directory is specified to which the log file is to be saved, the Installation Utility saves it to the directory identified by the first environment variable it finds among the following: `$TMP`, `$TEMP`, or `$TMPDIR`.

`-n`

Indicates that the batch installation will include a progress meter, displaying details about the current installation phase and percent complete.

Example

The following example shows a typical Silent installation command:

```
proinst -b /test/install.ini -l /log/test.log
```

Checking the status of the Silent Installation log file

The Silent Installation process automatically generates a log file, in which all messages—error and successful installation—are reported.

The following is an excerpt from the typical contents of the OpenEdge Installation Utility log file:

OpenEdge Installation Utility log file

```

OPENEDGE INSTALL UTILITY LOG <VERSION 11.3> (Wed Sep 27 11:30:52 2006)
[Application]
Name=OpenEdge
Version=11.3
Company=progress

[DetectPreviousInstallSettings]
Information=[09-28-2011 15:07:59] Unable to locate file /etc/progress.

[ResponseResult]
ResultCode=0
ResultDescription=The install completed successfully.

[CompletedEvents]
Event1=[09-28-2011 15:10:34] The Setup Utility is extracting archives

Event2=[09-28-2011 15:11:32] The Setup Utility has extracted archives

Event3=[09-28-2011 15:11:35] The Setup Utility has tailored files.

Event4=[09-28-2011 15:11:37] The Setup Utility has merged delta files.
[RuntimeStatus]
Progress=98
.
[UpdateUnixRegistry]

File=[09-28-2011 15:11:34] /etc/progress has been created successfully.
.
.
.
[FilesTailored]
File1=[09-28-2011 15:11:34] /usr/dlc/bin/OE_TC has been tailored
successfully.
File2=[09-28-2011 15:11:34] /usr/dlc/bin/proenv has been tailored
successfully.
.
.
.
[TailoringExtensions]
Extension1=[09-28-2011 15:11:41] /usr/dlc/bin/rptlr.dll has been executed
successfully.

[TailoringClasses]
Start=[09-28-2011 15:11:41]
Finish=[09-28-2011 15:11:55]

```

Optional data input activities

The following optional activities are also supported when you are performing a Silent Installation. However, keep in mind that creating the response file manually or editing the response file are more time-consuming and potentially error-prone approaches than creating it using the automatically-generated response file method described in the [“Understanding the response.ini file contents”](#) section on page 166.

Creating data input option

You can choose to record a separate response file any time you perform an interactive installation. All your installation choices are automatically recorded in a user-defined response file. If you do not specify a filename, the install creates the file `$TEMP/install.ini`.

The format and structure of any data input option is identical to that which is presented in the automatically-generated `response.ini` file. See the [“Response.ini sample excerpt”](#) section on page 167 to review an excerpt of the file’s content.

Use the following syntax to initiate a response file:

Syntax

```
proinst -r [<path-to-file>\<response-file>]
```

`proinst`

The command to initiate an OpenEdge installation.

`-r <path-to-file>\<response-file>`

Indicates that the installation is in record mode, and specifies a pathname to and filename for the data input file to be created. If you do not provide a filename, the installation creates the filename, `install.ini` and places it in the `$TEMP` directory.

Manually modifying data input option

You can edit any response file, whether you create it or use an automatically-generated response file. Although all sections of the response file are required, you do not need to add each of these required sections in the order presented. The installer only retrieves the specific data it needs regardless of where the information is located in the response file.

Addressing a detected JVM version

If you receive a warning message at the beginning of your installation stating that the detected JVM version does not match the version supported by OpenEdge, you can add an entry in the `.ini` file to allow batch installs to override this warning. Add the following entry to the `[java]` section of the `.ini` file if you want the installation to continue after detecting a mismatched JVM version:

```
jvmAllowUnsupported=yes
```

Performing postinstallation tasks

Before you run OpenEdge, you need to complete a few required postinstallation tasks, as listed:

- **Set environment variables** — For more information on setting environment variables (including SQL), see [Chapter 8, “Working in the OpenEdge Environment on UNIX.”](#)
- **Create customized executables** — To create customized product executables, see the information on building ABL executables in [OpenEdge Deployment: Managing ABL Applications](#); creating executables might be required for certain product configurations.
- **Convert existing databases** — After your OpenEdge installation is complete, you must convert your Progress databases to OpenEdge using the PROUTIL CONV910 utility. Note that if you have a Progress Version 8 database, you must convert it to a Version 9 database first. For instructions on converting your Progress databases to OpenEdge, see the chapter on administration utilities in [OpenEdge Data Management: Database Administration](#).

Setting AdminServer security

Once you have installed OpenEdge with products that use the AdminServer, you can optionally set the user and/or group AdminServer security. You set this option on the command line to require an individual user and/or groups of users to provide valid values during the AdminServer startup process. OpenEdge products such as the following use the AdminServer: AppServer, WebSpeed, OpenEdge Adapter for SonicMQ, OpenEdge Management or OpenEdge Explorer, and Web Services Adapter.

The AdminServer user-group authorization feature allows you to require a level of security that enables only authenticated operating systems users and groups access to, and use of, the Admin Service.

To effectively set up this security option for your AdminServer use, review your security needs and current authenticated operating system users and groups to determine how you will set up this option during the OpenEdge installation process.

To implement the User-group Authorization feature on a UNIX platform, you must first successfully complete the installation program.

Table 26 identifies and briefly describes the purpose of each command-line option.

Table 26: User-group parameter options

Parameter name	Syntax	Purpose
Individual user name and password required	<code>-requireusername</code>	Requires a minimum of one user ID to be resolved for each AdminServer operation before it can be executed.
Group authorization required	<code>-admingroup group</code> <code>[{, :}group...]</code>	Requires a minimum of one group to be resolved for each AdminServer operation before it can be executed. On a UNIX platform, a colon-separated list differentiates groups when you are specifying multiple groups on the command line.

On UNIX platforms, a group name can be any user-defined or NIS group name. UNIX can also support subgroups.

Performing a rolling upgrade of OpenEdge Management

This section provides information about using multiple consoles to upgrade OpenEdge Management. You can run several instances of the OpenEdge Management remote monitoring console on a single Linux or UNIX system; this is useful when you need to support a phased set of updates to remote OpenEdge Management locations.



To perform a rolling upgrade on multiple consoles, you must first install the latest versions of OpenEdge and OpenEdge Management on your monitoring system:

1. Install OpenEdge components to a new directory.
2. Install OpenEdge Management to a new directory.

Keep the following in mind when installing and configuring:

- To perform a rolling upgrade, ensure that you have the most recent release of OpenEdge and OpenEdge Management. For additional information, see [Chapter 3, “OpenEdge Installation Prerequisites”](#).
- You might encounter TCP/IP port conflicts due to multiple versions of OpenEdge and OpenEdge Management. For more information, see the [“Typical TCP/IP configuration with a hard disk on each machine”](#) section on page 519.
- Sufficient memory is required on the management console. For more information, see [Chapter 6, “Administration Utilities.”](#)

Making port updates

To avoid port conflicts between multiple installations, you will need to change some port settings. [Table 27](#) illustrates default port numbers used by OpenEdge and OpenEdge Management.

Table 27: Port defaults

Component	Symbol	Comment
AdminServer	20931	AdminServer
adminport	7839	adminport
Nameserver and unified broker	5162	Nameserver and unified broker
Database agent	8839	Database agent
Fathom Web Server	9090	Fathom Web Server
Fathom remote monitoring	6835	Fathom remote monitoring

When performing a rolling upgrade, make the following port updates:

- In **PluginPolicy.Progress.AdminServer**, change the `AdminServer` port and the `adminport` in `$DLC/properties/AdminServerPlugins.properties` to the following:

```
adminserver port=20955
adminport=7901
```

- Change the `NameServer` and `Unified broker` ports in the `$DLC/properties/ubroker.properties` file; specify any port number other than the default values.
- Change the *database agent* port in `$DLC/properties/agent.properties` to the following:

```
database agent port=8839
```

- Change the `Fathom Web` server port in `FATHOM/config/fathom.properties` by modifying the `[webserver]` section to include the following:

```
httpport=9095
```

- Execute the `fmconfig -enable -port <value>` command to enable monitoring for the Fathom remote port.

Installing a new console

After upgrading the software on your monitoring system, the next step in a rolling upgrade is to install a new console.



To install a new console:

1. Install OpenEdge components to a new directory.
2. Install OpenEdge Management to a new directory.
3. Create a new `fathom.properties` file in the new `FATHOM/config` directories and define the `httpport`.

Note: When defining a new port number for `httpport`, use a different number than 9095.

4. Copy the `FATHOM/db/fathom.o*` files from the old console to the new console.

5. Start the new AdminServer and configure OpenEdge Management.
6. Within OpenEdge Management, configure the following:

In the **Trend database location**, select **Store trend data in a remote Fathom database**. In the **Remote database hostname** and **Remote Fathom web server port** fields, specify the location of the existing OpenEdge Management application.

7. To enable remote monitoring, stop the new AdminServer and execute the following command:

```
fmconfig -enable -port 6836
```

Once you have installed the new console, you can bind remote servers to the new installation.

Upgrading a remote container

After installing a new console, the final step in a rolling upgrade is to upgrade your remote container.



To upgrade a remote container:

1. Ensure that the newly installed AdminServer console is running and is enabled for remote monitoring (using the `fmconfig` command), and ensure that OpenEdge Management is configured.
2. Stop the older version of the AdminServer.
3. Disable remote monitoring on the remote AdminServer:

```
fmconfig -disable
```

4. Configure the remote monitoring console to match the port number and hostname specified on the new console:

```
fmconfig -enable -port <portnumber> -host <hostname>
```

5. Restart the AdminServer.

Note: The AdminServer should connect to the new OpenEdge Management console. The previously configured OpenEdge Management container remains, but will be offline.

After configuring the remote container, you can monitor it from the new console. To migrate additional containers, execute the `fmconfig -enable -port <portnumber> -host <hostname>` command where the port number and hostname represent new AdminServer consoles.

Note: You must restart the remote AdminServer each time you migrate additional containers.

Uninstalling OpenEdge on UNIX and Linux operating systems

The `uninstall` script consolidates and formalizes the actions required to remove an OpenEdge 11.3 installation from all supported UNIX or Linux operating systems. The `uninstall` script is installed in the `install` subdirectory located within the `OpenEdge-install-dir`.

If you installed OpenEdge Management, stop the OpenEdge Management Trend Database before uninstalling. You can use OpenEdge Management or OpenEdge Explorer or the following command:

```
dbman -stop FathomTrendDatabase
```

The AdminServer must be running in order to stop the OpenEdge Management Trend Database. If you receive a warning during the uninstall that the `fathom.db` is in use, the OpenEdge Management Trend Database has not been stopped.

Caution: If you want to save trending data, be sure to copy the `<OpenEdgeManagement-install-dir>/db` before removing the OpenEdge Management installation.

If you want to save all the customized monitoring plan and resource definition information, be sure to copy `<OpenEdgeManagement-install-dir>/config/fathom.odb` before removing the OpenEdge Management or OpenEdge Explorer installation.

The following syntax identifies the command executed to perform the uninstall process:

Syntax

```
cd OpenEdge install-dir/install/uninstall
```

Progress Software Corporation recommends using the uninstall script to ensure the following uninstall activities occur properly:

- Uninstall third-party, embedded products.
- Remove Release-specific entries from the `/etc/progress` file and `/etc/ProDbgCK` (Progress Debugger Check) files, and remove the install directory where OpenEdge 11.3 was installed. In addition, the uninstall script creates an `OE<version>uninst.log` file in the home directory.

Uninstalling OpenEdge Replication

OpenEdge Replication provides a utility to properly uninstall the product.



To uninstall OpenEdge Replication:

1. Enter the following command from your OpenEdge Replication `/bin` directory:

```
./repl_unglue
```

The following warning appears:

```
WARNING WARNING WARNING WARNING WARNING WARNING WARNING WARNING WARNING

The Replication unglue script will disassociate the Progress product
version previously associated with the installation of Replication on this
machine. Choosing to do so will result in Replication not being able to
run on this machine, do you wish to continue? [y | n]
```

2. Choose **y** to unglue OpenEdge Replication.

Unglue removes all OpenEdge Replication-specific files from the OpenEdge directory, but it does not remove the OpenEdge Replication product itself.

3. To remove the OpenEdge Replication product, you must delete the OpenEdge Replication directory tree.

Manually removing earlier OpenEdge versions

OpenEdge version 11.3 contains an `uninstall` command you can use to safely remove your software. To remove previous versions (for example, version 9) of OpenEdge, you must manually uninstall components.



To manually remove earlier OpenEdge versions:

1. Log in under the same domain and user name you used when installing OpenEdge.
2. Ensure that OpenEdge is not running, and close all OpenEdge processes, including any online Help files you might have open.
3. Delete the OpenEdge program directory, including all of its subdirectories.
4. Shutdown any Web server running on your system and delete any OpenEdge-specific Web server files (such as `cgiip.exe` and `wsisa.dll`) from the Web server `cgi-bin/scripts` directory.
5. Reboot your machine and follow the installation instructions in the [“Performing the installation”](#) section on page 157.

Administration Utilities

This chapter provides step-by-step instructions to perform a variety of administrative tasks and details related to using and managing platform-specific resources, as outlined in the following sections:

- [Using the License Update utility](#)
- [Displaying license information using the SHOWCFG utility](#)
- [Managing user licenses on all supported platforms](#)
- [Removing license for installed product](#)
- [OpenEdge license information](#)
- [Using OpenEdge resources in Windows](#)
- [Manage memory and system configurations on UNIX platforms](#)
- [UNIX troubleshooting tips](#)
- [OpenEdge event logging](#)

Using the License Update utility

Use the License Update utility to review and, as needed, change the following license information: number of licensed users, the expiration date for an OpenEdge product, and/or update your evaluation license to an OpenEdge non-evaluation license. (Note that the License Update utility is also called the Product Update utility.)

Contact your Progress Software Corporation sales representative for a new *License Addendum* if you need to use this utility.

Note: For information on installing OpenEdge components using the License Addendum File, refer to the [“Using an Electronic License Addendum file”](#) section on page 116 for installing in Windows, and the [“Obtaining an Electronic License Addendum file”](#) section on page 72 for installing on UNIX.

Changes to accommodate license updates

The License Update process streamlines the process of updating an existing product license. You can now enter a different, but valid serial number (and associated, new control numbers) to update an existing license file (`progress.cfg`). In prior releases, the update process would not accept different serial numbers; you had to uninstall an existing license and then install the newer product license.

This new update process can be used to update licenses obtained through either the product evaluation process or PSDN subscription renewal process. You simply enter a new product serial number during the installation process to automatically update the current license data in the `progress.cfg`. If want to update an evaluation license to a non-evaluation license, you no longer have to uninstall the evaluation license and then install the non-evaluation license. You can perform the procedure to update the License Update utility by entering your new valid serial and control codes.



To use the License Update utility to update your license in Windows:

1. Use the Show Configuration utility (`SHOWCFG`) to display the product license information for each OpenEdge product installed on your system. See the [“Displaying license information using the SHOWCFG utility”](#) section on page 186 for instructions.
2. Choose the **License Update** icon from your OpenEdge group. After a welcome message appears, the **Serial & Control Numbers** dialog box appears.
3. Type the serial number and the new control numbers that Progress Software Corporation supplies when you upgrade your license.
4. Choose **Accept**. The **Product(s) Updated** dialog box displays the products you want to update. When you are finished updating the products, choose **Done**.



5. Type your company name, the serial number, and the new control numbers Progress Software Corporation (PSC) supplies when you upgrade your license.

Press **ENTER** again to return to the **Product Configuration Data** dialog box.

Displaying license information using the SHOWCFG utility

The OpenEdge installation program prompts you to enter product information contained in your *License Addendum*. During the installation process, the installation program records the license information in the OpenEdge configuration file (`progress.cfg`). You can use the `SHOWCFG` utility to display licensing information, product installation, and configuration details about each OpenEdge product that you install.

Using the SHOWCFG utility in Windows

The `SHOWCFG` utility displays product installation and configuration information for each OpenEdge product installed on your system. [Table 28](#) describes the different ways to run the `SHOWCFG` utility.

Table 28: Running the `SHOWCFG` utility

To run the <code>SHOWCFG</code> utility from the ...	Then ...
Start menu	Choose OpenEdge→Config
OpenEdge Group menu	Double-click the Config icon
Command line of the Proenv window	Type the <code>showcfg</code> command

The `SHOWCFG` utility opens the **OpenEdge Configuration Information** dialog box to display the product configuration information stored in the OpenEdge configuration file `progress.cfg`. This file is created and modified during product installation.

[Figure 1](#) shows a typical display of the **OpenEdge Configuration Information** dialog box.

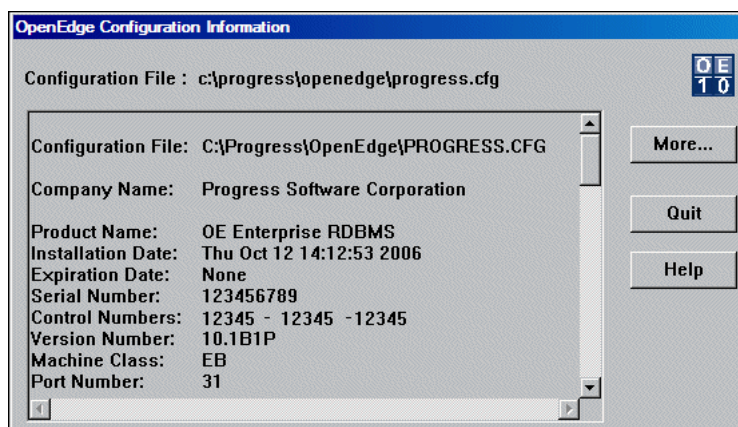


Figure 1: OpenEdge Configuration Information dialog box from the `SHOWCFG` utility

Table 29 identifies and briefly describes the detailed information that appears for each OpenEdge product you install on your system.

Table 29: Display fields associated with the SHOWCFG utility

This display field ...	Identifies the ...
Product Name	Name of the installed product
Installation Date	Date the product was installed
Expiration Date	Date the license expires
Serial Number	Number associated with the license agreement
Control Numbers	Numbers used by the OpenEdge installation software
Version Number	Software product version
Machine Class	Tier code associated with the license agreement
Port Number	Platform on which the software product is installed

Details about SHOWCFG functions in Windows

SHOWCFG searches for the `progress.cfg` file in the locations defined as `PROCFG` and `DLC` in the `progress.ini` file. To find the `progress.ini` file, SHOWCFG searches the following locations, in the order shown:

1. The current working directory
2. The special users directory (set using the Properties option)
3. The Windows directory

If the utility finds the `progress.cfg` file, it displays the contents in the **OpenEdge Configuration Information** dialog box.

If SHOWCFG cannot find the `progress.cfg` file, the **Open** dialog box appears so you can specify the file's location. You can also use the **Open** dialog box to specify a different OpenEdge configuration file to display. To display the **Open** dialog box, choose the **More** button in the **OpenEdge Configuration Information** dialog box.

OpenEdge does not accept optional qualifiers that point to a `.cfg` file other than `OpenEdge-install-dir:PROGRESS.CFG`.

Using the SHOWCFG utility on UNIX or Linux platforms

The SHOWCFG utility has the following syntax:

Syntax

```
<OpenEdge-install-dir>/bin/showcfg <OpenEdge-install-dir>/progress.cfg
```

For example:

```
/userdir/smith/101b/bin/showcfg /userdir/smith/101b/progress.cfg
```

The SHOWCFG utility displays the product configuration information stored in the OpenEdge Release 11.3 configuration file `progress.cfg`, which is created and modified during product installation.

[Figure 2](#) shows a typical display of the product configuration.

```
Configuration File: ../progress.cfg
    Company Name: Progress Software Corporation
    Product Name: OE Workgroup RDBMS
    Installation Date: Wed Oct  4 09:17:38 2006
    Expiration Date: None
    Serial Number:  -----
    Control Numbers:  -----
    Version Number:  10.1B
    Machine Class:   EB
    Port Number:    11
```

Figure 2: Product configuration details display using SHOWCFG utility

Refer to [Table 29](#) for an explanation of each of the display fields that appear in [Figure 2](#).

Displaying license information in Windows

You can display product license information such as the number of users and the platforms for each OpenEdge product installed on your system.



To display your current product license information in Windows platforms:

1. From the desktop, choose **Start**→**Run**. The **Run** dialog box appears.
2. Perform one of the following:
 - a. Type **showcfg** in the **Open** field, and choose **OK**.
 - b. At the command line, type the following command and press **ENTER**:

```
showcfg OpenEdge-install-dir\progress.cfg
```

The **OpenEdge Configuration Information** dialog box appears and displays the information you entered.

See the [“Managing user licenses on all supported platforms”](#) section on page 190 for more information about licensing.

Managing user licenses on all supported platforms

The OpenEdge license you purchase determines how many units are allowed to run your OpenEdge products. You are responsible for making sure users comply with your license agreement. OpenEdge provides reporting capabilities to help you ensure compliance with your license agreement.

The following sections describe:

- The OpenEdge license information that is shipped with your OpenEdge product
- How to read the OpenEdge usage log file
- How to produce a report of current licensed user connections

Removing license for installed product

Users can remove licenses for installed products. Removing individual licenses allows you to distribute parts of your product installation to another system without uninstalling the entire product.

Removing license from Windows platform



To remove a licence:

1. Choose **Start**→ **OpenEdge**→ **License Update**. The install program (`setup.exe`) starts and the **Welcome** dialog box appears.
2. Click **Next**. The **Serial Number and Control Codes** dialog box appears.
3. Select the license to remove from the **Product(s) List**, and click **Remove** to remove the license.

Removing license from UNIX platform



To remove a license:

1. Enter the following command in the command prompt:

```
./path/proupdt
```

Note: `path` refers to the directory (`d1c`) in which you have installed the software image. You cannot run `proupdt` if you are in the directory where you have saved the software image or the intended installation directory.

2. Press **Enter**. The **Welcome** dialog box appears.
3. Press **Enter**. The **Product configuration data** dialog box appears.
4. In the **Product configuration data** dialog box, press **CTRL+R** to display the **Installed Product List** dialog box. In the **Installed Product List**, Press the **UP** and **Down** arrow to traverse through the installed products.
5. In the **Installed Product List** dialog box, press **Enter** to select the product (s) for which you want to remove the license (s). A star icon appears against the selected product.
6. Traverse to the end of the **Installed Product List** and select **Remove Selected License (s)**. The **License Removal Confirmation** dialog box appears.
7. In the **License Removal Confirmation** dialog box, type **Y** to confirm the removal of the license (s) or **N** to abort the process.

OpenEdge license information

The *License Addendum* that accompanies your OpenEdge media package (ESD download) with your product provides specific information about the product license you purchased, including:

- A serial number
- A control number
- The maximum number of units allowed by the license

When you install OpenEdge, the installation procedure prompts you to enter product information from the *License Addendum*. The installation procedure records the license information in the OpenEdge configuration file `progress.cfg`. Use the `SHOWCFG` utility to display the product license information for each OpenEdge product installed on your system.

Note: For information on installing OpenEdge components using the License Addendum File, refer to the [“Using an Electronic License Addendum file”](#) section on page 116 for installing in Windows, and the [“Obtaining an Electronic License Addendum file”](#) section on page 72 for installing on UNIX.

For more information on the `SHOWCFG` utility, see the [“Using the SHOWCFG utility in Windows”](#) section on page 186, or the [“Using the SHOWCFG utility on UNIX or Linux platforms”](#) section on page 188.

Using the OpenEdge license file

OpenEdge creates a license file that records license-related information about OpenEdge database users. If the log file does not already exist, the broker creates it and places it in the same directory as the database (`.db`) file. The broker creates the file in the format `databasename.lic`, where `databasename` is the name of the database to which the user connects.

Note: If OpenEdge encounters an error while trying to open or write to the license file, the error is recorded in the database `.lg` file and no more entries are written to the license (`.lic`) file.

Reading the license file

Use a text editor to display the license file contents. The contents appear in the following order:

1. Current date
2. Current time
3. Number of licensed users specified by the configuration file
4. Current number of total connections
5. Maximum number of total connections

6. Minimum number of total connections
7. Current number of interactive connections
8. Maximum number of interactive connections for the past hour
9. Minimum number of interactive connections for the past hour
10. Current number of batch connections
11. Maximum number of batch connections for the past hour
12. Minimum number of batch connections for the past hour

For example, the following sample file entry illustrates the log format:

```
4/26/08 9:00 25 18 23 11 17 20 11 1 5 0
```

When OpenEdge writes to the license file, the maximum and minimum values are reset for the next hour.

Maintaining the license file

The database or system administrator should consider archiving license files periodically. In one year, a license file accumulates 8,760 entries. These entries occupy about 440,000 bytes of disk space.

Since the license file must be closed before the administrator archives it, the administrator must first shut down the database. At that point, the license file can be either archived immediately or renamed and archived later.

Creating a usage report

To produce a report of license-related information about current OpenEdge database users, run the `licrpt.p` procedure file. The report generator input data appears:

```
Enter Date Range:      To:      Enter Start Time (hours 0 to 23):
Enter Stop Time (hours 0 to 24):
Enter time division (in hours, or 0 for complete range):
Database Name:
```

This is a sample output from the `licrpt.p` procedure file:

Database Connection Counts									
Date	Period	LcnUsers	MaxTot	Excptns	MinTot	AveTot	MaxBat	MinBat	AvBat
4/11/06	8-17	100	20	0	0	10.	0	0	0.
4/13/06	8-17	100	20	0	18	19.	0	0	0.
4/16/06	8-17	100	23	0	17	20.	0	0	0.
4/20/06	8-17	100	33	0	17	25.	0	0	0.
4/24/06	8-17	100	32	0	26	29.	0	0	0.
4/26/06	8-17	100	26	0	17	22.	0	0	0.

Using OpenEdge resources in Windows

OpenEdge uses several operating system resources such as shared memory and memory locks, processes, and client memory in Windows. You can plan OpenEdge operations more effectively if you understand these resources.

Shared memory

Shared memory is an area in the system memory that multiple users can access concurrently. OpenEdge stores shared resources in the shared-memory area, enabling multiple users and servers access to each database. OpenEdge uses semaphores and spin locks to synchronize the activities of server and self-service client processes that are connected to a database. Each process uses its semaphore or relies upon the spin lock when it must wait for a shared resource.

You can tune OpenEdge performance by reconfiguring the size of the following shared-memory buffers:

- **Database buffers** — OpenEdge reads database blocks into the database buffer pool. Larger buffers usually result in less disk I/O.
- **Before-image (BI) buffers** — OpenEdge stores BI notes in memory before writing them to disk.
- **After-image (AI) buffers** — OpenEdge stores AI notes in memory before writing them to disk.

OpenEdge also creates shared-memory tables to provide essential information on the status of each process, server, transaction, and lock. These tables enable you to control all of the database activities from one shared area.

Processes on Windows and UNIX platforms

OpenEdge provides the following optional processes to improve performance in Windows and on UNIX platforms:

- **Asynchronous Page Writer (APW)** — Improves database 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 checkpointing (the process of synchronizing the buffer pool of modified blocks to the database).
- **Before-image Writer (BIW)** — Improves performance by continually writing before-image buffers to disk. These writes occur in the background.
- **After-image Writer (AIW)** — Improves performance by continually writing AI buffers to disk soon after OpenEdge fills the buffers.
- **OpenEdge Watchdog (PROWDOG)** — Cleans up after improperly terminated processes by releasing locks, backing out any live transactions and releasing shared-memory locks, and disconnecting and cleaning up the server's remote clients.

Manage memory and system configurations on UNIX platforms

The following sections describe how to manage your system's memory and configuration on UNIX platforms:

- [Calculating memory needs](#)
- [Managing shared memory and process resources](#)
- [Reducing memory usage](#)
- [Swap space](#)
- [Shared memory and kernel configuration](#)

Calculating memory needs

The tables in this section are provided to help you calculate the memory requirements for your system. Keep in mind that all memory usage figures are approximate and vary depending on the version of the operating system, UNIX parameters, the OpenEdge startup parameters, and the OpenEdge application you are using. For more information, see [OpenEdge Deployment: Startup Command and Parameter Reference](#).

Note: The background processes `APW`, `BIW`, `AIW`, and `PROWDOG` also take up memory. Remember to calculate these in your memory requirements.

[Table 30](#) lists the components you use to calculate system memory requirements.

Table 30: Components used to calculate memory needs (1 of 2)

Component	Symbol	Comment
Operating system	<code>os*</code>	Represents the memory requirements for one copy of your operating system shared in memory by all users, plus a certain percentage of physical memory to allow for operating system buffers; typically, 10%–15%.
OpenEdge	<code>_progres*</code>	Represents the size of one copy of OpenEdge shared in memory by all users running single-user or multi-user OpenEdge—allow for 15%–20% deviation in the <code>_progres</code> value to accommodate new releases.
Database server or broker	<code>_mprosrv*</code>	Represents the size of one copy of the OpenEdge database broker/server shared in memory by all users running multi-user OpenEdge. Use this component only when calculating memory requirements for a system running a multi-user version of an OpenEdge product.

Table 30: Components used to calculate memory needs (2 of 2)

Component	Symbol	Comment
OpenEdge user data	<code>proud</code>	<p>Represents the data area required for each user running OpenEdge.^{1 2} This value varies greatly, depending on the application you run and whether you use the compiler. It is also affected by many of the startup parameters. For single-user clients, the parameters are:</p> <ul style="list-style-type: none"> • Blocks in Database Buffers (<code>-B</code>) • Directory Size (<code>-D</code>) • Stack Size (<code>-s</code>). <p>For multi-user clients, the parameters are:</p> <ul style="list-style-type: none"> • Directory Size (<code>-D</code>) • Stack Size (<code>-s</code>) • Maximum Memory (<code>-mmax</code>)
OpenEdge server data	<code>psd</code>	Represents the data area required for each database server serving remote clients. (Not used for single-user or multi-user clients if the users are self-service). This space is used for communication buffers and other server memory requirements.
OpenEdge broker data	<code>pbd</code>	<p>Represents the data area required by each database broker. (One database broker is required for each different database simultaneously in use in multi-user mode whether you are using remote client/servers, self-service, or both.) This value is determined by the values of startup parameters² that consume memory, including:</p> <ul style="list-style-type: none"> • Database Buffers (<code>-B</code>) • Lock-table Entries (<code>-L</code>) • Number of Users (<code>-n</code>). <p>Note: Each increment of <code>-n</code> increases <code>pbd</code> by 2K.</p>

1. Use the UNIX `size` command to determine the exact size. See [Table 31](#) to determine the approximate value.

2. See [OpenEdge Deployment: Startup Command and Parameter Reference](#) for information about OpenEdge startup parameters.

Table 31 lists the startup options that affect memory requirements.

Table 31: Size increments for increasing startup parameters by 1

Startup	Size increment	Affects
Blocks in database buffers (-B)	db block size (.5K, 1K, 2K, 4K, 8K)	multi-user: pbd; single-user: proud
Directory size (-D)	100 bytes	proud
Lock-table entries (-L)	16 bytes	pbd
Shared-memory size (-Mxs)	1K	pbd
Number of users (-n)	2K	pbd
Stack size (-s)	1K	proud

Table 32 and Table 33 list approximate values for each calculation component for single and multiple users running an OpenEdge installation.

Table 32: Single-user memory requirements

Component symbol	Memory
_progres	3MB–4MB ¹

1. This is an approximate value. Use the size command to determine the exact size. If you are using a non-OpenEdge database, your value will be larger.

Table 33: Multi-user memory requirements

Component symbol	Memory
_progres	3MB–4MB ¹
_mprosrv	1MB–2MB ¹

1. This is an approximate value. Use the size command to determine the exact size. if you are using a non-Open-Edge database, your value will be larger.

Table 34 provides the formulas to calculate the memory requirements for your system without disk swapping.

Table 34: Formulas for calculating memory requirements

Single-user systems	Multi-user systems
$\text{os} + \text{_progres} + (\text{number of users} \times \text{proud})$	$\begin{aligned} &\text{os} + \text{_progres} + \text{_mprosrv} \\ &+ (\text{number of databases} \times \text{pbd}) \\ &+ (\text{number of remote client servers} \times \text{psd}) \\ &+ (\text{number of users} \times \text{proud}) \end{aligned}$

Note: Remote client/server processes share the same code as the broker and, therefore, require no additional `_mprosrv` (database server or broker) memory. Each remote client/server process does require an OpenEdge server data (`psd`) area.

Managing shared memory and process resources

OpenEdge uses several operating system resources, such as shared memory and memory locks, processes, and client memory. You can plan OpenEdge operations more effectively if you understand these resources.

Shared memory

Shared memory is an area in system memory that multiple users can access concurrently. OpenEdge keeps resources shared by all database users in shared memory and lets multiple servers access those resources efficiently. OpenEdge uses semaphores and spin locks to synchronize the activities of server and self-service client processes that are connected to a database. Each process uses its semaphore or relies upon the spin lock when it must wait for a shared resource.

You can tune OpenEdge performance by reconfiguring the size of the following shared memory buffers:

- **Database buffers** — OpenEdge reads database blocks into the database buffer pool. Larger buffers usually result in less disk I/O.
- **Before-image (BI) buffers** — OpenEdge stores BI notes in memory before writing them to disk.
- **After-image (AI) buffers** — OpenEdge stores AI notes in memory before writing them to disk.

OpenEdge also creates shared memory tables to provide essential information on the status of each process, server, transaction, and lock. These tables enable you to control all of the database activities from one shared area.

See [OpenEdge Data Management: Database Administration](#) for more information about improving performance.

Processes on UNIX platforms

OpenEdge supports the same optional processes in Windows as it does on UNIX or Linux platforms. For a list of these optional processes and a brief description of each, see the [“Processes on Windows and UNIX platforms”](#) section on page 195.

Reducing memory usage

If you run OpenEdge and find there is not enough main memory, try the following to reduce main memory use:

- Reduce the amount of memory allocated to OpenEdge database buffers, as controlled by the `-B` startup parameter
- Change other startup parameters, such as `-n` and `-L`

For more information about startup parameters, see [OpenEdge Deployment: Startup Command and Parameter Reference](#).

Swap space

When the amount of memory used by all processes running on a UNIX system exceeds the amount of physical memory, portions of memory are swapped to disk. A special area of the disk is reserved for this swapping. The system administrator can set the size of this area when configuring the system.

Note: Progress Software Corporation recommends that you set your swap space size to at least twice the size of your system memory.

A UNIX system can deadlock while accessing the disk when the swap space is used up. This can happen when too many large processes are running simultaneously. If you expect to have a larger than normal number of users, or if OpenEdge memory requirements are larger than your typical process, consider increasing the amount of swap space available on your system. Before you change the size of the swap area, back up and reformat the disk.

The UNIX user set-ID bit is turned on for the OpenEdge program module. Consequently, even though there might be no active OpenEdge users, this module remains in the UNIX swap area on the disk until you shut down the system.

Shared memory and kernel configuration

In OpenEdge, the multi-threaded architecture makes heavy use of file descriptors, shared memory, and semaphores. Allocation of these resources is controlled by system configuration parameters. On most systems, these parameters are set to values appropriate for OpenEdge applications. However, in some cases, one or more parameters might not be set optimally, thereby limiting the number of OpenEdge users. If you have to reset the parameters, you must reconfigure your kernel. See your operating system documentation for information on reconfiguring your operating system kernel.

The optimal parameter settings depend on the system, the application, the number of users, and some minor factors. [Table 35](#) lists the crucial parameters and provides guidelines for choosing adequate values for each one.

Table 35: Shared memory and semaphore parameter settings

Parameter	Meaning	Optimal setting
SHMMNI	Maximum number of shared memory (SHM) identifiers	Current value or system default + (total OpenEdge memory requirement)/SHMMAX
SHMSEG	Maximum number of SHM segments a single process can attach	4–8
SHMALL	Maximum number of in-use SHM segments	System default; increase if many databases are active simultaneously; decreasing -B, -n, and -L startup parameters decreases SHM requirements
SHMMAX	Maximum SHM segment size	System default; increase if you get OpenEdge error 1135 Note: On the AIX platform, when starting a database with large shared memory requirements (for instance, when the -B exceeds the allotted system paging space), the system may become unstable if the PSALLOC= early environment variable is not set.
SEMMNI	Number of semaphore (SEM) IDs; each represents an array of SEMs	1 per active multi-user database
SEMMSL	Maximum number of semaphores per SEM ID	(Max-local-users-on-any-database + Max-#servers-on-any-database + 4)
SEMMNS	Total semaphores in the system	(SEMMSL x #active-databases)
SEMMNU	Number of semaphore undo structures	Same value as SEMMNS
MAXUMEM	Maximum address space for a single user	> = server size process + SHMSEG * SHMMAX

The parameter settings in [Table 35](#) are guidelines. Parameter values near these are acceptable in most cases, but a particular system or application might require increasing the limits.

If shared memory or semaphores are allocated incorrectly, OpenEdge displays an error message when it attempts to start an additional user or server. For example, if `SEMMNS` is set too low, `PROSERVE` fails and displays the following message:

```
Server: Semaphore limit exceeded
Server: **The server terminated with exit code (X) (800)
```

Change the relevant parameter values and reconfigure the kernel in response to semaphore or shared-memory errors at startup. [Table 36](#) lists the parameters that you might have to raise in response to various OpenEdge error codes.

Table 36: Error codes and kernel reconfiguration parameters

Error code	Parameter to increase
1081	SEMMNU
1093	SEMMSL or SEMMNS
1130	SEMMSL
1131	SEMMNI and SEMMNS
1135	SHMMAX, MAXUMEM, and MAXUP On the AIX platform, when starting a database with large shared memory requirements (for instance, when the <code>-B</code> exceeds the allotted system paging space), the system may become unstable if the <code>PSALLOC=</code> early environment variable is not set.
1137	SHMMNI
1175	SHMSEG, MAXUMEM, and MAXUP
1195	SEMMNS

Note: The Blocks in Database Buffers (`-B`), Lock-table Entries (`-L`), and Number of Users (`-n`) startup parameters all affect shared-memory usage. The Number of Users (`-n`) and Maximum Servers (`-Mn`) parameters affect semaphore usage (each user or server process uses one semaphore). Before reconfiguring your kernel to increase shared memory or semaphore allocation, see whether you can lower these startup values.

UNIX troubleshooting tips

This section provides issues to consider when troubleshooting an installation as described in the following:

- [Error messages](#)
- [Altered or missing progress.cfg file](#)
- [Tailoring startup scripts](#)

Error messages

[Table 37](#) lists some typical error messages, probable causes, and where to find solutions.

Table 37: Error messages

Error message	Cause of error	Solution
Unable to read progress.cfg, reason -1 (1732)	The progress.cfg file is altered or missing.	See the “Altered or missing progress.cfg file” section on page 203.
Module-name not found	The environment variables are not set correctly or are not installed.	See the “Tailoring startup scripts” section on page 204.
Error 304 and 305	The <code>ULIMIT</code> is set too low.	Reset your ulimit.

Altered or missing progress.cfg file

If you receive the following error message, the `progress.cfg` file has been altered or deleted from the directory where you installed your OpenEdge products:

```
Unable to read progress.cfg, reason=-1.
```

If you receive this message, you must reinstall the OpenEdge product.

Caution: Do not alter or delete the `progress.cfg` file, as this will cause the OpenEdge broker startup to fail.

Table 38 lists the reasons for an altered or missing `progress.cfg` file.

Table 38: Reasons for altered or missing progress.cfg file

Reason	Description
-1	Could not find <code>OpenEdge-install-dir/progress.cfg</code>
-4	Bad checksum; invalid file
-6	Could not read the specified number of bytes; the file is truncated
-7	Could not allocate enough memory to read the configuration file

Tailoring startup scripts

Typically, the installation procedure automatically tailors the startup scripts for the OpenEdge products you install. Tailoring involves setting each script's environment variable to point to the directory where you installed the product referenced by the script. If the installation procedure is interrupted before the script tailoring is complete, or if the normal installation procedure is not used, you might have to tailor the scripts manually.

Depending on the products you purchase and install, your OpenEdge installation provides the required scripts. Some of the OpenEdge startup scripts are shown in the following table:

<code>adaptman</code>	<code>mpro</code>	<code>proadsv</code>	<code>prooidrv</code>
<code>aiaman</code>	<code>mssman</code>	<code>proaiw</code>	<code>proserve</code>
<code>asbman</code>	<code>nsman</code>	<code>proapw</code>	<code>proshut</code>
<code>bpro</code>	<code>odbman</code>	<code>probiw</code>	<code>prowdog</code>
<code>dbman</code>	<code>oraman</code>	<code>probrkr</code>	<code>wsaman</code>
<code>mbpro</code>	<code>pro</code>	<code>prooibrk</code>	<code>wtbman</code>

Note: The scripts listed are located in `OpenEdge-install-dir/bin`.

If the automatic tailoring does not take place, you receive the following error message when you try to start your OpenEdge product:

```
module-name not found
```

The `module-name` is the OpenEdge module that the script is trying to start. For example, if the script is `pro`, the module name is `_progres`.

**To tailor your startup scripts manually:**

1. Use any text editor to edit the scripts.
2. Look for the following syntax:

```
env_variable=${env_variable-pathname}; export env_variable
```

3. Change the pathname to the full pathname of the directory where you installed your OpenEdge product. For example:

```
DLC=${DLC-/usr/grp/dlc};export DLC
```

OpenEdge event logging

OpenEdge logs significant database events such as OpenEdge startup parameter settings, startup error messages, shutdown messages, system error messages, and application-related events, as described in the following sections:

- [OpenEdge event log file](#)
- [Managing the OpenEdge event log file size](#)
- [Event logging in Windows](#)

OpenEdge event log file

The OpenEdge event log is a text file that contains a history of significant database events, such as OpenEdge startup parameter settings and startup, shutdown, and system error messages. This file has a `.lg` extension.

Managing the OpenEdge event log file size

The event log (`LG`) file expands as you use the database. If it becomes too large, you can use the `PROLOG` utility, in either an offline or online mode, to reduce the event log file's size. Using the `PROLOG` utility, you can:

- **Truncate log entries offline** — Removes old log entries. To remove log entries from an `LG` file, use the OpenEdge Log Maintenance (`PROLOG`) utility or a text editor. The syntax to use the `PROLOG` utility in the offline mode is described in the [“Remove old log entries”](#) section on page 207.
- **Truncate log file entries online** — Removes entries in the database log file while the database is online. The online activity is intended to help you avoid bringing the database down and restarting it after the database has been truncated. Using this approach, the need to shutdown the database to archive the log file is eliminated. However, keep in mind that it is possible to lose some messages while performing this procedure due to the nature of the real-time processing. The syntax to use the `PROLOG` utility in the online mode is described in the [“Truncate the database log file”](#) section on page 207.

Caution: During the time in which the multi-stepped online truncation process occurs, some messages written to the log file might get lost because the database is neither quiet nor latched/locked to prevent writes.

Remove old log entries

This section describes topics related to removing and truncating log file entries.



To remove old log entries from an event log file, enter the following command:

```
prolog database-name
```

The `PROLOG` utility removes all but the most recent entries from the log file. For more information, see the details about `PROLOG` in the “[Truncate the database log file](#)” section on page 207 and also see [OpenEdge Data Management: Database Administration](#).

Truncate the database log file

The purpose of this activity is to allow you to truncate a database log file that exists and the size is greater than 3072 bytes.

Truncating a log file online

The online activity is intended to help you avoid bringing the database down and restarting it after the database has been truncated. This eliminates the need to shutdown the database to archive the log file. An online truncation log file records the start and end of truncation activities and records errors to indicate when a truncation failed, such as:

```
prolog database-name [-online]
```

`prolog`

Enables truncation of a log file. By default, a log file is truncated offline.

database-name

The name of the database to be truncated.

`-online`

Using the option `-online`, you do not have to shutdown and restart the database to truncate the database log file.

The online truncation option copies the last 3072+ bytes to a buffer, truncates the file, and then copies the buffer to the file.

Note: Keep in mind that if the `-online` option is used, the `prolog` command can truncate a log file even if the database is in use.

For more information about the syntax associated with these online and offline activities, see [OpenEdge Data Management: Database Administration](#).

Event logging in Windows

In addition to the OpenEdge event log, the OpenEdge Server writes events to the Event Log. The Event Log is the object that enables Windows users to view the status of applications, security, and system processes, and to view their associated events. OpenEdge is an application process and, as such, it writes Progress events to the Application Event Log. You use the Event Viewer to see the Event Log's contents. You can customize the Event Viewer so that it displays only the event types that you want to view. You access the Event Viewer through the Administrative Tools program group.

[Table 39](#) describes the components that enable the OpenEdge service to log messages to the Application Event Log database.

Table 39: Progress event logging components

Component	Function
Event viewer	The standard front-end that enables users to view the Event Log.
Event log	The standard Windows database that records event information.
CATEGORY.DLL	The OpenEdge resource that contains the 14 categories into which OpenEdge messages might fall.
PROMSGS file	The OpenEdge object that contains a single language version of the OpenEdge messages. OpenEdge supplies a PROMSGS file for each supported language version of Progress. The PROMSGS file is installed to the <i>OpenEdge-install-dir</i> location. See Appendix D, "OpenEdge National Language Support," and OpenEdge Development: Internationalizing Applications for more information on the PROMSGS file.

Managing OpenEdge events in Windows

You can define the level of event logging that you want your OpenEdge application to support by using either the Event Level Environment Variable (EVTLEVEL) or the Event Level startup parameter (-evtlevel).

Table 40 describes the valid Event Level values.

Table 40: Event Level values

Value	Description
None	No OpenEdge events are written to the event log.
Brief	OpenEdge messages defined as Error and Warning messages are written to the event log.
Normal	OpenEdge messages defined as Error and Warning messages are written to the event log. In addition, any Progress message that is normally written to the log file (.lg) is also written to the Event Log. This is the default.
Full	Every message generated by OpenEdge is written to the Event Log. Any OpenEdge messages generated using the Message Statements are also written to the log file.

Understanding the Windows Application Event Log components

The components of the Windows Application Event Log are standards defined by Windows. Figure 3 illustrates the Windows Application Event Log components when shown through the Event Viewer.

Type	Date	Time	Source	Category	Event	User	Comp...
Error	10/3/2006	5:23:44 PM	Userenv	None	1085	SYSTEM	NBBH...
Warning	10/3/2006	5:23:44 PM	SecCli	None	1202	N/A	NBBH...
Information	10/3/2006	3:33:29 PM	SCHEMA1_PROGRESS	DATASE...	2690	N/A	NBBH...
Information	10/3/2006	3:33:29 PM	SCHEMA1_PROGRESS	DATASE...	6490	N/A	NBBH...
Information	10/3/2006	3:33:29 PM	SCHEMA1_PROGRESS	DATASE...	6489	N/A	NBBH...
Information	10/3/2006	3:33:29 PM	SCHEMA1_PROGRESS	DATASE...	0	N/A	NBBH...
Information	10/3/2006	3:33:29 PM	SCHEMA1_PROGRESS	DATASE...	2689	N/A	NBBH...
Information	10/3/2006	3:33:29 PM	SCHEMA1_PROGRESS	DATASE...	5067	N/A	NBBH...
Information	10/3/2006	3:33:22 PM	ORACLE_TEST_PRO...	PROGRE...	12699	N/A	NBBH...
Information	10/3/2006	3:33:22 PM	ORACLE_TEST_PRO...	PROGRE...	13547	N/A	NBBH...
Information	10/3/2006	3:33:22 PM	ORACLE_TEST_PRO...	PROGRE...	7161	N/A	NBBH...
Information	10/3/2006	3:33:22 PM	ORACLE_TEST_PRO...	PROGRE...	5326	N/A	NBBH...
Information	10/3/2006	3:33:22 PM	ORACLE_TEST_PRO...	PROGRE...	451	N/A	NBBH...
Error	10/3/2006	3:24:39 PM	Userenv	None	1085	SYSTEM	NBBH...
Warning	10/3/2006	3:24:39 PM	SecCli	None	1202	N/A	NBBH...
Information	10/3/2006	3:14:57 PM	SPORTS2_PROGRESS	PROGRE...	12699	N/A	NBBH...
Information	10/3/2006	3:14:57 PM	SPORTS2_PROGRESS	PROGRE...	13547	N/A	NBBH...
Information	10/3/2006	3:14:57 PM	SPORTS2_PROGRESS	PROGRE...	7161	N/A	NBBH...
Information	10/3/2006	3:14:57 PM	SPORTS2_PROGRESS	PROGRE...	5326	N/A	NBBH...
Information	10/3/2006	3:14:57 PM	SPORTS2_PROGRESS	PROGRE...	451	N/A	NBBH...
Error	10/3/2006	1:49:34 PM	Userenv	None	1085	SYSTEM	NBBH...
Warning	10/3/2006	1:49:34 PM	SecCli	None	1202	N/A	NBBH...
Information	10/3/2006	1:01:56 PM	SPORTS2_PROGRESS	PROGRE...	334	N/A	NBBH...
Information	10/3/2006	1:01:56 PM	SPORTS2_PROGRESS	PROGRE...	334	N/A	NBBH...
Information	10/3/2006	1:01:56 PM	TESTORACLE1_PRO...	PROGRE...	334	N/A	NBBH...
Information	10/3/2006	1:01:56 PM	TESTSCHEMA_PROG...	PROGRE...	334	N/A	NBBH...
Information	10/3/2006	12:57:50 PM	ORACLE_TEST_PRO...	PROGRE...	334	N/A	NBBH...
Information	10/3/2006	12:15:18 PM	PROGRESS	DATASE...	2690	N/A	NBBH...
Information	10/3/2006	12:15:18 PM	PROGRESS	DATASE...	6490	N/A	NBBH...
Information	10/3/2006	12:15:18 PM	PROGRESS	DATASE...	6489	N/A	NBBH...
Information	10/3/2006	12:15:18 PM	PROGRESS	DATASE...	0	N/A	NBBH...
Information	10/3/2006	12:15:18 PM	PROGRESS	DATASE...	2689	N/A	NBBH...
Information	10/3/2006	12:15:18 PM	PROGRESS	DATASE...	5067	N/A	NBBH...

Figure 3: Windows Application Event Log components

Table 41 describes how Progress uses the Windows Application Event Log components.

Table 41: Windows Application Event Log components

Log component	Log information
Type	Identifies the type of message such as Information , Warning , or Error .
Date	Identifies the date the event occurred.
Time	Identifies the time the event occurred.
Source	<p>Source of the event. This is the name of the connected Progress database, if a database is connected. If no database is connected, then "Progress" is listed.</p> <p>If you are using the Progress AppServer, "Progress" is also the default source for Progress AppServer messages. However, you can override the default source name by specifying the <code>-logname</code> AppServer broker startup parameter.</p>
Category	<p>Provides information to help you isolate the cause of the message displayed in the Event Log. Progress supports 14 event categories. The event categories are: AIW, APW, BACKUP, BIW, DATASERVER, MON, OIBRKR, OIDRVR, Progress, RFUTIL, SERVER, SHUT, USER, and PROWDG. When no database is connected, Progress is specified as the category.</p> <p>All categories reside in a file called <code>category.dll</code>. These categories correspond to the existing categories of events that are displayed in the <code>progress.lg</code> file (AppServer broker and application server events are displayed in the AppServer log file, <code>proapsv.lg</code>).</p> <p>(Note that DATASERVER is not included as a category in the standard <code>progress.lg</code> file.)</p>
Event	Associates to the Progress message that was generated. These are the same message numbers that are displayed in the standard database <code>.lg</code> file.
User	Identifies the user logged in to the Windows workstation where the event occurred.
Computer	Identifies the name of the Windows workstation where the event occurred. The Event Viewer allows you to get more information about events by double-clicking on any event.

You can view additional information about an event by double-clicking on it. Windows displays the **Event Properties** dialog box, as shown in [Figure 4](#).

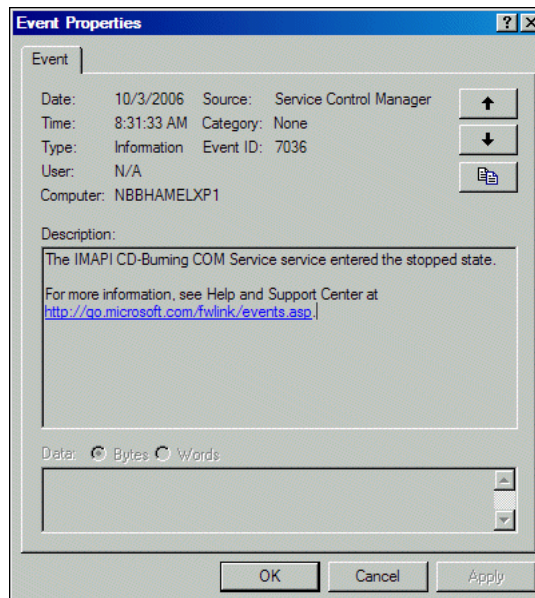


Figure 4: Windows application Event Properties dialog box

The **Event** tab displays details about the event you initially select. You can also use the arrow controls on the **Event** tab to scroll through detailed information about the other events that appear on the Windows Application Event Log components viewer as shown in [Figure 3](#).

Windows Event Log and registry

Windows requires applications that use the Event Log be bound to all of the necessary components. For Progress this means that the `PROMSGS.DLL` and the `CATEGORY.DLL` must be bound to any Progress database. Progress stores this information in the registry. Progress makes the registry entries and performs any binding operations that are necessary when you initially access a database. When Progress binds the DLL files to the database, it writes the fully qualified pathname to the registry. If you delete the database, you must manually remove the associated data from the registry. If you move the location of the DLLs after you access the database, you must manually edit the registry data.

The Progress components can be found in the following location in the registry:

```
HKEY_LOCAL_MACHINE
SYSTEM
  CurrentControlSet
    Services
      EventLog
        Security
        System
        Application
          PROGRESS
            <Database Name>
```

See the Microsoft documentation for more information about editing registry files.

When OpenEdge tries to find the DLLs before this information is included in the registry, it performs the search according to the sequence of the following rules:

1. OpenEdge searches the current directory.
2. If the DLL is not in the current directory, OpenEdge searches the directory where the Progress executable is located.
3. If the DLL is not in the same directory as the OpenEdge executable, OpenEdge searches the user's path.

If the DLL is not in the user's path, OpenEdge generates a message stating that the DLL cannot be found, and it writes a message to the OpenEdge log file.

Part 2

Configuration

[Chapter 7, Working in the OpenEdge Environment in Windows](#)

[Chapter 8, Working in the OpenEdge Environment on UNIX](#)

[Chapter 9, Managing OpenEdge Key and Certificate Stores](#)

[Chapter 10, Configuration](#)

[Chapter 11, Starting and Running OpenEdge](#)

Working in the OpenEdge Environment in Windows

This chapter describes how the OpenEdge environment works in Windows. It also provides steps to maintain OpenEdge versions on your system, as described in the following sections:

- [Reviewing environment variables](#)
- [Windows registry and the progress.ini file](#)
- [Setting OpenEdge Program Item properties](#)
- [Using the Proenv utility](#)
- [Getting started with the AdminServer](#)
- [OpenEdge products supported by the AdminServer](#)
- [Creating and configuring an OpenEdge database server](#)
- [Running OpenEdge](#)
- [Maintaining OpenEdge and Progress](#)
- [OpenEdge key and certificate stores](#)
- [Support for IPv6](#)
- [Windows 64-bit](#)

Reviewing environment variables

By default, the OpenEdge installation program tailors all the necessary OpenEdge and Java environment variables to the directories where they are installed. For example, the installation automatically sets the `%DLC%` environment variable to your OpenEdge installation path.

This section briefly reviews some system and Java environment variable details of which you should be aware. [Table 42](#) is a listing of supported environment variables.

System environment variables

The `%DLC%` environment variable is not set at the system level and should not be changed. After installing OpenEdge, however, you can set environment variables to suit your own preferences. You can use Proenv to set the `%DLC%` environment variable to the directory where OpenEdge is installed.

Caution: Although editing environment variables is an option, this procedure is not recommended if more than one version of an OpenEdge product exists on the same system.

For more information on environment variables, see the information on maintaining user environments in [OpenEdge Deployment: Managing ABL Applications](#), or see your specific product documentation.

Latest information updates

Before you continue, consult *OpenEdge Release Notes*. These notes contain the latest information about the current release that the OpenEdge documentation set might not include. Progress Software Corporation ships release notes in Microsoft Write (`readme.wri`) format. Click the **Release Notes** icon in your OpenEdge program group, or access `Readme.pro` with any text editor.

Java environment variables

OpenEdge bundles the Java Runtime Environment (JRE) component and the Java Development Kit (JDK) component with certain products that you install. For more information, see the “[Java considerations](#)” section on page 32.

Note: OpenEdge supports Java version 7.0. For specific information about these components, see the *OpenEdge 10 Platform & Product Availability Guide* on the Progress Software Corporation Web site
<http://www.progress.com/products/lifecycle/index.ssp>.

JDKHOME

Java is used by some products, such as WebSpeed, the AppServer, and SQL, for product functionality. After you install any of these products, you should verify that the `JDKHOME` value is set correctly in the registry. The value must be set to the directory where the JDK included in the OpenEdge installation resides (for example, `C:\Progress\OpenEdge\jdk`).

You can verify the `JDKHOME` value in the following location in the registry:

`HKEY_LOCAL_MACHINE\SOFTWARE\PSC\PROGRESS\version11.3\JAVA`

Windows registry and the progress.ini file

Applications running in Windows rely on the registry for startup information, such as color, font, and key bindings. Variables described in this section are reserved for use by the OpenEdge installation. The installation variables have already been defined.

Caution: You should proceed with extreme caution if you are considering a change to any of the variables listed in the following sections.

Information from the `progress.ini` resides under the following registry keys:

`HKEY_CURRENT_USER\SOFTWARE\PSC\PROGRESS\version11.3`

`HEKY_LOCAL_MACHINE\SOFTWARE\PSC\PROGRESS\version11.3`

Note: If you modify the `progress.ini` information, you must run the `ini2reg` utility. This utility updates the information in the registry.

See [OpenEdge Deployment: Managing ABL Applications](#) for more information on the `progress.ini` file and the `ini2reg` utility.

Environment variables

OpenEdge supports some environment variables for graphical user interface (GUI) clients in the [Startup] section of the `progress.ini` file. OpenEdge supports environment variables for character clients, such as the AppServer and WebSpeed Agents, in the [WinChar Startup] section of the `progress.ini` file.

Table 42 lists the supported environment variables.

Table 42: Windows environment variables

(1 of 4)

Variable	Progress.ini file section	Description	Example
DLC ¹	[Startup] [WinChar Startup]	The absolute pathname of the directory where you installed your OpenEdge system software. By default, the installation utility sets this variable.	set DLC=C:\Progress\ OpenEdge
EVTLEVEL	—	Specifies the level of information that OpenEdge writes to the Application Event Log. You can specify one of the following cases: <ul style="list-style-type: none"> • None — No OpenEdge events are written to the Event Log. • Brief — OpenEdge Error and Warning messages are written to the Event Log. • Normal — OpenEdge Error and Warning messages are written to the Event Log along with any OpenEdge messages that are normally written to the log file (.lg). This is the default. • Full — OpenEdge Error, Warning, and Informational messages are written to the Event Log along with any messages generated by the Message Statement. 	set EVTLEVEL = NORMAL

Table 42: Windows environment variables*(2 of 4)*

Variable	Progress.ini file section	Description	Example
PATH	—	<p>A list of directory paths separated by semicolons. When you run a program or batch file, the system searches for it in the current directory. Then it searches in the directory paths defined in <code>PATH</code> in the order they are mentioned.</p> <p>Your <code>PATH</code> should include any directory pathname that contains a program or batch file you want to run. Also, each directory pathname should include the drive letter of the disk that contains the directory. <code>PATH</code> is a system environment variable, not an OpenEdge environment variable. Set it in a manner appropriate for the operating system instead of in the registry or in <code>progress.ini</code>.</p> <p>Different OpenEdge products require different <code>PATH</code> settings. To set up <code>PATH</code> for your OpenEdge product, follow the instructions provided in the “Reviewing the Windows installation directory structure” section on page 79.</p>	<pre>set PATH=%PATH%; %DLC%\BIN;%DLC%</pre>
OEBUILD	[Startup]	The pathname of the directory that contains items referenced in link scripts produced by the <code>OEBUILD</code> utility. By default, the installation utility sets this variable.	<pre>set OEBUILD= C:\Progress\ OpenEdge\OEBUI LD</pre>
PROCFG	[Startup] [WinChar Startup]	The filename (or full pathname) of your product's configuration file. The configuration file is a data file that identifies the OpenEdge products and components that you are licensed to use. Reset <code>PROCFG</code> if you have moved your configuration file from the directory where you installed OpenEdge.	<pre>set PROCFG=%DLC%\ PROGRESS.CFG</pre>

Table 42: Windows environment variables (3 of 4)

Variable	Progress.ini file section	Description	Example
PROCONV	—	The filename (or full pathname) of the OpenEdge <code>convmap.cp</code> file. The <code>convmap.cp</code> file is a binary file that contains all of the conversion tables that are available to OpenEdge. See OpenEdge Development: Internationalizing Applications for more information on the <code>convmap.cp</code> file.	set PROCONV=%DLC%\CONVMAP.CP
PROMSGS	[Startup] [WinChar Setup]	The full pathname of your OpenEdge error messages file. The default value is <code>%DLC%\promsgs</code> . Set the <code>PROMSGS</code> environment variable only if you want to use an error messages file different from the default <code>PROMSGS</code> file in the <code>%DLC%</code> directory.	set PROMSGS=C:\Progress\OpenEdge\PROLANG\GER\PROMSGS.GER
PROPATH	[Startup] [WinChar Setup]	A list of directory paths separated by commas. By default, the installation utility sets this variable.	set PROPATH=., C:\Progress\OpenEdge
PROSTARTUP	—	The pathname of the OpenEdge default startup parameter file, <code>startup.pf</code> . This file is read by all OpenEdge modules at startup; it must exist for OpenEdge to execute properly.	set PROSTARTUP=C:%DLC%\STARTUP.PF
JDKHOME	—	Establishes the top-level directory for the Java Developer's Kit (JDK).	set JDKHOME=%DLC%\jdk
JREHOME	—	Establishes the top-level directory for the Java Runtime Environment (JRE).	set JREHOME=%DLC%\jre
JFCHOME	—	Establishes the top-level directory for the Java Foundation Classes (JFC).	set JFCHOME=%DLC%\jfc
JDKCP	—	Sets the classpath for <code>class.zip</code> ; Java Developer's Kit (JDK) only.	set JDKCP=%variable-name%/lib/class.zip
JRECP	—	Sets the classpath for Java Runtime Environment (JRE); if no JRE, then it sets classpath for JDK.	set JRECP=%variable-name%/lib/rt.jar

Table 42: Windows environment variables*(4 of 4)*

Variable	Progress.ini file section	Description	Example
PROGRESSCP	—	Contains a list of paths, jar files, and zip files for running Progress-specific products.	<code>set PROGRESSCP=%variable-name% %/java/progress.zip</code>
CLASSPATH	—	OpenEdge correctly sets the appropriate classpath variable based on the platform in use.	<code>set CLASSPATH=\$JDKCP;\$PROGRESSCP</code>
JIT	—	Sets the just-in-time compiler correctly.	<code>set JIT="-nojit"</code>
JVMEXE	—	Sets the Java virtual machine to run correctly.	<code>set JVMEXE=jre</code>

1. The DLC variable is set in the various command scripts and in the registry; the variable is not set at the system level.

Additional details for Java-related environment variables

The `JavaTools.properties` file is a common text file that contains configuration information for all ABL clients. The `JavaTools.properties` file is located in the `OpenEdge-instal-dir/properties/` directory. The configuration information and settings defined in the `JavaTools.properties` file provide more information than the Java-related environment variables.

The `AdminServer plugins.properties` file, a common text file that contains configuration details for all OpenEdge databases, is another valuable resource with which you should be familiar. It contains information for plug-ins that can be loaded and managed by the AdminServer. The `AdminServer plugins.properties` file is also located in the `$DLC/properties/` directory. For more information about these files, see [Chapter 10, “Configuration.”](#)

Caution: Do not make user-modifications to the `JavaTools.properties` files as these properties support OpenEdge and Progress products only. Contact Progress Technical Support if you want to modify these properties.

Setting OpenEdge Program Item properties

Although the OpenEdge Installation Utility creates an OpenEdge Group with Program Items, you must set item properties such as startup parameters and buffer pools.

For example, to change the properties for an OpenEdge Program Item in Windows, highlight the item and choose **File**→**Properties** in Windows Explorer. See the appropriate Windows documentation for more information about setting Program Item properties.

For information on OpenEdge startup parameters, buffer pools, and related topics, see [OpenEdge Deployment: Startup Command and Parameter Reference](#), the OpenEdge DataServer Guides ([OpenEdge Data Management: DataServer for Microsoft SQL Server](#), [OpenEdge Data Management: DataServer for ODBC](#), and [OpenEdge Data Management: DataServer for Oracle](#)) and [OpenEdge Deployment: Managing ABL Applications](#).

Using the Proenv utility

The `%DLC%` environment variable is not set at the system level. The `Proenv` utility can automatically set the `%DLC%` environment variable to the directory where OpenEdge is installed. It then adds `%DLC%/bin` to your `PATH`.

To start the `Proenv` utility from the desktop, choose **Start**→**Programs**→**OpenEdge** (or the actual directory where you installed **OpenEdge**→**Proenv**). You can also start it from the command line. `Proenv` opens a DOS window, sets the environment variables, and then changes the current directory to the working directory you set when you installed OpenEdge.

Getting started with the AdminServer

The AdminServer is the central controlling element of the Unified Broker framework. It facilitates the tasks associated with managing and configuring your installation.

For details about the AdminServer, its role in the Unified Broker framework, and the tasks to use the AdminServer, see [Chapter 10, “Configuration.”](#)

OpenEdge products supported by the AdminServer

An AdminServer is installed on every system that contains the following OpenEdge products:

- OpenEdge databases: OpenEdge Personal RDBMS, OpenEdge Workgroup RDBMS, and OpenEdge Enterprise RDBMS
- DataServers: OpenEdge DataServer for ORACLE, OpenEdge DataSever for MS SQL Server, and DataServer for ODBC
- OpenEdge Application Server—Basic
- OpenEdge Application Server—Enterprise
- OpenEdge Studio
- Progress Developer Studio for OpenEdge
- Web Services Adapter
- AppServer Internet Adapter (AIA)
- OpenEdge Adapter for SonicMQ
- OpenEdge Adapter for Sonic ESB
- NameServer
- OpenEdge Development Server
- OpenEdge NameServer Load Balancer
- WebSpeed Workshop
- OpenEdge Replication
- OpenEdge Replication Plus

In Windows, the AdminServer starts automatically and runs as a service. The AdminServer must be running to use any of the OpenEdge Management or OpenEdge Explorer configuration tools, or any of the following command-line managing or validating utilities. For additional information about these topics, see the [“Overview of OpenEdge Management or OpenEdge Explorer”](#) section on page 268.

AdminServer considerations

Note the following points relevant to AdminServer usage:

- Before you start a WebSpeed or AppServer application, you must start the AdminServer.
- The AdminServer User-Group Authorization feature requires that you have privileges set to allow you access and operational privileges for the AdminServer. See the Installation Utility online help for detailed procedures on how to set up this feature.

- The AdminServer must be running before you can use the OpenEdge Explorer to configure and manage your applications.
- The `DLC` directory name for a remotely-enabled AdminServer cannot contain spaces.

AdminServer group name conventions and restrictions

During or after the installation process, you can optionally establish the following AdminServer authorization options for OpenEdge products that support the AdminServer:

- **User Authorization** — To require each individual user to provide a valid user name and password before the AdminServer can be started.
- **Group Authorization** — To setup user-defined group names for which operational privileges, at a group level, are required. Group name definitions must conform to specific guidelines.

The procedures to establish AdminServer authorization options are located in the Windows online help system under the topic titles “Establishing AdminServer Authorization Options during the Installation” and “Selecting the Authorization Feature when Starting the AdminServer.”

Additional AdminServer-related details can be found in [Appendix G, “AdminServer Authorization and Authentication.”](#)

Creating and configuring an OpenEdge database server

When you create an OpenEdge database, you can either create a new database or convert an existing database:

- Use the PRODB command, the Data Administration Tool, or the Data Dictionary to create a new OpenEdge database.
- Convert an existing OpenEdge or Progress database to OpenEdge.

For more information on creating databases, see [OpenEdge Data Management: Database Administration](#).

You can also create and configure an OpenEdge database server in Windows.



To use OpenEdge features with a server:

1. From the desktop, choose **Start**→ **All Programs**→ **Control Panel**→ **Administrative Tools**→ **Services**. Verify that the status for the **AdminService for OpenEdge Release 11.3** is **Started**.

Note: If **Administrative Tools** is not available, right click from the Task Bar. Choose **Properties**, then select the **Advanced** tab. Select the **Display Administrative Tools** check box, then choose **OK**.

2. Use OpenEdge Management or OpenEdge Explorer Tool to add a database configuration. (You cannot create the physical database with OpenEdge Explorer Tool.)

For more information on the AdminServer and the OpenEdge Management or OpenEdge Explorer, see the [“Understanding and using the AdminServer”](#) section on page 282 and the OpenEdge Management or OpenEdge Explorer online help.

Running OpenEdge

Select an icon from the OpenEdge Program Group to begin running your applications. Note that WebSpeed products might need additional setup requirements.

Caution: Never run OpenEdge products from the directories in which you installed them. Doing so could result in changes to the software that affect its proper operation.

For complete information about starting OpenEdge or WebSpeed products, see either [OpenEdge Getting Started: Progress OpenEdge Studio](#) or [OpenEdge Getting Started: WebSpeed Essentials](#).

Note: Before you start an OpenEdge or an Application Server application, you must start the AdminServer. The AdminServer must be running before you can use the OpenEdge Management or OpenEdge Explorer to configure and manage your applications. For details, see the [“Understanding and using the AdminServer”](#) section on page 282.

Maintaining OpenEdge and Progress

To maintain OpenEdge along with one or more versions of Progress or OpenEdge on your system, you perform a typical installation (that is, a complete or custom installation as described in [Chapter 3, “OpenEdge Installation Prerequisites”](#)), with the following exceptions:

- When you are prompted for a **Destination Directory**, make sure the directory you specify is not the same as for other installed versions. Type the pathname of a separate directory in which to install OpenEdge.
- Redefine your `PATH`, using the `Proenv` command-line utility.
- To run AdminServers for OpenEdge and Progress, you must set unique `-port` and `-adminport` as described in the [“Understanding and using the AdminServer”](#) section on page 282.

Note: To access previous version tools or utilities, you must use complete pathnames.

OpenEdge key and certificate stores

All OpenEdge server and client components that implement Secure HTTP (HTTPS) or Secure Socket Layer (SSL) connections require access to private keys and digital certificates to negotiate these connections and to enable them to function securely.

For all OpenEdge components, OpenEdge provides utilities that allow you to install and manage keys and digital certificates (in key stores and certificate stores) so the components can use them. For Open Clients and Web services clients, OpenEdge provides utilities for some clients or relies on utilities provided by the client platform to manage the required certificate stores.

For more information on managing certificate stores for Open Clients and Web service clients, see [OpenEdge Development: Open Client Introduction and Programming](#). For details about using the OpenEdge utilities to manage key stores for OpenEdge servers and manage certificate stores for OpenEdge clients, see [Chapter 9, “Managing OpenEdge Key and Certificate Stores”](#) and [Appendix C, “Command and Utility Reference.”](#)

Support for IPv6

OpenEdge supports IPv6. If your Windows environment uses this form of network communication, you must explicitly request it during startup in your server properties file, or on the command line.

Specifying IPv6

You can specify IPv6 in one of two ways:

- `-ipver` startup parameter. Add this startup parameter to your command line, as shown:

```
-ipver version
```

- `ipver` property in a `*.properties` file. Add the startup parameter to your `*.properties` file, as shown:

```
ipver=version
```

The startup parameter and the property each take the same values for `version`.

[Table 43](#) shows the possible values and their meaning.

Table 43: Values for specifying IP version

Value	Action
IPv4	Allow connections with IPv4 only. If <code>-ipver</code> is not specified, this is the default behavior.
IPv6	Allow connections with IPv6 and mapped IPv4.

The startup parameter and property name is case sensitive, and must be specified in all lower case. The values for `version` are not case sensitive, and can be specified in any case.

Table 44 shows the appropriate method for specifying IP version.

Table 44: Specifying IP version

To start ...	Specify IP version with ...
AdminServer	<code>-ipver</code> on command line
Unified brokers: <ul style="list-style-type: none"> • AppServer • NameServer • WebSpeed Agent • AppServer Internet Adapter (AIA) 	<code>ipver</code> in <code>ubroker.properties</code>
Database servers	<code>-ipver</code> on command line <code>ipver</code> in <code>commgr.properties</code>
DataServer broker	<code>-ipver</code> on command line <code>ipver</code> in <code>ubroker.properties</code>
Client	<code>-ipver</code> on command line
Debugger	<code>-ipver</code> on command line inherit from client

Note: For information on configuring IPv6 properties for underlying Java code, see the [“Specifying IP version for underlying Java code”](#) section on page 309.

Windows 64-bit

OpenEdge on Windows 64-bit provides access to memory in excess of 2GB to database servers, WebSpeed, AppServer, batch clients, character clients, and GUI clients. In addition, the Microsoft .NET Open Client interface .dll files are included, allowing you to develop 64-bit applications using the Open Client interface as discussed in the following sections:

- [Notes for the 64-bit GUI client](#)
- [Application development and deployment](#)
- [Product and database interactions](#)

Notes for the 64-bit GUI client

Note the following for 64-bit GUI client:

- The 64-bit Windows GUI and character clients ignore the CDECL, PASCAL, and STDCALL calling conventions if they are specified in an external procedure declaration. The 64-bit Windows GUI and character clients always use the standard 64-bit FASTCALL calling convention.
- The 64-bit GUI client supports JPEG (.jpg), Windows Bitmap (.bmp), Windows Icon (.ico), TIFF (.tif), PNG (.png), and GIF (.gif) image file formats.
- The 64-bit GUI client can load and use .NET assemblies compiled for the x64 (64-bit) and Any CPU configurations.
- The cscmb32.ocx, cslist32.ocx, and csspin32.ocx controls that are available for the 32-bit GUI client are not available for the 64-bit GUI client. The only OCX control that is available for 64-bit GUI client is pstimer.ocx.
- OpenEdge products for Windows 64-bit include a 32-bit version of Report Engine.

The following products include the Microsoft .NET Open Client interface .dll files, allowing customers who use the Open Client interface to develop 64-bit applications:

- OpenEdge Workgroup RDBMS
- OpenEdge Enterprise RDBMS
- OpenEdge DataServer for ORACLE
- OpenEdge DataServer for ODBC
- OpenEdge DataServer for Microsoft SQL Server
- OpenEdge Development Server
- OpenEdge Application Server Basic
- OpenEdge Application Server Enterprise

Application development and deployment

Developing and deploying applications for Windows 64-bit, requires special considerations, as discussed in the following sections:

- [ABL Development](#)
- [ABL Deployment](#)

ABL Development

Application development, using OpenEdge development tools, requires the OpenEdge Windows 32-bit product suite. This must be installed on a separate machine. The graphical development products are not part of the Windows 64-bit product, and you can not install both the 32-bit and 64-bit OpenEdge Windows products of the same version on the same machine.

The general steps for development and deployment are:

1. Develop your ABL application in Windows 32-bit.
2. Compile the application on Windows 32-bit.
3. Copy the r-code to Windows 64-bit for deployment.

You are now ready to deploy your application.

Note: If you follow standard guidelines for r-code portability, r-code from a UNIX 64-bit platform can be deployed on Windows 64-bit.

ABL Deployment

Deployment of your application with a Windows 64-bit server, will require a mix of products and versions of r-code. [Figure 5](#) displays this scenario.

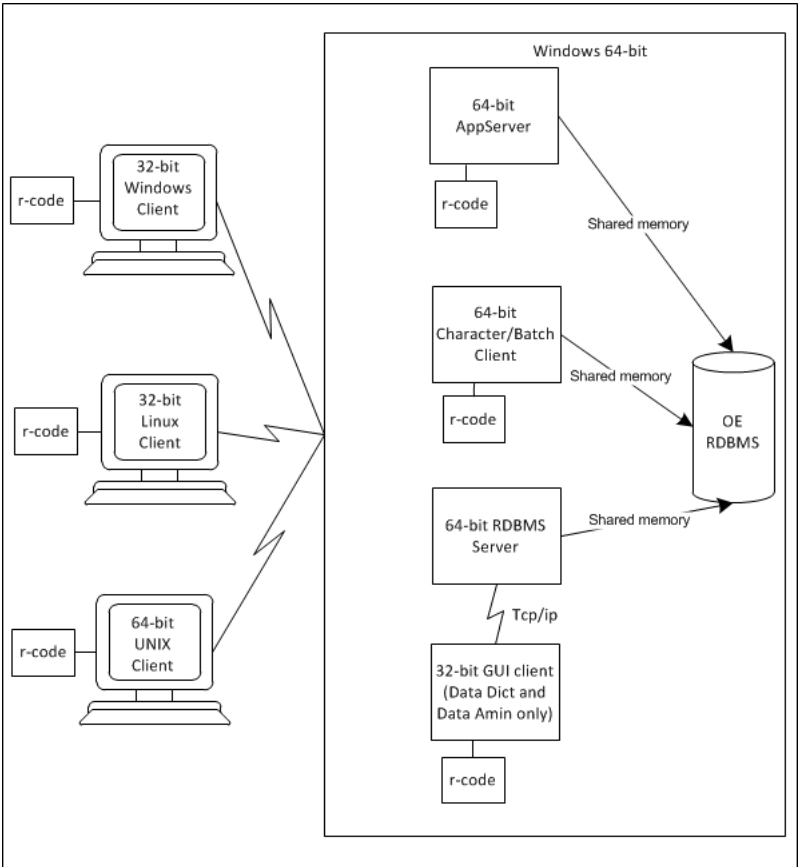


Figure 5: Windows 64-bit deployment

.NET Open Client development

The architectural model for developing a 64-bit application with the Microsoft .NET Open Client interface `.dll` files is depicted in [Figure 6](#).

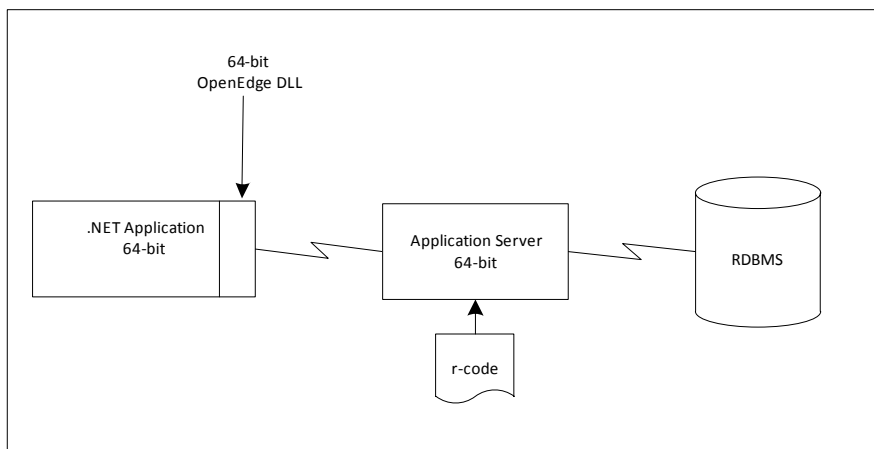


Figure 6: 64-bit .NET Open Client model

For information on developing Open Client applications, see [OpenEdge Development: Open Client Introduction and Programming](#).

Product and database interactions

You cannot have both the 32-bit and 64-bit version of OpenEdge for Windows installed on the same system. However, both versions can access the same OpenEdge database, so there is no need to convert or do a dump and load when migrating from Windows 32-bit to Windows 64-bit.

The 32-bit client (`prowin32.exe`) supplied with the Windows 64-bit product for Data Dictionary and Data Administration functions can access an offline database in single-user mode. If a 64-bit server is started against the database, the 32-bit client cannot connect using shared memory, and must connect to the 64-bit server using a client/server connection.

Working in the OpenEdge Environment on UNIX

This chapter explains how to run OpenEdge on UNIX, as described in the following sections:

- [Default environment variables settings](#)
- [UNIX environment variables](#)
- [Setting Java environment variables](#)
- [Setting SQL client environment variables](#)
- [Using the Proenv utility](#)
- [Getting started with the AdminServer](#)
- [Understanding the built-in terminal definitions](#)
- [OpenEdge key and certificate stores](#)

Default environment variables settings

By default, the OpenEdge installation program tailors all the necessary OpenEdge and Java environment variables to where they are installed. After installing OpenEdge, you can use the command-line utility (Proenv) to access these environment variables.

Caution: Although editing environment variables is an option, this procedure is not recommended if Progress Version 8.3 and Version 9 (or WebSpeed Version 2.x and Version 3.x) products exist on the same system.

For more information on environment variables for OpenEdge, see the information on maintaining user environments in [OpenEdge Deployment: Managing ABL Applications](#) or refer to your specific product documentation.

UNIX environment variables

This section describes the operating system-specific environment variables on a UNIX operating system.

For information about setting environment variables related to OpenEdge AppServer, OpenEdge WebSpeed, an OpenEdge DataServer, or the OpenEdge Adapter for SonicMQ, see [OpenEdge Application Server: Administration](#), [OpenEdge Application Server: Developing WebSpeed Applications](#), the OpenEdge Data Server Guides ([OpenEdge Data Management: DataServer for Microsoft SQL Server](#), [OpenEdge Data Management: DataServer for Oracle](#), and [OpenEdge Data Management: DataServer for ODBC](#)).

After installation, OpenEdge requires little if any additional configuration. However, some environment variables can be customized. As needed, you can access these environment variables using the Proenv command-line utility.

Running the `Proenv` script sets `DLC` to this directory automatically. `Proenv` also adds `$DLC/bin` to your path and changes your current directory to the OpenEdge work directory you set during installation.

You can edit the `.profile` of a user to set up environment variables automatically each time the user logs onto the system. Also, be sure to export environment variables to make them available to child processes.

[Table 45](#) describes the UNIX or Linux environment variables. These descriptions help determine the variables you want to set. Usage with the Bourne shell is given, but other shells use similar syntax.

Table 45: UNIX environment variables*(1 of 4)*

Variable	Description	Code example
DLC	The pathname of the directory where you installed the OpenEdge software. The default value is <code>/usr/dlc</code> . You must set this variable if you install the OpenEdge software in an alternate directory.	<code>DLC=/usr/dlc</code>
PATH	<p>A list of the directories UNIX searches to find any commands that you provide. OpenEdge also searches these directories for UNIX commands or programs you name when using the <code>INPUT THROUGH</code> and <code>OUTPUT THROUGH</code> statements. These directories include:</p> <ul style="list-style-type: none"> • <code>\$DLC/bin</code> in the <code>PATH</code> environment variable. To keep end users out of the <code>/DLC</code> directory, you can provide scripts to perform all OpenEdge-related actions. These scripts can reside somewhere else in the <code>PATH</code> and invoke OpenEdge commands with full pathnames. Place your startup, shutdown, and maintenance scripts somewhere in the path directories. • <code>\$JAVAHOME/bin</code> in the <code>PATH</code> environment variable. This value must be set to ensure that the installation can detect a java installation because the java executable is located in the <code>\$JAVAHOME/bin</code>. <p>Note: If during installation you chose yes to copy scripts to <code>/usr/bin</code>, ensure that <code>PATH</code> is set to <code>/usr/bin</code>: <code>PATH=:usr/bin</code>.</p>	<pre>PATH=\$PATH:\$DLC/bin PATH=\$JAVAHOME/bin</pre>
PROCFG	The filename (or full pathname) of your product's configuration file. The configuration file is a data file that identifies the OpenEdge product and components that you are licensed to use. The default value is <code>\$DLC/progress.cfg</code> . Reset <code>PROCFG</code> if you have moved your configuration file from the directory where you installed OpenEdge.	<code>PROCFG=\$DLC/products.cfg</code>

Table 45: UNIX environment variables

(2 of 4)

Variable	Description	Code example
PROCONV	The filename (or full pathname) of the OpenEdge <code>convmap.cp</code> file. The <code>convmap.cp</code> file is a binary file that contains all of the conversion tables that are available to OpenEdge. The default value is <code>\$DLC/convmap.cp</code> . See OpenEdge Development: Internationalizing Applications for more information on the <code>convmap.cp</code> file.	<code>PROCONV=\$DLC/convmap.cp</code>
PROEXE	The pathname of your OpenEdge executable file. The default value is <code>\$DLC/bin/_progres</code> . If you move <code>_progres</code> out of <code>\$DLC/bin</code> , rename <code>_progres</code> , or use the <code>OEBuild</code> utility to generate a customized module; set <code>PROEXE</code> appropriately (or modify your scripts).	<code>PROEXE=\$DLC/bin/_progres</code>
PROLOAD	The pathname of the directory where you installed the <code>OEBUILD</code> product, if you installed it. The default value is <code>\$DLC/oebuild</code> . For example, if you installed <code>OEBUILD</code> in the directory <code>/voll/devdir/load</code> , use the code example.	<code>PROLOAD=/voll/devdir/load</code>
PROMSGS	The full pathname of your OpenEdge run-time messages file. The default value is <code>\$DLC/promsgs</code> . For example, if you want to use the German run-time messages file, use the code example in your profile. You only set the <code>PROMSGS</code> environment variable if you want to use a run-time messages file different from the default <code>PROMSGS</code> file in the <code>\$DLC</code> directory.	<code>PROMSGS=\$DLC/prolang/ger/promsgs.ger</code>
JDKHOME	Establishes the top-level directory for the Java Developer's Kit (JDK). Note: When you first execute an OpenEdge command or utility that requires Java, OpenEdge correctly sets the Java environment variables based on your version of UNIX.	<code>JDKHOME=\$DLC/jdk</code>
JREHOME	Establishes the top-level directory for the Java Runtime Environment (JRE).	<code>JREHOME=\$DLC/jre</code>
JFCHOME	Establishes the top level directory for the Java Foundation Classes (JFC).	<code>JFCHOME=\$DLC/jfc</code>
CLASSPATH	OpenEdge correctly sets the appropriate classpath variable based on the platform in use.	<code>CLASSPATH=\$JDKCP:\$PROGRESSCP</code>
JIT	Sets the just-in-time compiler correctly.	<code>JIT="-nojit"</code>

Table 45: UNIX environment variables

(3 of 4)

Variable	Description	Code example
JVMEXE	Sets the Java Virtual Machine to run correctly.	JVMEXE=jre
PROPATH	<p>A list of directories OpenEdge searches to find procedures.</p> <p>OpenEdge AppServer and OpenEdge WebSpeed use the <code>PROPATH</code> property in <code>\$DLC/properties/ubroker.properties</code>.</p> <p>Otherwise, by default, OpenEdge searches these subdirectories, using the order specified:</p> <ol style="list-style-type: none"> 1. <code>\$DLC/tty</code> 2. <code>\$DLC</code> 3. <code>\$DLC/bin</code> <p>Use the following syntax to set the <code>PROPATH</code> environment variable: <code>PROPATH=[:] {<i>dir-name</i>/<i>env-var</i>}(...);</code> Where:</p> <p><code>[:]</code> — Tells OpenEdge to search your working directory before searching any other directories</p> <p><i>dir-name</i> — Specifies the name of a directory that you want OpenEdge to search</p> <p><i>env-var</i> — Specifies the environment variable whose definition names one or more directories that you want to search</p> <p><code>(:...)</code> — Separates multiple <i>dir-name</i> or <i>env-var</i></p> <p><code>;</code> — Ends the definition of the <code>PROPATH</code> environment variable and indicates the start of a new command.</p>	PROPATH=:persapp:\$DLC
PROSRV	The pathname of your executable PROSERVE file. The default value is <code>\$DLC/bin/_mprosrv</code> . The PROSERVE script includes the code example. Therefore, if you move <code>_mprosrv</code> out of <code>\$DLC/bin</code> , rename <code>_mprosrv</code> , or use the <code>OEBuild</code> utility to create a customized module, set <code>PROSRV</code> appropriately (or modify your <code>proserve</code> script).	PROSRV=\$DLC/bin/_mprosrv
PROSTARTUP	The pathname of the OpenEdge default startup parameter file, <code>startup.pf</code> . This file is read by all OpenEdge modules at startup; it must exist for OpenEdge to execute properly.	PROSTARTUP=\$DLC/startup.pf

Table 45: UNIX environment variables

(4 of 4)

Variable	Description	Code example
PROTERMCAP	The full pathname of the terminal definition file that you want your OpenEdge session to use. The default terminal definition file is called <code>PROTERMCAP</code> and is installed by default in the <code>/DLC</code> directory. You only have to set the <code>PROTERMCAP</code> environment variable if you want to use a different terminal definition file from the default <code>PROTERMCAP</code> file.	<code>PROTERMCAP=\$DLC/SPECIALCAP</code>
TERM	The type of terminal you are using. For example, to define your terminal type as <code>wy75</code> , use the code example.	<code>TERM=wy75</code>
WRKDIR	The full pathname of the working directory of your OpenEdge installation.	<code>WRKDIR=/usr/wrk</code>
OEM	The full pathname of the directory where OpenEdge Management is installed.	<code>OEM=/usr/oemgmt</code>
OEMWRKDIR	The full pathname of the working directory of OpenEdge Management.	<code>OEMWRKDIR=/usr/wrk_oemgmt</code>
DBUTIL	The pathname of your OpenEdge RDBMS utility executable file. The default value is <code>\$DLC/bin/_dbutil</code> .	<code>DBUTIL=\$DLC/bin/_dbutil</code>

Notes: `$DLC` is an environment variable for the full pathname of the directory where OpenEdge is installed. You can run `Proenv` to automatically set `DLC` to this directory.

If you want to use a remote DataServer, you must set additional environment variables depending on the type of DataServer you want to use (for example, ORACLE or ODBC). See the DataServer documentation for more information on the other variables set.

When you first execute an OpenEdge command or utility that requires Java, OpenEdge correctly sets the Java environment variables based on your UNIX platform.

Setting Java environment variables

By default, the OpenEdge installation program tailors all the necessary OpenEdge and Java environment variables to the directories where they are installed. For example, the installation automatically sets the `%DLC%` environment variable to your OpenEdge installation path.

OpenEdge bundles the Java Runtime Environment (JRE) component and the Java Development Kit (JDK) component with certain products that you install. For more information, see the “[Java considerations](#)” section on page 32. For additional details regarding Java and platform-specific information, see the “[Requirements for using Java](#)” section on page 54. And, for more information on Java environment variables for OpenEdge products, see your specific product documentation.

Notes: OpenEdge supports Java version 6.0. For specific information about these components, see the *OpenEdge 10 Platform & Product Availability Guide* on the Progress Software Corporation Web site
<http://www.progress.com/products/lifecycle/index.ssp>.

Setting the JDK environment variable

In most circumstances, you will not need to set the JDK environment variable. When you load your installation medium, the Installation Program determines whether JVM is on your machine. It also verifies that you have the correct JVM version required to run OpenEdge. See the “[Starting the installation process](#)” section on page 156 for more information. If necessary, you can correctly set up your JDK environment for products that rely on the environment variables set by the script file `$DLC/bin/java_env` (for example, `JDKHOME` and `JREHOME`).

- To correctly set up your JDK environments if this task was not accomplished when your installation medium was loaded, you must edit this file and change the `JDKHOME` value from:

```
#JDKHOME=
```

To:

```
JDKHOME=/usr1/jdk-directory
```

Where `/usr1/jdk-directory` is the JDK install directory.

Note: This modification applies to the HP-UX sections of the `$DLC/bin/java_env` file. The JDK is bundled on the Sun Solaris platform, and therefore is not needed. The root directory owns the `java_env` file, and the individual modifying the file must have root access.

Setting SQL client environment variables

SQL client environment variables are automatically set for you in `$DLC/bin/sql_env`.

Using the Proenv utility

The `$DLC` environment variable is not set at the system level. The `Proenv` utility can automatically set the `$DLC` environment variable to the directory where OpenEdge is installed. It then adds `$DLC/bin` to your `PATH`.

`Proenv` opens a new window, sets the environment variables, and then changes the current directory to the working directory you set when you installed OpenEdge.

Getting started with the AdminServer

The AdminServer is the key controlling element of the Unified Broker framework. It facilitates the tasks associated with managing and configuring your installation.

This section briefly introduces the AdminServer. For details about the AdminServer, its role in the Unified Broker framework, and the tasks to use the AdminServer, see [Chapter 10, “Configuration.”](#)

OpenEdge products supported by the AdminServer

An AdminServer is installed on every system where you install any of the following OpenEdge products are installed:

- OpenEdge Databases: OpenEdge Personal RDBMS, OpenEdge Workgroup RDBMS, and OpenEdge Enterprise RDBMS
- DataServers: OpenEdge DataServer for ORACLE and OpenEdge DataSever for MS SQL Server, and DataServer for ODBC
- OpenEdge Application Server—Basic
- OpenEdge Application Server—Enterprise
- OpenEdge Studio
- Progress Developer Studio for OpenEdge
- Web Services Adapter (AIA)
- AppServer Internet Adapter
- OpenEdge Adapter for SonicMQ
- OpenEdge Adapter for Sonic ESB
- NameServer
- OpenEdge NameServer Load Balancer
- WebSpeed Workshop
- OpenEdge Replication
- OpenEdge Replication Plus

A command-line utility, `PROADSV`, supports OpenEdge administrative capabilities on UNIX. `PROADSV` allows you to start up, shut down, and query the status of the AdminServer. See the sections about the `PROADSV` command in [Chapter 10, “Configuration”](#) for detailed syntax information.

AdminServer considerations

Note the following points that pertain to AdminServer usage:

- Before you start a WebSpeed or AppServer application, you must start the AdminServer.
- The AdminServer User-Group Authorization feature requires that you have privileges set to allow you access and operational privileges for the AdminServer. See the [“How to implement the User-Group Authorization feature”](#) section on page 250.
- The AdminServer must be running before you can use the OpenEdge Explorer from a remote Windows machine to configure and manage your applications.

How to implement the User-Group Authorization feature

To implement the User-Group Authorization feature on a UNIX platform, you must first successfully complete the OpenEdge installation program. [Table 46](#) identifies and briefly describes the purpose of each new command-line option.

Table 46: User-Group parameter options

Parameter name	Syntax	Purpose
Individual user name and password required	<code>-requireusername</code>	Requires a minimum of one user ID to be resolved for each AdminServer operation before it can be executed.
Group authorization required	<code>-admingroup group</code> <code>[{,[:}group...]</code>	Requires a minimum of one group to be resolved for each AdminServer operation before it can be executed. A colon-separated list differentiates groups when you are specifying multiple groups on the command line.

On UNIX platforms, a group name can be any user-defined or NIS group name. UNIX can also support subgroups.

Understanding the built-in terminal definitions

A list of built-in terminal definitions is supplied with OpenEdge as described in the following sections:

- [Terminal issues](#)
- [Terminal identifiers](#)

Terminal issues

When you start OpenEdge, you might receive the following message:

```
** You cannot use DEL for both stty intr and DELETE-CHARACTER.
```

The message as presented in the previous example indicates that you were trying to use the **DEL** key as the UNIX interrupt key and as the OpenEdge **DELETE-CHARACTER** key. To avoid this message, add the following line to your `.profile` file:

```
stty intr ^C
```

This command resets your UNIX interrupt key from **DEL** to **CTRL+C**.

Built-in terminal definitions are supplied with OpenEdge for the terminals listed in [Table 47](#), which indicates the terminal identifiers you can use so that OpenEdge can successfully access that terminal definition. Be sure the operating system environment variable `TERM` is set to the appropriate value. For example:

```
TERM=wyse60;export TERM
```

Terminal identifiers

[Table 47](#) shows a list of terminal identifiers.

Table 47: Terminal identifiers (1 of 2)

Terminal model	Terminal identifier	Notes
xterm	xterm	—
CDE dtterm	dtterm	—
DEC VT100	V1, vt100, vt100-80	Asian languages are supported. For more information on supported languages, see OpenEdge Development: Internationalizing Applications .

Table 47: Terminal identifiers

(2 of 2)

Terminal model	Terminal identifier	Notes
DEC VT200 series	V2, vt200, vt200-80, vt220, vt220-80, vt240, vt241	Asian languages are supported. For more information on supported languages, see OpenEdge Development: Internationalizing Applications .
DEC VT300 series	V3, vt300, vt320, vt330, vt340, pt300, pt-100	—
DEC VT400 series	V4, vt400, vt400-80, vt420	—
DEC VT500 series	V5, vt500, vt500-80, vt510, vt520, vt525	—
IBM 3151	3151, m3, ibm3151	—
IBM PC/AT XENIX console	li, ansi	Ansi driver.
IBM PC/AT XENIX color console	lc, ansic	Ansi driver. Uses reverse video for input fields.
Linux console	linux, linux-lat	—
Sun console	Mu, sun	—
Wyse 60	w6, wy60, wyse60 wy60t, wyse60tall wy60w, wyse60wide wy60tw, wyse60tall + wide	Assumes that the function keys are set to the factory defaults. Check the PROTERMCAP entry for setup mode. Terminal in following mode: <ul style="list-style-type: none"> • 43 lines X 80 columns • 25 lines X 132 columns • 43 lines X 132 columns
Wyse 370	wy370, wyse370	—

Additional terminal identifier considerations

The following points are relevant to terminal identifiers:

- [Table 47](#) is complete as of the print date of this guide.
- The IBM native console terminal type `hft` is not supported.
- OpenEdge does not support spacetaking terminals, unless the terminal has a firmware setup option to change it to a non-spacetaking mode.



To determine your terminal identifier if it is not listed in [Table 47](#):

1. Try to run OpenEdge using a terminal definition for a terminal that functions similarly to yours, or try to configure your terminal to emulate one of the supported terminals.

Note: Progress Software Corporation does not support terminal emulation.

2. In the directory where you installed your OpenEdge product, find the `PROTERMCAP` file that contains terminal definitions.

Note: All terminal types supported by OpenEdge are documented in the `$DLC/protermcap` file.

3. Search through the `PROTERMCAP` file to see if your terminal is listed. The `PROTERMCAP` file is similar in structure to the UNIX `/etc/termcap` file. Each terminal type is followed by a description of that terminal. For more information about the `PROTERMCAP` file, see [OpenEdge Deployment: Managing ABL Applications](#).

OpenEdge key and certificate stores

All OpenEdge server and client components that implement Secure HTTP (HTTPS) or Secure Socket Layer (SSL) connections require access to private keys and digital certificates to negotiate these connections and to enable them to function securely.

For all OpenEdge components, OpenEdge provides utilities that allow you to install and manage keys and digital certificates (in key stores and certificate stores) so the components can use them. For Open Clients and clients of Web services, OpenEdge provides utilities for some clients or relies on utilities provided by the client platform to manage the required certificate stores.

For more information on managing certificate stores for Open Clients and Web service clients, see [OpenEdge Development: Open Client Introduction and Programming](#). For details about using the OpenEdge utilities to manage key stores for OpenEdge servers and manage certificate stores for OpenEdge clients, see [Chapter 9, “Managing OpenEdge Key and Certificate Stores”](#) and [Appendix C, “Command and Utility Reference.”](#)

Managing OpenEdge Key and Certificate Stores

All OpenEdge server and client components that implement Secure HTTP (HTTPS) or Secure Socket Layer (SSL) connections require access to private keys and digital certificates to negotiate these connections and to enable them to function securely.

For all OpenEdge components, OpenEdge provides utilities that allow you to install and manage keys and digital certificates (in key stores and certificate stores) so the components can access them. For Open Clients, clients of OpenEdge Web services, OpenEdge provides utilities for some clients or it relies on utilities provided by the client platform to manage the required certificate stores.

This chapter describes how to use the OpenEdge utilities, as detailed in the following sections:

- [Managing key stores for OpenEdge servers](#)
- [Managing certificate stores for OpenEdge clients and servers](#)

An SSL server requires access to a private key and a digital (public-key) certificate to authorize the identity of the server. Clients require access to public-key certificates that allow them to authenticate the servers that they access. Both servers and clients must obtain their keys and certificates from a trusted source, a Certificate Authority (CA). The server can trust the CA to authorize the server's identity and the client can trust the CA to provide proof of the server's identity. For more information on keys, certificates, and how CAs support them, see the chapters on security in [OpenEdge Getting Started: Core Business Services - Security and Auditing](#).

Managing key stores for OpenEdge servers

You can manage the private keys and the corresponding digital certificates for OpenEdge servers that support SSL connections using a *key store* located in the `OpenEdge-Install-Dir\keys` directory. Each SSL server requires at least one *key store entry* that contains a single private key and corresponding digital (public-key) certificate. With this key store entry, you can configure any supported OpenEdge server to enable and manage SSL connection from clients. For more information on the OpenEdge servers that support SSL server configuration, see the sections on the OpenEdge-supported SSL server components described in [OpenEdge Getting Started: Core Business Services - Security and Auditing](#).

If you require only data encryption and do not need to verify the identity of SSL servers (typically, for intranet configurations only), OpenEdge comes installed with a default key store entry. This default entry contains a common private key and digital certificate pair that you can use without any further management beyond enabling SSL connections on OpenEdge clients and servers. For more information on the default SSL server identity, see the sections on SSL in [OpenEdge Getting Started: Core Business Services - Security and Auditing](#).

However, to establish a trusted OpenEdge SSL server identity suitable for use on the Internet or a more secure intranet, you must complete several steps using the functions of the `pkiutil` and `certutil` command-line utilities installed with OpenEdge.

Notes: Before you run an OpenEdge command-line utility, set the `DLC` environment variable to the `OpenEdge-Install-dir` pathname and set the `WRKDIR` environment variable to your working directory. For an example, see the `OpenEdge-install-dir/bin/pkiutil` shell script on UNIX or the `OpenEdge-install-dir\bin\pkiutil.bat` file in Windows.

Running the command-line utility in a `Proenv` command window properly sets `DLC` and `WRKDIR` for you.

Establishing a trusted SSL server identity

There are several steps required to establish a trusted identity for any OpenEdge SSL server using the `pkiutil` command-line utility.

Caution: While the `default_server` key store entry provided by the Progress Server Certificate Authority also uses a default password ("`password`"), you must password-protect any private key store entries that you create from a public-key certificate issued by a trusted external CA. The secrecy of your password is critical to using this key store entry for authenticating a server.



To establish and maintain a trusted SSL server identity using the `pkiutil` utility:

1. Use the `-newreq` operation to generate a proposed public and private-key pair together with a digital certificate request that is suitable for sending to any CA for authorization. You must provide a password to secure this certificate request. You must later provide this password to any OpenEdge server which you want to access this key store entry for securing SSL connections to it. See the [“Supplying a key store entry password to an OpenEdge server”](#) section on page 257.
2. Use e-mail (or some other method required by the CA) to send a copy of the certificate request to the trusted CA you want to return a public-key certificate. This process authenticates any server providing access to the private key.
3. Use the `-import` operation to import the digital certificate returned by the CA for this request and store it together with the associated private key as an entry in the key store.
4. Use the `-display` or `-list` operations to review an individual digital certificate file or any and all key store entries for important digital certificate information, such as expiration dates.
5. Use the `-remove` operation to remove any unused or expired key store entries that you specify and retain them in a backup area of the key store.

For an overview of the `pkiutil` command-line utility, see the [“Using pkiutil to manage an OpenEdge key store”](#) section on page 257.

Supplying a key store entry password to an OpenEdge server

When you configure an OpenEdge server to access a key store entry, you must provide it with the same password that you used to create the key store entry. If you configure the server using OpenEdge Management or OpenEdge Explorer, you can enter this password directly in the fields provided. However, if you configure the server by manually editing the `ubroker.properties` file for that server or specifying the password on a command line or in a startup script (as required when starting a database server for the OpenEdge RDBMS), you must provide an encrypted value for the password. This will protect the password itself from being easily discovered. OpenEdge provides the `genpassword` command-line utility for obtaining a password's encrypted value. For more information, see the [“Using genpassword to obtain a key store password-encrypted value”](#) section on page 260.

Using pkiutil to manage an OpenEdge key store

The `pkiutil` command-line utility provides all the operations necessary to create and manage key store entries for OpenEdge SSL servers (see the [“Managing key stores for OpenEdge servers”](#) section on page 256). This utility manages all input and output for the key store in the `OpenEdge-Install-Dir\keys` directory. For more information on the structure of this directory, see the [“Understanding key store content”](#) section on page 258.

The `pkiutil` utility has the following general command-line syntax:

Syntax

`pkiutil [options] function arguments`

options

Change the type of information and defaults for different functions (*function*) of the utility.

function arguments

One of the following functions (*function*) and the objects they affect (*arguments*):

- **`-newreq alias`** — Generates a new private/public-key pair and a corresponding public-key certificate request (suitable for submission to a CA), stored under the alias name specified by *alias*
- **`-import alias cert-file`** — Imports a CA-issued SSL server digital (public-key) certificate from the disk file *cert-file*, pairs it with the private key generated for a public key request identified by the alias name *alias*, and places the pair in the key store as a new entry identified by *alias*
- **`-print alias`** — Displays the public-key certificate request identified by *alias*.
- **`-list [alias ...]`** — Displays a list of specified (*alias*) or all current key store entries
- **`-display cert-file`** — Displays the digital certificate file information contained in the operating system disk file *cert-file*
- **`-remove alias ...`** — Removes one or more specified (*alias*) key store entries

For complete information on the options and functions of the `pkiutil` command-line utility, see [Appendix C, “Command and Utility Reference.”](#)

Understanding key store content

The OpenEdge key store maintains private keys and digital certificates for OpenEdge SSL servers in several locations. These include private keys and digital certificates that you have authorized by a CA and imported for use by an SSL server, and private keys and public-key certificate requests that you generate and have pending for authorization by a CA. You must manage this key store entirely with the `pkiutil` command-line utility. See the [“Using pkiutil to manage an OpenEdge key store”](#) section on page 257 for additional information.

The key store resides in the `OpenEdge-Install-Dir\keys` directory. This directory contains the following files and subdirectories:

- **alias.pem** — Files containing a single key store entry that you have created from an imported CA-authorized digital certificate that contains the public key joined with the private key that you generated along with the original public-key certificate request. Each file is named with the *alias* that you chose for the original private key and certificate request using the `-newreq` operation of `pkiutil`. The initial key store entry is the default OpenEdge entry `default_server.pem`, as authorized by the Progress Software Corporation CA. For more information on this default key store entry, see the sections on SSL in [OpenEdge Getting Started: Core Business Services - Security and Auditing](#).
- **policy** — A subdirectory containing a `psecpki.cnf` configuration file. The `pkiutil` utility uses this file to control the process of generating new SSL server private/public keys and generating digital certificate requests that can be sent to a CA in order to obtain a public-key certificate for the OpenEdge SSL server. Initially, this is the only subdirectory.
- **requests** — A subdirectory containing all newly generated private keys and public-key certificate requests in the form of the following two files:
 - **alias.pk1** — This file holds the PKCS #1-formatted, password-encrypted, private key for the given key store *alias* entry.
 - **alias.pk10** — This file holds the PKCS #10-formatted public-key certificate request that you send to a CA to obtain the SSL server's public-key certificate for the given key store *alias* entry.
- **backup** — A subdirectory containing any removed key store entries. The `pkiutil` utility removes an existing key store entry when you:
 - Explicitly remove it using the `-remove` operation of `pkiutil`.
 - Update an existing key store entry with a new digital certificate. You will perform this operation when the previous public-key certificate has expired and you have applied to the CA for a renewed public-key certificate.

In all cases, `pkiutil` places removed key store entries in this directory in case you find it necessary to recover and use them again.

Note: Performing successive `-remove` or `-import` operations on the same key store entry repeatedly overwrites that entry in the `backup` subdirectory.

Caution: If you upgrade or uninstall OpenEdge, Progress Software Corporation recommends that you back up your current version of the OpenEdge key store directory tree (`OpenEdge-Install-Dir\keys`) to prevent losing valuable keys and certificates.

Using genpassword to obtain a key store password-encrypted value

When you must configure an OpenEdge SSL server by manually editing the `ubroker.properties` file, or for the OpenEdge RDMS when you start up the database server to enable SSL connections, you must specify the password to allow access to the required private-key alias. The value you specify is available to anyone who can read the file or command line where you enter it. In order to prevent access to this password by unauthorized users, you must specify an encrypted form of the password that is equivalent to the password itself.

Note: You must also provide the encrypted form of the password ("`password`") for the `default_server` alias. In OpenEdge Management or OpenEdge Explorer, when you configure an SSL server with the `default_server` alias, OpenEdge automatically provides the encrypted form of this password.

OpenEdge provides the `genpassword` command-line utility that you can use to obtain the encoded and encrypted form for the real password.

For example, when the following code is executed in the OpenEdge `Proenv` command window, you can generate an encrypted value for a password whose value is `"topsecret"`:

```
proenv>genpassword -password topsecret
243d3f343726213624

proenv>
```

Later, to verify that an existing encrypted value matches the real password value, you can run `genpassword`, as follows:

```
proenv>genpassword -password topsecret -verify 243d3f343726213624
The passwords match.

proenv>
```

For more information on the options of the `genpassword` command-line utility, see [Appendix C, "Command and Utility Reference."](#)

Managing certificate stores for OpenEdge clients and servers

You can manage trusted CA/root digital (public-key) certificates for OpenEdge clients and servers that support SSL connections using a *root certificate store* located in the `OpenEdge-Install-Dir\certs` directory. Each OpenEdge SSL client and server requires the *root certificate store entry* that contains the public-key certificate from the CA who signed and issued the public-key certificate for the SSL server that the client and server needs to access. Without access to this CA's root digital certificate the OpenEdge clients and servers will be unable to validate the identity of the SSL server and will abort the SSL connection process. For more information on the OpenEdge client and server components that support SSL client and server configuration, see the sections on the supported SSL client components in [OpenEdge Getting Started: Core Business Services - Security and Auditing](#).

If you require only data encryption and do not need to verify the identity of SSL servers (typically, for intranet configurations only), OpenEdge comes with the root digital certificate from the Progress Software Corporation CA (who also signed and issued the `default_server` key store digital certificate for OpenEdge SSL servers already installed). The Progress Software Corporation CA root digital certificate is distributed in PEM format as `d9855a82.0` and in DER format as `pscca.cer` (suitable for importing into a Windows workstation for use by an OpenEdge .NET Open Client). This default entry contains a common root public-key certificate that you can use to access any supported OpenEdge SSL server. For more information on the default root public-key certificate, see the sections on the OpenEdge default server identity in [OpenEdge Getting Started: Core Business Services - Security and Auditing](#).

Installing trusted CA/root certificates

To allow OpenEdge client access to an SSL server whose identity you need to verify, you must install the appropriate root digital certificate to authenticate that server. An SSL server can have its identity established from one of two basic sources:

- One of the trusted public CA root digital certificates distributed by Progress Software Corporation that includes RSA, Thawte, and Verisign
- A root digital certificate from an internal CA that you have set up on your own certificate server or from another external or public CA other than RSA, Thawte, or Verisign

OpenEdge automatically installs root certificates in the OpenEdge root certificate store from RSA, Thawte, and Verisign. However, if you use your own internal-use CA or a public CA other than these three, you must install the required root certificates yourself.

OpenEdge provides the following command-line utilities to install and manage root certificates in the OpenEdge certificate store:

- **certutil** — Installs, lists, and manages CA/root certificates from any CA as entries in the OpenEdge root certificate store, and manages the certificate store for the client and server. You can also remove certificate store entries using this utility. The utility moves all removed entries to a `backup` subdirectory of the root certificate store for future recovery and use.

Note: For .NET and Java Open Clients and Web service clients of OpenEdge application servers, you must use other utilities to manage the root certificate stores for those clients and servers. For more information, see [OpenEdge Development: Open Client Introduction and Programming](#).

- **mkehashfile** — Provides simple installation of PEM-encoded root certificates into the OpenEdge root certificate store from any CA, but provides no other management functions for the OpenEdge certificate store. You can use `certutil` for the additional root certificate management.

Notes: Before you run an OpenEdge command-line utility, set the `DLC` environment variable to the `OpenEdge-install-dir` pathname and set the `WRKDIR` environment variable to your working directory. For an example, see the `OpenEdge-install-dir/bin/certutil` shell script on UNIX or the `OpenEdge-install-dir\bin\certutil.bat` file in Windows.

Running the command-line utility in a `Proenv` command window properly sets `DLC` and `WRKDIR` for you.

Using certutil to manage an OpenEdge root certificate store

The `certutil` command-line utility provides functions to install root certificates from any CA and to manage all of the entries in the OpenEdge root certificate store.

The `certutil` utility has the following general command-line syntax:

Syntax

```
certutil [ options ] function arguments
```

options

Changes the type of information and defaults for different functions (*function*) of the utility.

function arguments

Specify one of the following functions (*function*) and the objects they affect (*arguments*):

- **-import cert-file** — Imports a trusted CA root certificate from the disk file *cert-file*, and creates a root certificate store entry identified by a generated alias name (*alias*, as specified for other functions of this utility)
- **-list [alias ...]** — Displays a list of specified (*alias*) or all current certificate store entries
- **-display cert-file** — Displays the digital certificate file information contained in the operating system disk file *cert-file*
- **-remove alias ...** — Removes one or more specified (*alias*) certificate store entries

For more information on the options and functions of the `certutil` command-line utility, see [Appendix C, “Command and Utility Reference.”](#)

Using mkhashfile to install root certificates in the OpenEdge root certificate store

The `mkhashfile` command-line utility provides a simple way to install a root certificate that is authorized by your own internal-use CA, or any CA that can provide you with a PEM-encoded certificate (typically in a file named with the `.pem` extension). If you are using your own certificate server to provide the certificate, refer to the documentation for the certificate server administration software for information on how to obtain PEM-encoded certificates. Once you have the certificate accessible to your OpenEdge SSL client machine, you can use the `mkhashfile` command-line utility to install it in the OpenEdge root certificate store.

Note: If the root certificate is not a PEM-encoded certificate, it is recommended that you use the `certutil` command-line utility, specifying the format option. For details about the `certutil` command-line utility and all its options and functions, see the detailed syntax information for the `certutil` command listed in [Appendix C, “Command and Utility Reference.”](#)

To use `mkhashfile` to create an entry in the OpenEdge root certificate store for a local PEM-encoded certificate file, `vsigntca.pem`, specify the file with the `mkhashfile` command that you enter in the OpenEdge `Proenv` command window. For example:

```
proenv>mkhashfile vsigntca.pem

OpenEdge Release 11.3 as of Fri Oct 14 00:15:12 EST 2011

Running SSLC command ...
Copying vsigntca.pem and 18d46017.0 to C:\Progress\OpenEdge\certs
proenv>
```

The utility generates the entry as a file with an encrypted filename, `18d46017.0`, which is the alias used to identify the certificate store entry. You can then manage this entry along with all other entries in the OpenEdge certificate store using the `certutil` utility. For more information see the [“Using certutil to manage an OpenEdge root certificate store”](#) section on page 263.

For more information on the `mkhashfile` command-line utility, see [Appendix C, “Command and Utility Reference.”](#)

Configuration

Once you have installed OpenEdge, you can perform configuration tasks as needed to support your application goals. This chapter introduces OpenEdge Explorer, a common administrative architecture for installed OpenEdge server products, focusing on Unified Brokers, as described in the following sections:

- [Introducing OpenEdge Management and OpenEdge Explorer](#)
- [Overview of OpenEdge Management or OpenEdge Explorer](#)
- [Working with Unified Brokers](#)
- [Understanding and using the AdminServer](#)
- [Using OpenEdge Management or OpenEdge Explorer](#)
- [Mergeprop utility overview](#)
- [Ubroker.properties file and product configurations](#)
- [Command-line utilities reference](#)

Introducing OpenEdge Management and OpenEdge Explorer

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. OpenEdge Management monitors the following:

- Local and remote OpenEdge databases
- System resources (CPU, disk, memory, file system)
- File resources
- OpenEdge server resources (AppServer, NameServer, DataServers for ODBC, Oracle, and MS SQL Server, and WebSpeed Transaction Server)
- WebSpeed Messengers
- Adapters (AppServer Internet Adapter, SonicMQ Adapter, and Web services adapter)
- TCP-based network services

In addition to monitoring, you can use OpenEdge Management to configure database, server, Messenger, and adapter properties.

Progress Software Corporation believes that you need a product that provides the best business and development solution, plus the highest level of services and support to back it up. OpenEdge Management's deep monitoring provides more information and more detail about your environment, enabling you to proactively manage operations and make your workday easier.

OpenEdge Explorer provides the functionality currently available in OpenEdge Explorer, but within the OpenEdge Management console. You can set configuration properties, start or stop, and view the status of log files for various OpenEdge resources.

For detailed information about OpenEdge Management and OpenEdge Explorer, see the following books:

- [*OpenEdge Management and OpenEdge Explorer: Getting Started*](#)

Describes how to start OpenEdge Management and OpenEdge Explorer for the first time and how to establish initial configuration settings and secure communications. It also describes the management console and how to set up remote monitoring and configuration for OpenEdge Management and remote configuration for OpenEdge Explorer.

- [*OpenEdge Management and OpenEdge Explorer: Configuration*](#)

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, WebSpeed Transaction Servers, WebSpeed Messengers, and SonicMQ Adapters in OpenEdge Management and OpenEdge Explorer. In addition, this manual also includes details about viewing status and log files. [*OpenEdge Management: Resource Monitoring*](#)

Provides detailed information about the management console; the procedures to set up and run resource monitors, jobs, job templates; and the procedures to perform OpenEdge Management-based import and export activities.

- [*OpenEdge Management: Database Management*](#)

Describes how to use OpenEdge Management to monitor and manage OpenEdge database resources.

- [*OpenEdge Management: Alerts Guide and Reference*](#)

Presents alert concepts and procedures and provides a comprehensive reference section to assist you in working with the OpenEdge Management alerts feature.

- [*OpenEdge Management: Servers, DataServers, Messengers, and Adapters*](#)

Describes how OpenEdge Management supports monitoring and managing specific resources associated with the OpenEdge server products (AppServer, WebSpeed Transaction Server, and NameServer), DataServers (ODBC, Oracle, and MS SQL Server), WebSpeed Messengers, and Adapters (AppServer Internet Adapter, SonicMQ Adapter, and Web Services Adapter).

- [*OpenEdge Management: Reporting*](#)

Provides detailed information about creating and working with report instances and templates.

- [*OpenEdge Management: Trend Database Guide and Reference*](#)

Describes how to manage the OpenEdge Management Trend Database by compacting and purging data. This book also provides a detailed look at the Trend Database schema.

Overview of OpenEdge Management or OpenEdge Explorer

OpenEdge Explorer is a common administrative architecture for installed OpenEdge server products, as described in the following sections:

- [Introduction](#)
- [OpenEdge Management or OpenEdge Explorer elements and descriptions](#)
- [Additional OpenEdge Management or OpenEdge Explorer considerations](#)

Introduction

OpenEdge Management or OpenEdge Explorer is a system administration utility that provides a consistent interface in which specific OpenEdge products can be managed. OpenEdge Explorer supports common administrative tasks and activities you can use to start and stop processes, and to manage, configure, and validate properties for specific OpenEdge products. The AdminServer process enables supported products to address their specific requirements. The AdminServer also supports various management utilities to provide similar configuration and management functions for all of these products. For a complete list of products that use the AdminServer, see the [“OpenEdge products supported by the AdminServer”](#) section on page 226.

Some OpenEdge products administered through and managed by OpenEdge Management or OpenEdge Explorer are designed to help manage an application's resources. For example, these products are based on receiving and sending requests through brokers. Brokers poll for available resources (that is, client and agents), attempting to fulfill these requests. OpenEdge Management or OpenEdge Explorer facilitates the common administrative tasks and configuration activities that are fundamental to the technology these broker-based products use.

OpenEdge products that support broker functionality include:

- **Unified Brokers** — WebSpeed, AppServer, DataServer for MS SQL Server, Oracle DataServer, and ODBC DataServer
- **Adapters** — AppServer Internet Adapter, Web Services Adapter, and OpenEdge Adapter for SonicMQ
- **Messengers** — CGIIP, WSASP, WSISA, and WSNSA

Data for each Unified Broker product is stored in a common text file called `ubroker.properties` file. The file stores the property and configuration information for each Unified Broker. OpenEdge Management or OpenEdge Explorer, and tools like the `mergeprop` utility, help you use to manage the contents of these files.

This chapter focuses on the Unified Brokers and how OpenEdge Management or OpenEdge Explorer supports them to manage an application's resources and make these resources available to clients.

Note: The OpenEdge database is another key product that is part of OpenEdge Management or OpenEdge Explorer. In contrast to the Unified Brokers and their relationship to the `ubroker.properties` file, all configuration changes made to any database administered through the AdminServer are stored in the Configuration Manager properties (`conmgr.properties`) file. For information on OpenEdge database administration, see [OpenEdge Data Management: Database Administration](#).

OpenEdge Management or OpenEdge Explorer elements and descriptions

[Figure 7](#) shows the conceptual relationship among the OpenEdge Management or OpenEdge Explorer elements, products, and tools. It also identifies each element of OpenEdge Management or OpenEdge Explorer, briefly defines the element, and points to other sections within this document or the OpenEdge documentation set where additional information about an element can be found.

Although all Unified Brokers are controlled through the same Unified Broker Properties file (`ubroker.properties`), each broker type maintains a unique port separate from any other server in the group. Any configuration change made to a Unified Broker administered through the AdminServer is stored in the `ubroker.properties` file.

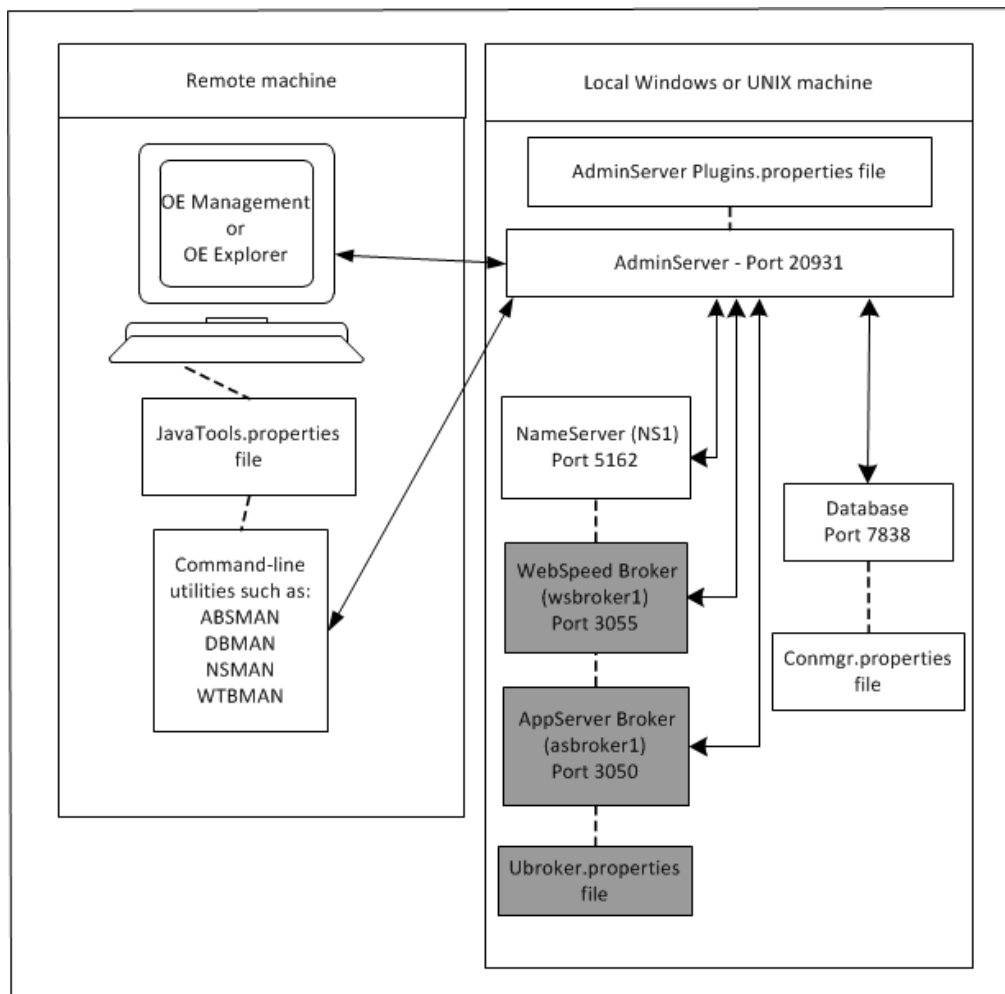


Figure 7: Overview of the OpenEdge Management or OpenEdge Explorer interactions

The shaded elements in [Figure 7](#), WebSpeed and AppServer and Ubroker.properties, are two of the Unified Brokers products. They are intended to represent all Unified Broker products in this graphic to show how the Unified Brokers relate to the other elements. The OpenEdge Management or OpenEdge Explorer only runs in Windows. However, you can connect to remote UNIX systems and administer various supported components on those remote UNIX systems.

[Table 48](#) highlights and describes each element of OpenEdge Management or OpenEdge Explorer.

Table 48: Elements of OpenEdge Management or OpenEdge Explorer(1 of 3)

Element	Description	For more information about this element, see ...
AdminServer ¹	<p>As the central control, the AdminServer:</p> <ul style="list-style-type: none"> Manages each instance of an installed OpenEdge server² product by providing administrative access to OpenEdge products installed on your network Governs remote management and configuration capabilities Supports an administrative feature which users can set to limit access to OpenEdge products 	The “ Getting started with the AdminServer ” section on page 225 (Windows platforms), the “ Getting started with the AdminServer ” section on page 249 (UNIX platforms), and Appendix G, “AdminServer Authorization and Authentication.”
OpenEdge Management or OpenEdge Explorer configuration tool ³	<p>A graphical user interface tool to:</p> <ul style="list-style-type: none"> Initiate administrative tasks on local or remote machines that require a running AdminServer Perform a variety of administrative, managerial, configuration, and validation activities for OpenEdge products 	The “ Using OpenEdge Management or OpenEdge Explorer ” section on page 288; also, see the OpenEdge Management or OpenEdge Explorer online help for extensive information about using the tool.
Mergeprop utility ³	A command-line utility that supports functionality similar to that supported by the OpenEdge Management or OpenEdge Explorer configuration tool. The mergeprop utility can be used as an alternative approach when the OpenEdge Management or OpenEdge Explorer configuration tool is not available.	The “ Mergeprop utility overview ” section on page 291.
Command-line utilities	A basic command-line tool that allows you to control (that is, start, stop, and query) servers and validate property files associated with OpenEdge products.	The “ Command-line utilities reference ” section on page 311.

Table 48: Elements of OpenEdge Management or OpenEdge Explorer(2 of 3)

Element	Description	For more information about this element, see ...
Unified Broker	<p>As the central control within particular OpenEdge products, the Unified Broker is:</p> <ul style="list-style-type: none"> • The key process through which each of these products' resources are individually managed by the product, and these resources are made available to clients. • A collective term used to identify specific OpenEdge products that employ the same mechanism to implement the broker processes. • A standard processing technology within certain OpenEdge products. 	The "Working with Unified Brokers" section on page 276.
ubroker.properties file ³ (Unified Broker properties file)	<p>Common text file location in which data for each Unified Broker⁴ product is stored.</p> <p>The OpenEdge Management or OpenEdge Explorer's and the mergeprop utility's capabilities can be applied to the contents of the <code>ubroker.properties</code> file to manage, configure, or validate properties for each of these products.</p> <p>The <code>ubroker.properties</code> file is located in <code>OpenEdge-install-dir/properties/</code> directory.</p>	The "Ubroker.properties file and product configurations" section on page 304.

Table 48: Elements of OpenEdge Management or OpenEdge Explorer(3 of 3)

Element	Description	For more information about this element, see ...
<code>conmgr.properties</code> file (Configuration Manager properties file)	<p>Common text file that contains configuration information for all OpenEdge databases⁵.</p> <p>The OpenEdge Management or OpenEdge Explorer's and mergeprop utility's capabilities can be applied to the contents of the <code>conmgr.properties</code> file to manage, configure, or validate properties for each of these products.</p> <p>The <code>conmgr.properties</code> file is located in <code>OpenEdge-install-dir/properties/</code> directory.</p>	The "Ubroker.properties file and product configurations" section on page 304; also, see OpenEdge Data Management: Database Administration .
AdminServer <code>plugins.properties</code> file	<p>Common text file that contains information for plugins to be loaded and managed by the AdminServer. The <code>AdminServer.plugins.properties</code> file is located in <code>OpenEdge-install-dir/properties/</code> directory.</p>	The "Changing the default port" section on page 284
<code>JavaTools.properties</code> file (OpenEdge clients' configuration file)	<p>Common text file that contains configuration information for all OpenEdge clients. The <code>JavaTools.properties</code> file is located in <code>OpenEdge-install-dir/properties/</code> directory.</p> <p>Note: Do not make user-modifications to these property files as these properties support OpenEdge and Progress products only.</p>	Contact Progress Technical Support if you want to modify these properties.

1. The AdminServer must be running to use the management command-line management utilities (such as ASBMAN, DBMAN, NSMAN, and WTBMAN) or the OpenEdge Management or OpenEdge Explorer configuration tools. Only mergeprop and the OpenEdge Management or OpenEdge Explorer configuration tool perform the actual configuration of Progress server products and such changes affect the data stored in the `ubroker.properties` or `conmgr.properties` files.
2. See the "OpenEdge products supported by the AdminServer" section on page 226 (Windows platforms) and the "OpenEdge products supported by the AdminServer" section on page 249 (Unix platforms) for specific products.
3. Commands entered and accepted either through OpenEdge Management or OpenEdge Explorer tool or the mergeprop tool immediately affect the data stored in the `ubroker.properties` file or `conmgr.properties` file.
4. The Unified Broker products include these OpenEdge servers: AppServer, WebSpeed, NameServer, and the Oracle DataServer, the DataServer for MS SQL, and the DataServer for ODBC. See the specific book within the product documentation set for more information about each Unified Broker product.
5. Only those OpenEdge databases that are configured to autostart will start when the AdminServer starts.

Additional OpenEdge Management or OpenEdge Explorer considerations

In addition to the footnotes associated with [Table 48](#), note the following relevant to [Figure 7](#):

- The AdminServer, through its default port 20931 as shown in the diagram, processes start, stop, and query requests initiated from a requesting OpenEdge product. Similarly, the database, NameServer, AppServer, and WebSpeed brokers have their own default ports.

As part of setting up and maintaining security measures for your machines, it is advisable to change default port numbers and server names. Doing so helps to protect the identities of these ports from personnel outside your organization. You might also consider documenting and monitoring all of your ports (that is, the port numbers and types) that you use for the AdminServer.

- The dotted lines connecting the `ubroker.properties` file and `conmgr.properties` file to their respective OpenEdge products is intended to indicate that the commands entered and accepted either through OpenEdge Management or OpenEdge Explorer or the `mergeprop` utility directly affect the data stored in the `ubroker.properties` file or `conmgr.properties` file.
- Ensure that you backup the `ubroker.properties` and `conmgr.properties` files periodically because they contain the detailed configuration data for each OpenEdge product.

Configure OpenEdge Explorer post installation

If you did not set OpenEdge Explorer while installation to work with the server functionality in Progress Developer Studio for OpenEdge, you can manually configure OpenEdge Explorer.

To configure OpenEdge Explorer post installation:

1. Open `fathom.init.params` file located at the installation directory `$DLC/OpenEdge`.
2. Set the `SET fathomEnabled` value to `true`.

Note: You must restart Progress Developer Studio for the new value to take effect.

Working with Unified Brokers

The Unified Broker products include a Unified Broker process that is the initial point of client connection to a Unified Broker product instance. This broker process is responsible for managing other process resources that are part of the product, and making those resources available to clients. For more information about the “[Unified Broker products and associated clients](#)” section on page 305.

A Unified Broker and its related components can be set up to run locally or remotely.

Running locally

When a Unified Broker is run locally, the Unified Broker and all of its components are on the same machine. All Unified Broker products require that these components reside on the same machine: the Unified Broker instance and associated processes, the AdminServer, and the `ubroker.properties` file.

Running remotely

When a Unified Broker is run remotely, some components are distributed on separate machines, but connected on the same network.

When a Unified Broker product is distributed remotely, a separate AdminServer and `ubroker.properties` file must exist on each machine for access by OpenEdge Management or OpenEdge Explorer. And, for WebSpeed, the Unified Broker Client (that is, WebSpeed Messenger) also resides on the same machine as the Web server, and Web clients (that is, browsers) can reside anywhere on the Internet, intranet, or extranet serviced by the Web server.

Regarding a DataServer, the separate database host for a DataServer applies only to WebSpeed or the AppServer. For a DataServer, the Unified Broker host is the DataServer host. The location of the target database management system (DBMS)—either residing on a separate database host or residing on the same machine as the DataServer host—depends on the DataServer and its platform.

Therefore, you can distribute a Unified Broker instance and its management components among three separate machines on the same network.

Unified Broker common elements

Complete administration for a Unified Broker application potentially involve these components shown in [Figure 7](#):

- The shaded elements which represent individual Unified Broker products
- OpenEdge Management or OpenEdge Explorer
- Command-line utilities

The AdminServer unifies the components previous listed. For some components and administration tasks you can use AdminServer-based management utilities (including the OpenEdge Management or OpenEdge Explorer) or a text editor and configuration validation utilities to accomplish the task.

Note: Similarly, the NameServer resides on its own machine, is installed with an administration framework, including an AdminServer and its own `ubroker.properties` file. If you use a text editor to modify the `ubroker.properties` file, the editor and configuration utilities must reside on the same machine as the Unified Broker or NameServer instance, or have network file system access to the respective Unified Broker and NameServer installation files.

Using default sample brokers

Most Unified Broker products have a default, sample broker that is immediately available for use. The purpose of these brokers is to help you quickly become familiar with and use the functionality associated with these products.

In most instances, the sample brokers require little, if any, modification and no validation. Although you can continue to use these sample brokers when you are operational in either a production or development mode, it is not advisable to do so. Consider modifying these files, using the edit capabilities of such tools as OpenEdge Management or OpenEdge Explorer or mergeprop utility, for the purposes of security and tailoring them to your exact needs. A sample broker, by default, is not connected to a database. If you elect to use a default sample broker, you will need to modify it if you need a database connection.

[Table 49](#) identifies each default, sample brokers associated with each Unified Broker product.

Table 49: Default sample broker for each Unified Broker product

This Unified Broker product ...	Has broker type ...	And also supports the default broker identified as ...
AppServer	AppServer Broker	asbroker1
OpenEdge Adapter for SonicMQ	Adapter for SonicMQ Broker	sonicMQ1
WebSpeed	WebSpeed Transaction Broker	wsbroker1
DataServer for MS SQL Server	DataServer MS SQL Broker	mssbroker1
ORACLE DataServer	ORACLE DataServer Broker	odbbroker1
DataServer for ODBC	DataServer for ODBC Broker	orabroker1
Name Server	NameServer Broker	NS1

For general information about configuring a Unified Broker process, see the [“Configuring and starting Unified Broker instances”](#) section on page 278. For specific details on how to configure the Unified Broker process for a product, how clients specify connections, and how the Unified Broker manages connections with clients, see your product documentation.

Additional Unified Broker characteristics

Each Unified Broker process, default or user defined, manages only one Unified Broker process instance of the same type. The Unified Broker process:

- Can register the following information with a controlling NameServer:
 - The broker’s location on the network
 - The weight factor that you specify for load balancing
 - The Application Services that you specify

Note: Keep in mind that the NameServer is not required. Therefore, the registration of a Unified Broker with the NameServer is dependent on your specific implementation. See [Appendix E, “NameServer and NameServer Load Balancing Details,”](#) for more information about the NameServer and the Unified Broker and NameServer relationship.

- Manages connections between clients and the Unified Broker instance.
- Provides other services unique to a Unified Broker product. For example, it maintains the status of each ABL process running on an AppServer and scales the number of processes according to changing demand.

Configuring and starting Unified Broker instances

You must configure unified broker instances. This section describes:

- The two preliminary tasks you must complete before you can begin configuring and operating a Unified Broker product
- The general steps to configure and start up Unified Broker instances

Prerequisites to configure and use Unified Broker products

There are two preliminary tasks you must complete before you can begin configuring and operating a Unified Broker product:

- **Configure all machines involved in product installation and operation** — This task depends on how you plan to distribute your product and its applications on a network. For more information on configuring OpenEdge products on a network, see [Appendix F, “Configuration Models.”](#)
- **Install the necessary product components** — Typically, this involves installing, on one or more network machines, the OpenEdge Unified Broker product and additional software components that are required to use the product, such as OpenEdge client or Web server software. If you plan to configure fault-tolerant servers or use load balancing, you must install a product that includes load balancing, or install the load-balancing option for your product.

For more information on the OpenEdge product installation, see [Chapter 3, “OpenEdge Installation Prerequisites,”](#) [Chapter 4, “Performing an OpenEdge Installation in Windows,”](#) or [Chapter 5, “Performing an OpenEdge Installation on UNIX or Linux,”](#) and the Windows- or UNIX-specific online help. For more information on distributing resources in a Unified Broker environment, see the [“Working with Unified Brokers”](#) section on page 276.

Once you complete these preliminary tasks, you can configure and start up Unified Broker instances.

How to configure and start up Unified Broker instances

The procedure that follows presents the general steps required to configure and start up Unified Broker instances. Although much of this information has previously been presented in this chapter’s earlier sections, it is helpful to have a general outline of the configuration and startup activities.

The properties file that comes installed with your Unified Broker product includes one sample Unified Broker and NameServer instance for each type of Unified Broker. You can use these as a guide.

To configure and start up Unified Broker instances:

1. Start the AdminServer process on the machine on which each Unified Broker is installed:
 - In Windows, OpenEdge installs the AdminServer as a service that starts automatically at system boot time.
 - On UNIX, you can have the AdminServer started at system startup by editing your boot script to execute the `PROADSV` command.

For information on starting the AdminServer, see the [“Starting the AdminServer”](#) section on page 283.

2. Create and/or modify Unified Broker configurations using any of the following options:
 - **Mergeprop utility** — A command-line utility you can use through the **Proenv** command-line interface to manage the contents of all properties files of which the `ubroker.properties` file pertains to the Unified Brokers discussed in this section. The utility supports functionality similar to OpenEdge Management or OpenEdge Explorer. For more information about the mergeprop utility, see the [“Mergeprop utility overview”](#) section on page 291.
 - **OpenEdge Management or OpenEdge Explorer** — This browser-based user interface can be used in a Windows or UNIX machine to access configurations installed on UNIX or in Windows machines. See the OpenEdge online help for details about using OpenEdge Management or OpenEdge Explorer to configure Unified Broker properties files.
 - **Command-line utilities** — A command-line tool for Windows and UNIX that allows you to control basic activities such as starting, stopping, and querying servers and validating property files associated with OpenEdge products. For more information about the command-line utilities, see [Chapter C, “Command and Utility Reference.”](#)

Note: The properties file that comes installed with your Unified Broker product includes one sample Unified Broker and NameServer instance for each type of Unified Broker that you can use as a guide.

If you plan to configure instances on a UNIX host, you must modify the properties file (`ubroker.properties`) directly on the host for each Unified Broker instance.

Note: To perform most configuration and administrative tasks, use either the mergeprop utility or OpenEdge Management or OpenEdge Explorer because each offers more capabilities than does the command-line utility.

3. Using the OpenEdge Management or OpenEdge Explorer (or the management utility for your Unified Broker product), start up each Unified Broker instance. As it starts, each Unified Broker instance starts additional processes or accesses resources, depending on the product and its configuration.

4. A client can now make a Unified Broker connection request after you verify that it knows:
 - The correct network location of the NameServer to access
 - The Application Service name required to connect to the broker that the client needs

At any time after this step, you can also use any of the appropriate management utilities (mergeprop, OpenEdge Management or OpenEdge Explorer, or command-line) to shut down or query the status of any running Unified Broker instance.

5. When you shut down an AdminServer process at any time and if you have not already shut the Unified Broker instance that it controls, the instance shuts down automatically when you shut down the AdminServer.

During Unified Broker operation, in addition to checking NameServer and Unified Broker status using the OpenEdge Management or OpenEdge Explorer and utilities, you can also review log files being generated by the NameServer and Application Server instance.

The properties file that comes installed with your Unified Broker product includes one sample Unified Broker and NameServer instance for each type of Unified Broker. This can be used as a guide.

Understanding and using the AdminServer

As noted in [Table 48](#), the AdminServer is the central control of OpenEdge Management or OpenEdge Explorer. It facilitates the tasks associated with managing and configuring your installation by ensuring that start and stop requests initiated by OpenEdge products are recognized. In addition to the footnotes identified in [Table 48](#), note the following about the AdminServer:

- To start and stop the AdminServer, you can either enter the `PROADSV` command on the `Proenv` command line or access the **Services** tab by choosing **Control Panel**→**Administrative Tools**→**Services**.
- To manage and configure plug-ins (such as WebSpeed or AppServer) you can use OpenEdge Management or OpenEdge Explorer.
- To minimize the potential for security risks through the AdminServer functionality by ensuring that you do not start the AdminServer as root. Keep in mind that if you do start the AdminServer in this state, all broker processes start as root by default, leaving your entire system vulnerable to security issues.
- The AdminServer has an extensible framework to host the OpenEdge products as plug-ins.

The AdminServer loads the plug-ins and can accept local and remote requests from OpenEdge Management or OpenEdge Explorer, mergeprop utility, and the command-line utilities. However, the actual work is performed within the plug-ins themselves as they provide the specific management functions for a particular product.

Starting the AdminServer

In Windows, the AdminServer starts automatically and runs as a Windows Service (AdminService for OpenEdge). From the desktop, you can perform a variety of administrative tasks. On UNIX, the command-line utility `PROADSV` supports a variety of tasks you can perform in support of the AdminServer.

To start the AdminServer on your machine from the Windows desktop:

- Choose **Start**→**Programs**→**Administrative Tools**→**Services**. Select the **AdminService for OpenEdge 11.3**, and double-click. The **AdminService for OpenEdge 11.3 Properties** dialog box appears. Choose **Start**, then choose **OK**.

Note: If **Administrative Tools** is not available, right-click from the Task Bar. Choose **Properties**, then select the **Advanced** tab. Select the **Display Administrative Tools** check box, then choose **OK**.

If you start the AdminServer, using a specific username and password, that user must have Administrator rights.

For most product installations, the AdminServer is set to Autostart.

- On UNIX, a command-line utility, `PROADSV`, supports OpenEdge administrative capabilities. `PROADSV` allows you to start up, shut down, and query the status of the AdminServer, among other tasks. See [Appendix C, “Command and Utility Reference”](#) for detailed syntax information.

Stopping the AdminServer

You can stop the AdminServer from the desktop or using the command-line utility, `PROADSV`. This section notes methods for the Windows and UNIX.

To stop the AdminServer on your machine from the Windows desktop:

- Choose **Start**→**Programs**→**Administrative Tools**→**Services**. Select the **AdminService for OpenEdge 11.3**, and double-click. The **AdminService for OpenEdge 11.3 Properties** dialog box appears. Choose **Stop**, then choose **OK**.

Note: If **Administrative Tools** is not available, right-click from the Task Bar. Choose **Properties**, then select the **Advanced** tab. Select the **Display Administrative Tools** check box, then choose **OK**.

Performing the task using `PROADSV` on UNIX

On UNIX, a command-line utility, `PROADSV`, supports OpenEdge administrative capabilities. `PROADSV` allows you to start up, shut down, and query the status of the AdminServer, among other tasks. See [Appendix C, “Command and Utility Reference”](#) for detailed syntax information.

Changing the default port

This section discusses ways to change the AdminServer's default port, 20931.

Performing the task from the Windows desktop

In Windows, the AdminServer runs as a service. The AdminServer is configured to start automatically. However, you can choose to change the listening port for the AdminServer as shown in the following code fragment from the AdminServer `rmi` registry:

```
[PluginPolicy.Progress.AdminServer]

pluginclasspath={!{value-of:classpath}
classpath=$DLC/java/...
#In the following code snippet, the port sets the AdminServer rmi registry port
number, the adminport sets the database plugin port, and the jvmargs sets the
database log level to the maximum setting allowed.

port=4321
adminport=7899
```

Note that if you specify values for the `-port` on the command line, these values override values defined in the `%DLC/properties/AdminServerPlugins.properties` file.

Performing the task using PROADSV on UNIX

On UNIX, `PROADSV` is a command-line utility that you can enter on the `Proenv` command line to support OpenEdge administrative capabilities. `PROADSV` allows you to start up, shut down, and query the status of the AdminServer, among other tasks. See [Appendix C, “Command and Utility Reference”](#) for detailed syntax information.

Note that you can also use the code fragment identified in the [“Performing the task from the Windows desktop”](#) section on page 284 for UNIX.

Changing the startup setting

For most product installations, the AdminServer is set to Autostart. You can change this setting to Manual mode.

To change the startup settings from the Windows desktop:

1. In Windows operating systems, choose **Start→ Programs→ Administrative Tools→ Services**. Select the **AdminService for OpenEdge 11.3** and double-click. The **AdminService for OpenEdge 11.3 Properties** dialog box appears. Choose **Manual** in the **Startup type** field, then choose **OK**.

Note: If **Administrative Tools** is not available, right-click from the Task Bar. Choose **Properties**, then select the **Advanced** tab. Select the **Display Administrative Tools** check box, then choose **OK**.

2. Modify the [PluginPolicy.Progress.AdminServer] group of the `$DLC/properties/AdminServerplugins.properties` file to use additional command line startup options.

On UNIX `PROADSV` is a command-line utility that you can enter on the `Proenv` command line to support OpenEdge administrative capabilities. `PROADSV` allows you to start up, shut down, and query the status of the AdminServer, among other tasks. See [Appendix C, “Command and Utility Reference”](#) for detailed syntax information.

Running more than one AdminServer

If you plan to run more than one OpenEdge release, be aware of the following:

- You must run multiple AdminServers. That is, each release requires its own, dedicated AdminServer. For example, if you currently have OpenEdge Release 10.1C installed and in use and are adding OpenEdge Release 11.3, each installation requires its own unique AdminServer.
- You can use default port values for only one of your installed releases. Contention over default values among multiple installations must be avoided. Many of the port parameters will initially contain default values, and require modification. For example, `-port`, `-adminport`, and `agent.properties` file (all of which can be set in the `adminserver.plugins.properties` file, and the `agent.properties` file which are used only if you are using OpenEdge Management) initially contain default values.

It is recommended that you evaluate your port configuration needs **before** running a second, or additional OpenEdge installation in production mode. This pro-active effort helps to ensure that duplicate ports do not conflict in their attempt to use identical default values. See the [“Performing the task from the Windows desktop”](#) section on page 284 for an example of the `adminserver.plugins.properties` file.

Note: The default value available for the `-adminport` is automatically changed for each major OpenEdge release.

Querying the AdminServer

UNIX users generally make more use of the `PROADSV` utility. However, in Windows, you can use the `PROENV` utility with the **Proenv** window, as shown:

Syntax

```
proadsv -query -port 9998
```

`-query`

Displays the AdminServer status.

`-port`

Specifies the listening port for the AdminServer. This is needed if you specified a port other than the default port when you started the AdminServer.

Note: If you specify values for the `-port` or `-adminport` on the command line, these values override values defined in the `%DLC/properties/AdminServerPlugins.properties` file.

Additional AdminServer considerations

The following information is relevant to AdminServer usage:

- Before you start a WebSpeed or AppServer application, you must start the AdminServer.
- The AdminServer User-Group Authorization feature requires that you have privileges set to allow you access and operational privileges for the AdminServer. For details to implement this feature on UNIX systems, see the [“How to implement the User-Group Authorization feature”](#) section on page 250. For details to implement this feature in Windows, see the Windows online help topic “Establishing AdminServer Authorization Options during the Installation.”

For other details about the AdminServer, Windows users should refer to the [“Getting started with the AdminServer”](#) section on page 225 and UNIX users should refer to the [“Getting started with the AdminServer”](#) section on page 249.

AdminServer-related authorization option

In Windows, you can optionally establish AdminServer authorization options for OpenEdge products that support the AdminServer during the installation process. These options are:

- **User Authorization** — Requires each individual user to provide a valid user name and password before the AdminServer can be started.
- **Group Authorization** — Sets up user-defined group names for which operational privileges, at a group level, are required. Group name definitions must conform to specific guidelines.

For information about these options, see the Windows online help topics “Establishing AdminServer Authorization Options during the Installation,” and “User-defined Group Name Conventions and Restrictions.”

AdminServer Logging

There are logging entries that are specifically related to user authentication and authorization.

Using OpenEdge Management or OpenEdge Explorer

OpenEdge Management or OpenEdge Explorer is a browser-based administration tool that provides a simple way to manage OpenEdge servers. It runs both as Windows and UNIX client of the AdminServer, it can also be used with UNIX installations. depending on how your machines are configured.

OpenEdge Servers supported by OpenEdge Management or OpenEdge Explorer


OpenEdge Management or OpenEdge Explorer manages the following OpenEdge servers:

- AppServer
- AppServer Internet Adapter (AIA)
- Database
- DataServer for MS SQL Server
- DataServer for ODBC (Windows only)
- DataServer for ORACLE
- NameServer
- OpenEdge Adapter for SonicMQ
- WebSpeed Adapter
- WebSpeed Messenger
- WebSpeed Transaction Server

OpenEdge Management or OpenEdge Explorer capabilities

Using OpenEdge Management or OpenEdge Explorer, you can:

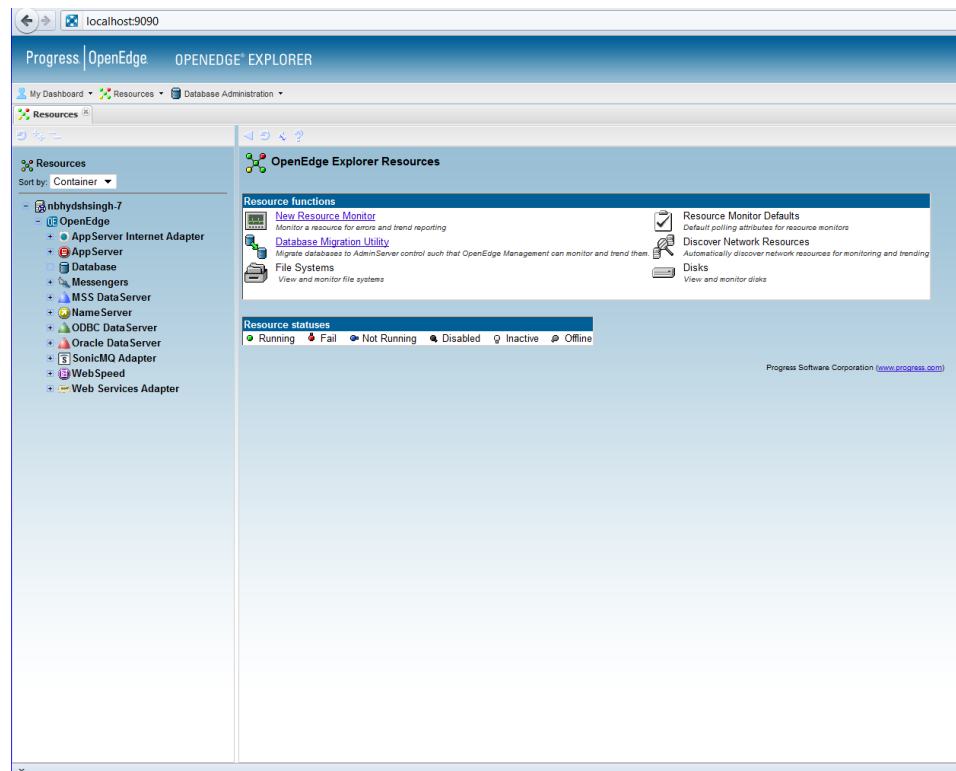
- Create new instances of OpenEdge servers and configure their property settings
- Modify property settings of existing OpenEdge server instances
- Start and stop OpenEdge servers
- Monitor the status of OpenEdge servers
- Remove existing OpenEdge product instances

 To use the OpenEdge Management or OpenEdge Explorer configuration tools, you must first start OpenEdge Management or OpenEdge Explorer and connect to the port where the AdminServer is running. In this situation, the AdminServer must be running on the same host where the OpenEdge products you want to start, stop, or query are installed.

To work with OpenEdge Management or OpenEdge Explorer, you must have started the AdminServer. For complete instructions on using OpenEdge Management or OpenEdge Explorer, see the OpenEdge Management or OpenEdge Explorer online help.

To launch OpenEdge Management or OpenEdge Explorer:

1. From the **Start** menu choose **Progress**→**OpenEdge**→**Management Console**, or from the OpenEdge Program Group, double-click **Management console** icon. A Web page opens, prompting you to provide your login credentials:



2. Enter your username and password to proceed.

Note: The default username and password are **admin** and **admin** respectively. After logging in for the first time, you must change the default login credentials.

3. Follow the instructions in the OpenEdge Management or OpenEdge Explorer online help to establish a connection to an AdminServer.

Saving configurations

OpenEdge Management or OpenEdge Explorer allows you to save:

- NameServer, AppServer, AppServer Internet Adapter, OpenEdge Adapter for SonicMQ, WebSpeed Transaction Server, WebSpeed Messenger, DataServer for MS SQL Server, DataServer for Oracle, and DataServer for ODBC configurations

Configurations you create with any other OpenEdge Management or OpenEdge Explorer configuration tool are saved in the `ubroker.properties` file. You can edit the `ubroker.properties` directly with any text editor. OpenEdge Management or OpenEdge Explorer will automatically update if the `ubroker.properties` is manually edited. When you want to edit the configuration of a NameServer, AppServer, AppServer Internet Adapter, OpenEdge Adapter for SonicMQ, WebSpeed Transaction Server, or WebSpeed Messenger, use only one configuration tool at a time.

- Database configurations

Database configurations you create with the OpenEdge Management or OpenEdge Explorer Database configuration tool are saved in the `conmgr.properties` file. Do not edit the `conmgr.properties` file directly; use OpenEdge Management or OpenEdge Explorer to create and edit database configurations.

- NameServer, AppServer, AppServer Internet Adapter, OpenEdge Adapter for SonicMQ, WebSpeed Transaction Server, WebSpeed Messenger, DataServer for MS SQL Server, DataServer for Oracle, and DataServer for ODBC configurations

Configurations you create with any other OpenEdge Management or OpenEdge Explorer configuration tool are saved in the `ubroker.properties` file. You can edit the `ubroker.properties` directly with any text editor. OpenEdge Management or OpenEdge Explorer will automatically update if the `ubroker.properties` is manually edited. When you want to edit the configuration of a NameServer, AppServer, AppServer Internet Adapter, OpenEdge Adapter for SonicMQ, WebSpeed Transaction Server, or WebSpeed Messenger, use only one configuration tool at a time.

Mergeprop utility overview

The mergeprop utility allows you to manage the content of OpenEdge property files. Property files store configuration information that specifies and controls the behavior of various components.

The mergeprop utility is an alternative command-line or API-based tool, that enables you to update the `comgr.properties`, `ubroker.properties`, and other property files. Using the mergeprop utility provides a consistent means to manage and maintain property files, allowing increased flexibility and ease of maintenance.

When using the mergeprop utility, the user or program specifies an optional *target file* to contain the output of executing mergeprop on the existing property file, and a *delta file* which contains the changes to be made.

Detailed instructions are presented in the [“Using the mergeprop utility”](#) section on page 292.

Operating interfaces

The mergeprop utility has two operating interfaces:

- A command-line interface lets users view and manage properties as necessary.
- A Java API enables programmers to integrate mergeprop functionality with custom applications.

Note: The mergeprop utility is not intended for general user access. You must have Administrator rights to use this utility.

Property value

[Table 50](#) identifies the value types you can use to manage property files with the mergeprop utility. These property files allow you to:

- Use the value of another existing property by reference.
- Use the value of a Java system property by reference.
- Specify a list of any number of syntactically valid values. The list entries are evaluated sequentially, and the first to be successfully resolved is the value of the property.
- Specify a hexadecimal value.

In addition, two special value types can be included in the delta file specified with the mergeprop utility. See the [“Delta file”](#) section on page 294. These value types are valid **only** in the context of a delta file:

- A value formed by appending a specified string to a current value
- A system-generated globally unique identifier (UUID, or universally unique ID)

Using the mergeprop utility

This section provides the syntax and instructions to use the mergeprop utility from a command line.

Command syntax

The following example shows the mergeprop command syntax:

Syntax

```
mergeprop -type file_type
          [-action operation_type [group_name]]
          [-target target_file]
          [-delta delta_file]
          [-validate]
          [-nobackup]
          [-silent]
          [-recurse]
```

[Table 50](#) summarizes the syntax elements used with the mergeprop command. The command line switches can be specified in any order.

Table 50: Command line input to the mergeprop command (1 of 2)

Switch ¹	Arguments	Notes
-type (required)	ubroker database tools plugin none	Each argument (other than none) implies a specific target file in the properties directory (see the “File type” section on page 295).
-action	update (default) create delete list group_name listall group_name	If no action is specified, update is assumed by default. The list and listall actions require an additional argument, the name of the property group to be displayed (for example, ubroker.AS.asbroker1). Do not include the square brackets ([]) that enclose the group name in the ubroker.properties file. On update and create actions, groups listed with no properties in the delta file are ignored.
-target (optional)	Path to the property file to be modified	If you are updating a property file that is in the OpenEdge-install-dir/properties subdirectory, you can omit this option. Only use this option when the property file you plan to update exists in a location other than the OpenEdge-install-dir/properties/subdirectory.

Table 50: Command line input to the mergeprop command (2 of 2)

Switch ¹	Arguments	Notes
-delta	Path to the <code>delta</code> file containing changes to be made	The file containing create, update, or delete changes.
-validate	None	Performs all processing without actually making changes to the target file. This option lets you test for errors.
-nobackup	None	Does not create a backup to the target file before making changes. Unless you invoke this option, mergeprop saves a copy of the original target file in the same directory. The backup copy has a system-generated unique string appended to the name (for example, <code>ubroker.properties 31420040644533</code>).
-silent	None	Suppresses all messages.
-recurse	None	Lists or deletes all groups, server groups, and configurations associated with the specified database. The <code>recurse</code> option is only valid when the file type is specified as <code>database</code> , and the action is identified as either <code>list all</code> or <code>delete</code> .

1. Command switches can occur in any order following the mergeprop command.

Mergeprop parameter details

The following sections provides greater detail about the following parameters:

- [Target file](#)
- [Delta file](#)
- [File type](#)
- [Action switch](#)

Target file

The *target file* is the existing property file in which you are operating. You can add, delete, modify or list properties in the target file. The mergeprop program automatically creates a backup of the original target file, unless you instruct it not to do so. You can also list existing properties without making any changes.

You can explicitly specify a target file, but it is not necessary to do so if you are operating on one of the standard property files listed in [Table 51](#). The file type that you provide as input implies a specific property file, which the program targets by default if no file is specified. These standard property files are located in the `OpenEdge-Install-Directory\properties` directory.

Table 51: Property files managed by the mergeprop utility

Property file	Components configured	Corresponding file type
<code>ubroker.properties</code>	Unified Broker products, including all products managed through OpenEdge Management or OpenEdge Explorer with the exception of the database products	ubroker
<code>conmgr.properties</code>	Database startup parameters	database
<code>JavaTools.properties</code>	Client-side tool configuration, for example OpenEdge Management or OpenEdge Explorer and command line tools	tools
<code>AdminServerPlugins.properties</code>	Plugin products loaded by the AdminServer	plugins

If explicitly specified, the target file is expressed as an argument to the `-target` switch or as a parameter to the `setTargetFile()` or `mergeprop()` method.

Delta file

To make changes with the mergeprop utility, you must list the affected groups and properties in a delta file. The delta file must specify at least one property group; it can also specify one or more properties and associated values. The content of the delta file must conform to the syntax rules for property files, as explained in the [“Logical structure and syntax of property files”](#) section on page 301.

Note: When simply listing (not changing) properties, you do not specify a delta file.

The delta file is expressed as an argument to the `-delta` command or as a parameter to the `setTargetFile()` or `mergeprop()` method.

File type

There are five distinct property file types:

- Ubroker
- Database
- Tools
- Plugin
- None

As indicated in [Table 51](#), one standard property file of each type is found in the *OpenEdge-install-dir\properties* directory.

Specifying the file type enables the mergeprop utility to process delta and target files appropriately. It also makes it unnecessary to explicitly identify the target file, unless you are operating on a copy or test file other than the standard file in the *properties* directory. The program can recognize “none” as a valid type and perform default processing, but you should provide a specific type as input.

The file type is expressed as an argument to the `-type` command switch.

Action switch

Based on the *operation_type* you specify with the `-action` switch, the **mergeprop** utility operates on the target file in one of the following ways:

- **Update** — Creates any new property groups and modifies any existing groups found in the delta file. Properties in the target file are updated to match those in the delta file. This operation is performed by default if you do not explicitly specify an action.
- **Create** — Creates new property groups listed in the delta file, with properties as specified in the delta file. (The delta file must contain only new groups; inclusion of a group that already exists in the target file causes an error and cancels the operation.)
- **Delete** — Removes from the target file any property groups listed in the delta file. The entire groups are deleted; individual properties are not processed. No exception occurs if the delta file contains groups that do not exist in the target file; such groups are simply ignored.

- **List** — Displays (to `stdout`) all properties and values defined specifically for a given group. Inherited properties are not displayed.
- **List all** — Displays (to `stdout`) all properties and values defined for a given group, including inherited properties.

In this context, group refers to a group as listed in the `ubroker.properties` file. For example, `[UBroker.AS.asbroker1]` note that the brackets are not part of the command. For more information about groups, see the [“Logical structure and syntax of property files”](#) section on page 301.

Note: The **List** and **List all** actions are useful for creating a delta file. You can redirect output to a file and use the result as a template for modifying existing instances or creating new ones.

Mergeprop examples

The following examples demonstrate how you can perform various modifications using the mergeprop utility.

Example 1: Updating and adding

The first code fragment shows the contents of the delta file in which a new AppServer Broker instance `addasbroker2`, is defined. The contents of this delta file is based on minor changes made to the sample default broker `asbroker1`, as shown:

```
$ cat addasbroker2
[UBroker.AS.asbroker2]
appserviceNameList=asbroker2
brokerLogFile=@{WorkPath}\asbroker2.broker.log
portNumber=3092
uuid=932.99.999.XX:lee77e:cf3bbe3d33:-8000
```

The following command line adds the new `asbroker2` to the standard, OpenEdge-supplied `ubroker.properties` file:

```
$ mergeprop -type ubroker -action update -delta addasbroker2
```

This same command structure can be used to update a group.

Note: On an add action, you are only required to specify those properties whose values you intend to override. Default values are applied in all other circumstances.

Example 2: Adding a property

This example demonstrates how to add a property specified as an “environment” property to the `asbroker2` created in [Example 1](#).

The following code fragment shows the environment property being added to the `asbroker2` definition in the `ubroker.properties` file:

```
$ cat asbroker2prop
[UBroker.AS.asbroker2]
environment=asbroker2

[Environment.asbroker2]
MYENV=hello

$ mergeprop -type ubroker -action update -delta asbroker2prop
```

Example 3: Deleting a property

It is also helpful to know how to perform a deletion. Remember that you can only perform group-level deletions; you cannot delete a single property within a group. The command line demonstrates how to delete the instance of `asbroker2`:

```
$ mergeprop -type ubroker -action delete ubroker.AS.asbroker2
```

Example 4: Updating properties

You can delete items in a property, by updating the property in a series of steps:

1. Save the property to a temporary file, as shown:

```
$ mergeprop -type ubroker -action list UBroker.AS.asbroker2 >
changeasbroker2
```

2. Delete the property, as shown:

```
$ mergeprop -type ubroker -action delete ubroker.AS.asbroker2
```

3. Edit the temporary file, `changeasbroker2`, to remove the items no longer required.
4. Add the property back to the `ubroker.properties` file, as shown:

```
$ mergeprop -type ubroker -action update -delta changeasbroker2
```

Example 5: Listing properties

The following command line lists the properties defined specifically for the `UBroker.AS.asbroker1` group in `ubroker.properties`, omitting inherited properties:

```
$ mergeprop -type ubroker -action list UBroker.AS.asbroker1
```

The following command line lists all properties of the `UBroker.AS.asbroker1` group, including inherited properties:

```
$ mergeprop -type ubroker -action listall UBroker.AS.asbroker1
```

The following command line lists all properties, including inherited properties of the `FMCONFIGCLI.OSFI` group in the file `installation-path\%DLC%\properties/JavaTools.properties`:

```
$ mergeprop -type tools -action listall FMCONFIGCLI.OSFI
```

The following command line shows how to list a full group definition, specifically a full database group definition. In this example the sports database is referenced and its full group definition which lists all configurations and server groups associated with the sports database is noted:

```
$ mergeprop -type database -action listall sports -recurse
[database.sports]
Autostart=false
Configurations=sports.defaultconfiguration
DatabaseName=/usr1/sports
DefaultConfiguration=sporsts.defaultconfiguration
DisplayName=sports
MonitoredLicense=true
[configuration.sports.defaultconfiguration]
AfterImageProcess=false
AsynchronousPageWriters=1
BeforeImageProcess=true
Database=sports
DisplayName=defaultConfiguration
Monitored=true
OtherArgs=
ServerGroups=sports.default.configuration.defaultservergroup
WatchDogProcess=true
[servergroup.serverGroups=sports.default.configuration.defaultservergroup]
Configuration=sports.defaultconfiguration
DisplayName=defaultServerGroup
Port=4441
Type=both
```

You can update a port specification for the sports database using the following commands:

```
$ cat changeport
[servergroup.sports.defaultconfiguration.defaultservergroup]
port=4444

$mergeprop -type database -action update -delta changeport
```

Java API details

The API for programmatic access to the functionality described in this chapter is defined in the `MergeProperties` class, which resides in the `com.progress.common.property` Java package. The class definition is as follows:

```
package com.progress.common.property;

public class MergeProperties implements MergePropertiesConst
{
    public      MergeProperties();

    public      MergeProperties(int prop_type, String target_filename)
        throws mergeFileException,
               mergePropertyException,
               mergeGroupException,
               mergeException

    public void setBackup(boolean backup_type);
    public void setValidate(boolean validate_type);
    public void setRecurse(boolean recurse_type);
    public void setType(int prop_type);
    public void setAction(int action_type);
    public void setTargetFile(String target_filename);
    public void setDeltaFile(String delta_file);

    public void mergeprop()
        throws mergeFileException,
               mergePropertyException,
               mergeGroupException,
               mergeException

    public void mergeprop(int action_type, String delta_file)
        throws mergeFileException,
               mergePropertyException,
               mergeGroupException,
               mergeException

    public void mergeprop(int prop_type, int action_type, String target_file,
String delta_file, boolean backup)
        throws mergeFileException,
               mergePropertyException,
               mergeGroupException,
               mergeException
```

Constructors

The default `MergeProperties()` constructor creates an object with no values assigned.

Alternatively, you can use the `MergeProperties(prop_type, target_filename)` constructor to load a target file on which multiple actions are to be performed. You can then call the `mergeprop(action_type, delta_file)` method repeatedly as required without reloading the target file.

Methods

Call the `set...()` methods as needed to specify the file type, action, and other input values globally.

To execute operations, call the `mergeprop()` method. The three variations of this method enable you to use any of the following approaches, as appropriate:

- Use all global parameter values as declared by the `set...()` methods
- Directly specify the action and the delta file; this method is useful for executing multiple operations on the same target file
- Directly specify the file type, action, target file, delta file, and backup option

File type and action parameters

Valid values for the `prop_type` and `action_type` parameters are defined in `MergePropertiesConst`.

The `prop_type` parameters are used with the `setType()` and `mergerprop()` methods. See the “[File type](#)” section on page 295 for more information. Valid values are:

- `TYPE_UBROKER`
- `TYPE_DATABASE`
- `TYPE_TOOLS`
- `TYPE_PLUGINS`
- `TYPE_NONE`

The `action_type` parameters are used with the `setAction()` and `mergerprop()` methods. See the “[Action switch](#)” section on page 295 for more information. Valid values are:

- `ACTION_UPDATE`
- `ACTION_CREATE`
- `ACTION_DELETE`
- `ACTION_LIST`
- `ACTION_LISTALL`

Logical structure and syntax of property files

All property files use a hierarchical structure consisting of named groups and subgroups. Each group or subgroup can define a set of properties, for which the values can be either specified or null.

A subgroup inherits all properties from its parent group. By default, it also inherits the values of those properties. Within a subgroup, inherited defaults can be overridden and additional properties can be defined. The lowest level subgroup defines a specific instance of the component type.

Note: Properties are valid only if they are allowed by the appropriate schema file. An attempt to create an unsupported property results in an error.

The syntax in a property file is as follows:

- **Group names** — Names are enclosed in square brackets. A subgroup name consists of the parent group's name followed by a period and the identifier for the subgroup. Thus, the names `[WebSpeed]`, `[WebSpeed.Messengers]`, and `[WebSpeed.Messengers.CGIIP]` form a three-level hierarchy of property groups.
- **Properties and values** — Property name-value pairs are listed immediately following the name of the group for which they are defined. The property name is followed by an “equals” sign and, optionally, a value. For example, `controllingNameServer=` defines a property with a null value; `controllingNameServer=NS1` assigns a specific value to that property.

For example, consider the following groups defined in the as-installed version of `ubroker.properties`:

```
[UBroker]
    controllingNameServer=
    srvrLogEntryTypes=
[UBroker.AS]
    srvrLogEntryTypes=ASPlumbing,DB.Connects
    description=AppServer Broker
[UBroker.AS.asbroker1]
    controllingNameServer=NS1
    description=A sample AppServer setup for State-reset
```

The top-level `[UBroker]` group defines a set of properties that are inherited by the subgroup `[UBroker.AS]` and by all other subgroups.

The subgroup `[UBroker.AS]` defines properties for all AppServer instances. It assigns a default value to the inherited `srvrLogEntryTypes` property, and it defines an additional `description` property.

The subgroup `[UBroker.AS.asbroker1]` defines an AppServer instance. It assigns a value to the `controllingNameServer` property inherited from `[UBroker]`, and it overrides the value of the `description` property inherited from `[UBroker.AS]`.

Property value formats

This section provides a summary of the supported formats for expressing property values. These formats are presented in three categories:

- Newly supported formats (introduced in OpenEdge 10) that are valid in all property files
- Formats that are valid only in delta files used as input to the mergeprop utility
- Formats that were supported prior to OpenEdge 10

[Table 52](#) lists the formats that were introduced in OpenEdge 10 for use in all property files.

Table 52: New value formats supported in all property files

Description	Syntax and example
Reference to another property value.	<code>!{value-of:group.property}</code> Example: <code>jvmargs=!{value-of:Common.jvmargs}</code>
Reference to a Java system property.	<code>!{SystemProperty:java_property}</code> Example: <code>userName=!{SystemProperty:userName}</code>
List of references to be evaluated sequentially. The first reference to be resolved is used. The last entry can be an explicit value. The delimiter between references is a question mark (?), and the list must also be terminated with a question mark.	<code>?value1?value2?value3...value-n?</code> Example: <code>description=?!{SystemProperty:userName}?!{value-of:NameServer.NS1.hostName}?NS1 Host?</code>
Hexadecimal value.	<code>hex_value</code> Example: <code>srvrLoggingLevel=0x0BF</code>

[Table 53](#) lists the formats that were introduced in OpenEdge 10 for use exclusively in delta files used as input to the mergeprop utility.

Table 53: New value formats supported in mergeprop delta files only

Description	Syntax and example
Value formed by appending a specified string to the existing value	<code>!{current-value}<i>append_string</i></code> Example: <code>description=!{current-value}</code> <code>UPDATED</code>
Reference to a Java system property	<code>!{newValue:UUID}</code> Example: <code>uuid=!{newValue:UUID}</code>

[Table 54](#) lists the remaining supported formats, which were introduced prior to the release of OpenEdge 10.

Table 54: Value formats supported prior to OpenEdge 10

Description	Syntax and example
An explicit integer or string constant	<i>value</i> Example: <code>portNumber=3095</code>
Reference to a system environment variable	<code>\${env_variable}</code> Example: <code>workDir=\${WORKDIR}</code>
Reference to a Windows registry value	<code>@{registry_value}</code> Example: <code>workDir=@{WorkPath}</code>

Ubroker.properties file and product configurations

The `ubroker.properties` file stores all the configuration definitions for each instance of the following OpenEdge products:

- OpenEdge NameServer
- AppServer
- AppServer Internet Adapter
- OpenEdge Adapter for SonicMQ
- DataServer for MS SQL Server
- ODBC DataServers (in Windows only)
- ORACLE DataServer
- WebSpeed Transaction Server

The UNIX and Windows `ubroker.properties` files are the same except for platform-specific differences (for example, differences in directory path separators and the differences between environment variable references on UNIX and registry references in Windows).

There is one copy of this file local to each OpenEdge installation. The AdminServer reads and updates the file according to your instructions using the OpenEdge Management or OpenEdge Explorer and management utilities. The `ubroker.properties` file is installed in the `properties` subdirectory of the OpenEdge installation directory (for example, `$DLC/properties/ubroker.properties` on UNIX, or `%DLC%\properties\ubroker.properties` in Windows). In order for the AdminServer to access the properties file, the file must reside in this directory.

Unified Broker products and associated clients

Table 55 identifies each Unified Broker product and indicates the types of clients that can use the Unified Brokers' services.

Table 55: Unified Broker products and the clients they support

Unified Broker product	Client types
AppServer	ABL clients (including other AppServers and WebSpeed instances) and Open Clients
AppServer Internet Adapter	ABL clients (including AppServer and WebSpeed instances)
DataServers	ABL clients (including AppServer and WebSpeed instances)
OpenEdge Adapter for SonicMQ	ABL clients (including AppServer and WebSpeed instances)
WebSpeed	The WebSpeed Messenger, which directs Web client requests to WebSpeed Transaction Servers

Of these clients, you can use the Unified Broker administration framework to manage only WebSpeed Messengers. For specific information on configuring these clients, see your Unified Broker product documentation.

Unified Broker installation prerequisites

Before you install a new Unified Broker version, either to overwrite an existing installation or to add additional OpenEdge components to the current installation, make copies of your `ubroker.properties`, `conmgr.properties`, and `proxygen.preferences` files and place them in another directory. This is necessary because the new installation **automatically** upgrades the files in the `install-path\properties` directory. After you have finished your new installation, replace the newly installed versions of these files with your copies. When you start the AdminServer, your older files will be updated to match the current standards for these files.

When you uninstall an existing Progress or OpenEdge product, the process copies the `ubroker.properties`, `conmgr.properties`, and `proxygen.preferences` files in the `OpenEdge-install-path\properties` directory, to `%TEMP%`. After installing a new OpenEdge Release 11.3 product, you can manually copy and replace the files from `%TEMP%`.

Under certain conditions, you might have to modify this file. You can use the `mergeprop` utility or a text editor to do this.

Note: Each configuration definition contains environment variables, registry entries, and property settings for each product instance. OpenEdge Management or OpenEdge Explorer and the associated command-line configuration utilities use this file to store and validate the configurations for the products.

Ubroker.properties file structure

The `ubroker.properties` file consists of a hierarchical structure of configuration entities, where 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. The AppServer configurations in `ubroker.properties` are shown in the [Table 56](#).

Table 56: Ubroker.properties file structure

Configuration entity name	Configuration entity function
[UBroker]	Defines default property settings for all NameServer, AppServer, DataServer, and WebSpeed Transaction Server brokers.
[UBroker.AS]	Defines default property settings for all instances of an AppServer.
[UBroker.AS.product-instance-name]	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> .

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 the redefinition 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. The `conmgr.properties` file is installed in the `OpenEdge-install-dir\properties\conmgr.properties`.

AdminServer and requirements to set an owner for the broker

The AdminServer honors a user's permissions, according to the user's profile that was used to start an AdminServer. For example, a user who intends to start an AdminServer for another user's process must have the rights to start this second process. These rights or settings must be previously set in the broker's Owner Information properties category. For more information about the Owner Information and the owner feature, see the OpenEdge Management or OpenEdge Explorer online help.

Working with the supported products

For the definitions and usage of all properties supported in the properties file, see the comments in the `ubroker.properties.readme` file that is available at `C:\OpenEdge-install-dir\properties` from the installed `ubroker.properties` file. For more information, see [Table 57](#).

Table 57: Additional sources of information for property files

Product	Documentation
Configuration management and validation	OpenEdge Application Server: Administration and OpenEdge Data Management: Database Administration
<code>ubroker.properties</code> for AppServer installations	OpenEdge Application Server: Administration
<code>ubroker.properties</code> for DataServer installations	OpenEdge DataServer Guide for your specific DataServer product
<code>ubroker.properties</code> for WebSpeed installations	OpenEdge Application Server: Administration
<code>ubroker.properties</code> for NameServer installations	The “Ubroker.properties file and product configurations” section on page 304; also see the specific OpenEdge product manual, referencing the section that includes the NameServer in its configuration
<code>ubroker.properties</code> for AppServer Internet Adapter installations	OpenEdge Application Server: Administration
<code>ubroker.properties</code> for the OpenEdge Adapter for SonicMQ installations	OpenEdge Application Server: Administration

Editing and validating the content of the `ubroker.properties` file

OpenEdge Management or OpenEdge Explorer can be used in Windows, and can connect remotely to UNIX host machines to perform configuration activities. Use the `mergeprop` utility on either platform if OpenEdge Management or OpenEdge Explorer is not available.

To ensure proper run-time processing, the file must be named `ubroker.properties` and must be located in the `properties` subdirectory of the OpenEdge installation directory.

Guidelines for editing the properties file

In general, you should update all configurations (local or remote) using either OpenEdge Management or OpenEdge Explorer or the **mergeprop** utility. If you must update a configuration locally using a text editor:

- Do not directly change the values in the `ubroker.properties` file unless you have a complete understanding of how the changes affect Unified Broker components.
- Make a copy of this file, edit the copy, and verify the result before replacing the original with your edited copy.
- For complete definitions of all the properties and detailed information on how to set them, see the comments included in the properties file.
- Verify the result using the appropriate configuration validation utilities.

To edit the properties file and create or modify a product configuration:

1. Make a backup copy from the installed `ubroker.properties` file and name it, for example, `test.properties`.
2. Make changes to `test.properties` with a text editor.
3. Run the appropriate validation utility on `test.properties` using the `-propfile` option to validate your changes. For a complete list of the command-line utilities you can use to validate property file changes, see [Table 60](#).

Shut down any running brokers or NameServers using the `-stop` option of the appropriate configuration management utilities (for example, `nsman` and `asbman`).

4. Copy `test.properties` to `ubroker.properties` in the `properties` subdirectory of the OpenEdge installation directory. If you might return to the previous configuration, store a backup copy of the `ubroker.properties` file before replacing it with your modified version.
5. Restart your brokers and NameServers using the `-start` option of the appropriate configuration management utilities. For a complete list of the command-line utilities you can use to configure property files, see [Table 59](#).

For more information on editing and validating the `ubroker.properties` file to configure a NameServer, see the [“Editing and validating the content of the `ubroker.properties` file”](#) section on page 307. For more information on editing and validating the file for each Unified Broker product, see your product documentation.

Specifying IP version for underlying Java code

The OpenEdge NameServer and AppServer broker are implemented in Java. You must set Java system properties in the **ubroker.properties** file to properly configure IP communications, as described in [Table 58](#).

Table 58: Java properties for IPv6

Define the property ...	As ...	To configure ...
java.net.preferIPv4Stack	true	The AppServer or NameServer to only use IPv4 sockets. The AppServer or NameServer are not able to communicate with IPv6 clients.
	false	The AppServer or NameServer to communicate with both IPv4 and IPv6 hosts.
java.net.preferIPv6Addresses	true	The default preference of IPv6 addresses over IPv4 addresses if IPv6 is available on the host system. This setting impacts the default hostname resolution for NameServer registration.
	false	The default preference of IPv4 addresses over IPv6 addresses if IPv6 is available on the host system. The AppServer will resolve the default hostname to an IPv4 address, even if an IPv6 address is configured.

You can add the Java system properties to your **ubroker.properties** file by adding the `jvmArgs` property. The `jvmArgs` property is not defined by default. The following example shows the `jvmArgs` property specified for a sample AppServer named `doc`:

```
[Ubroker.AS.doc]
jvmArgs=-Djava.net.preferIPv4Stack=false -Djava.net.preferIPv6Addresses=true
```

Database connection notes

You can further configure your database server communications through AdminServer configuration or broker startup. For example:

- **AdminServer and RDBMS** — Add an additional parameter for the AdminServer to direct the database broker and the database agent ports to only accept connections from the exact type address specified. Edit the file `AdminServerPlugins.properties` to indicate the following information:

```
[PluginPolicy.Progress.AdminServer]
jvmargs=-DforceIPver=<any value>
```

- **Client and RDBMS** — If the underlying implementation of TCP/IP supports V4 mapped addresses, then a broker that opens a V6 connection can accept connection requests from both V4 and V6 clients. If V4 mapped addresses are not supported, and your database needs to accept connections from both V4 and V6 clients, then you must start a broker for each IP version. Client connection attempts must specify the version-specific broker.

Log file updates

Log files are updated to include version information about connections, as follows:

- The unified broker records the following information in the unified broker log file at startup, when the logging level is set to at least 3:

```
[07/09/20@11:35:02.022-0400] P-019544 T-L-3090 3 UB Basic   ipver   : IPv6
```

- The unified broker records the following information in the unified broker log file, when an AppServer is started with an IPv6 connection:

```
[07/09/21@14:29:15.026-0400] P-031358 T-S-0001 2 UB Basic      Started
server: /usr1/stat/progress/101c/dlc/bin/_proapsv -logginglevel 2
-logfile /usr1/stat/progress/101c/wrk/asbroker1.server.log -ubpid 31358
-Ms 1 -logname asbroker1 -logentrytypes ASPlumbing,DB.Connects
-logthreshold 0 -numlogfiles 3 -ASID 1 -ubpropfile
/usr1/stat/progress/101c/dlc/properties/ubroker.properties -svrefresh
-ipver IPv6 -db onekplusdb (8108)
```

- The database broker records the following information in the database log file, when the broker is started with an IPv6 connection:

```
[2007/09/21@16:16:51.496-0400] P-30328      T-0      I BROKER   0: (5644)
Started for 5678 using TCP IPV6 address fd00:19d:807f:1::19, pid 30328.
```

- The database server records the following information in the database log file, when a client connects with an IPv6 connection:

```
[2007/09/21@16:19:29.156-0400] P-30413      T-0      I SRV      1: (742)
Login usernum 24, userid docqa client type ABL , on devlinux01 5 using
TCP/IP IPV6 address ::1.
```

Command-line utilities reference

The command-line management utilities are a set of utilities for Windows and UNIX that allow you to manage existing configurations. Like OpenEdge Management or OpenEdge Explorer, the command-line management utilities run as clients of the OpenEdge AdminServer to manage the NameServer and Unified Broker products.

Using these utilities, you can locally or remotely start, stop, manage, and monitor the status of Unified Broker execution. Unlike OpenEdge Management or OpenEdge Explorer, they do not help you create, remove, or modify properties for Unified Broker configurations.

The framework supports several different product-specific command-line configuration utilities that you can use to manage—that is, start, stop, and query activities—installed OpenEdge server products.

[Table 59](#) identifies the product-specific command-line utilities available.

Table 59: Command-line utilities to start and stop installed OpenEdge products

To start and stop ...	Use this utility ...
A configured OpenEdge Adapter for SonicMQ	ADAPTMAN
A configured AppServer	ASBMAN
The current configuration of an OpenEdge database, or its agent	DBMAN
A configured DataServer for Microsoft SQL Server	MSSMAN
A configured NameServer	NSMAN
The operation of a configured DataServer for ODBC	ODBMAN
The operation of a configured DataServer for ORACLE	ORAMAN
And configure the Web Services Adapter	WSAMAN
The operation of a configured WebSpeed Transaction Server	WTBMAN

OpenEdge supports two approaches to validate property files associated with installed OpenEdge products:

- Mergeprop utility. For more information, see the [“Mergeprop utility overview”](#) section on page 291.
- Through command-line utilities that are available to validate property files associated with installed OpenEdge products. [Table 60](#) identifies these utilities.

Table 60: Command-line utilities to validate property files

To validate property files associated with...	Use this utility...
An existing OpenEdge Adapter for SonicMQ configuration	ADAPTCONFIG
An existing AppServer Internet Adapter configuration	AIACONFIG
An existing configuration for an AppServer	ASCONFIG
An existing configured DataServer for Microsoft MS SQL	MSSCONFIG
An existing configured NameServer	NSCONFIG
An existing configured DataServer for ODBC	ODBCONFIG
An existing configured DataServer for ORACLE	ORACONFIG
The configured Web Services Adapter	WSACONFIG
An existing configured WebSpeed Messenger	WSCONFIG
All databases	DBCONFIG

For more information on starting and managing the NameServer using the OpenEdge Management or OpenEdge Explorer and NSMAN utility, see [Chapter C, “Command and Utility Reference,”](#) and [Chapter E, “NameServer and NameServer Load Balancing Details.”](#)

Starting and Running OpenEdge

This chapter describes how to start up and connect to an OpenEdge database, as detailed in the following sections:

- [Starting OpenEdge in Windows](#)
- [Starting OpenEdge on UNIX platforms](#)
- [Running OpenEdge clients and servers on a network](#)

Starting OpenEdge in Windows

OpenEdge startup commands differ with certain operating systems, user interfaces, and network software. In Windows, you can use the OpenEdge Management or OpenEdge Explorer to start a server. See [Chapter 10, “Configuration,”](#) for more information on launching OpenEdge Management or OpenEdge Explorer. See the OpenEdge Management or OpenEdge Explorer online help for instructions on connecting to an AdminServer.

If you are an international customer, you can set code pages for different application components at startup. You can also set numerical and date/time formats at startup by specifying internationalization parameters. See [OpenEdge Development: Internationalizing Applications](#) and [OpenEdge Data Management: Database Administration](#) for more information on using internationalization parameters at startup.

Startup and shutdown

You can use either the Client or OpenEdge Management or OpenEdge Explorer to perform many of the startup and shutdown tasks. These methods provide you with a GUI interface for managing and configuring databases and servers. If you are not using a Windows environment, or if you prefer a command-line interface, you can choose to enter commands at the command line to perform these tasks. The following sections explain how to use the GUI and command-line interfaces to perform startup and shutdown tasks.

Using the GUI interface

You can use either the Client or OpenEdge Management or OpenEdge Explorer to perform startup and shutdown tasks, indicated in [Table 61](#).

To perform one of the tasks listed using the Client, open the properties of the Client and modify the shortcut target as indicated.

To perform one of the tasks using OpenEdge Management or OpenEdge Explorer, start OpenEdge Management or OpenEdge Explorer, select the server you want to start or stop, and follow the instructions in the online help.

Note: These instructions refer to the ABL Client. To perform SQL tasks, you must start the SQL Explorer and use SQL Client. See SQL Explorer online help for more information about using SQL Client.

[Table 61](#) summarizes tasks and methods to perform startup and shutdown tasks using the graphical user interface (GUI).

Table 61: Windows GUI startup and shutdown commands

Task	OpenEdge program group icon	Action
Start the Procedure Editor and connect to a single-user database	Client	Modify shortcut target properties: <i>install-path\bin\prowin32.exe pathname\db-name -1</i>
Start the Procedure Editor and connect to a multi-user database	Client	Modify shortcut target properties: <i>install-path\bin\prowin32.exe pathname\db-name</i>
Start the ADE Desktop and connect to a single-user database	Client	Modify shortcut target properties: <i>install-path\bin\prowin32.exe -p _desk.p pathname\db-name -1</i>
Start the ADE Desktop and connect to a multi-user database	Client	Modify shortcut target properties: <i>install-path\bin\prowin32.exe -p _desk.p pathname\db-name</i>
Start an OpenEdge batch session and connect to a single-user database	Client	Modify shortcut target properties: <i>install-path\bin\prowin32.exe -b pathname\db-name -1 -p procedure</i>
Start an OpenEdge batch session and connect to a multi-user database	Client	Modify shortcut target properties: <i>install-path\bin\prowin32.exe -b pathname\db-name -p procedure</i>
Start a server or broker for an OpenEdge database	OpenEdge Management or OpenEdge Explorer	See online help. Command-line alternative: <i>proserve pathname\db-name</i>
Shut down a server or broker for an OpenEdge database	OpenEdge Management or OpenEdge Explorer	See online help. Command-line alternative: <i>proshut pathname\db-name</i>

Using the command-line interface

Startup commands start an OpenEdge session and connect you to a database. [Table 62](#) summarizes the startup and shutdown commands for Windows and its functions. For detailed information on these commands and their parameters, see the descriptions of the commands in [OpenEdge Deployment: Startup Command and Parameter Reference](#) and [OpenEdge Data Management: Database Administration](#).

Table 62: Windows startup and shutdown commands

Task	Command
Start a Windows character Procedure Editor and connect to a single-user database	<code>pro db-name</code>
Start a Windows character Procedure Editor and connect to a multi-user database	<code>mpro db-name</code>
Start a Windows character client session in batch mode and connect to a single-user database	<code>bpro db-name -p procedure-name</code>
Start a Windows character client session in batch mode and connect to a multi-user database	<code>mbpro db-name -p procedure-name</code>
Start an OpenEdge server-group	<code>proserve -servergroup server-group-name</code>
Start a server or broker for a multi-user OpenEdge database	<code>proserve db-name -S service-name -H host-name -N network-type</code>
Shut down a multi-user server or broker for an OpenEdge database	<code>proshut db-name</code>
Start a remote OpenEdge DataServer broker	<code>probrkr -S service-name -H host-name -N network-type</code>
Start an asynchronous page writer (APW) for a database ¹	<code>proapw db-name</code>
Start a before-image writer (BIW) ¹	<code>probiw db-name</code>
Start an after-image writer (AIW) ¹	<code>proaiw db-name</code>
Start the OpenEdge Watchdog utility ¹	<code>prowdog db-name</code>
Shut down a remote OpenEdge DataServer	<code>proshut db-name -S service-name -H host-name -N network-type</code>
Shut down an APW, AIW, BIW, or Watchdog process ¹	<code>proshut db-name</code> Choose option 1 (Disconnect a User) to disconnect the process.

1. Option available only on Enterprise product.

Starting OpenEdge as a Windows service

To run OpenEdge as a Windows service, you must start ProService before starting a Progress session. To do this, start the OpenEdge Management or OpenEdge Explorer and connect to an AdminServer.

Note: ProService is run as a Windows service. This means it runs under the system account. It does not run under the account the user is currently logged into. You must grant system access to the directory containing the database for ProService to work properly.

Using the OpenEdge Management or OpenEdge Explorer to connect to the AdminServer

You use the OpenEdge Management or OpenEdge Explorer to connect to the AdminServer.



To connect to the AdminServer using the OpenEdge Management or OpenEdge Explorer:

1. From the **Start** menu choose **Progress**→**OpenEdge**→**Management console**, or from the Progress Program Group, double-click **Management console** icon:



2. A Web page opens, prompting you to provide your login credentials.

Note: The default username and password are **admin** and **admin** respectively. After logging in for the first time, you must change the default login credentials.

3. Establish a connection to an AdminServer. See [Chapter 10, “Configuration,”](#) for more information about OpenEdge Management or OpenEdge Explorer, and see the OpenEdge Management or OpenEdge Explorer online help for detailed instructions on connecting to an AdminServer.

Starting single-user OpenEdge in interactive mode

To start single-user OpenEdge, enter the following command:

```
prowin32 [ db-name ] -1 [ parameters ]
```

db-name

Specifies the database you want to start (-db is implicit but can be specified).

parameters

Specifies the startup parameters you want to use to describe system and application characteristics. For a detailed description of the Progress startup parameters, see [OpenEdge Deployment: Startup Command and Parameter Reference](#), [OpenEdge Data Management: Database Administration](#) and the Progress DataServer Guides ([OpenEdge Data Management: DataServer for Microsoft SQL Server](#), [OpenEdge Data Management: DataServer for ODBC](#), and [OpenEdge Data Management: DataServer for Oracle](#)).

Starting single-user OpenEdge in batch or background mode

Batch or background processing is convenient for large-scale database updates or procedures that you can run unattended (at night, for example).

To start single-user OpenEdge in batch or background mode, enter the following command:

```
prowin32 [ db-name ] -1 -b -p procedure-name [ parameters ]
```

db-name

Specifies the database you want to start

-b

Specifies that OpenEdge should run in batch mode

-p procedure-name

Specifies the procedure to run at startup

parameters

Specifies the startup parameters you want to use

output-file

Specifies the name of the file that receives all output to the default stream

Starting the multi-user server or broker

Before you can run multi-user Progress, you must start the multi-user server process. The server process coordinates all the database requests from all the users using a single database. You can use OpenEdge Management or OpenEdge Explorer to start the multi-user server process, or you can use the command-line interface. The sections that follow describe these methods of starting the multi-user server process.

Using OpenEdge Management or OpenEdge Explorer to start the multi-user server process

You can use OpenEdge Management or OpenEdge Explorer to start a multi-user server process.

Start OpenEdge Management or OpenEdge Explorer and establish a connection to one or more AdminServers. See [Chapter 10, "Configuration,"](#) for more information about starting OpenEdge Management or OpenEdge Explorer, and see the OpenEdge Management or OpenEdge Explorer online help for detailed instructions on using OpenEdge Management or OpenEdge Explorer to start a multi-user server process.

Using the command line interface to start the multi-user server process

Enter the following command at the command line to start the multi-user server process:

```
proserve db-name [ parameters ]
```

db-name

Specifies the database you want to start Progress against (-db is implicit).

parameters

Specifies the startup parameters for the broker/server. For a list of broker/server startup parameters, see [OpenEdge Deployment: Startup Command and Parameter Reference](#) and the Progress DataServer Guides ([OpenEdge Data Management: DataServer for Microsoft SQL Server](#), [OpenEdge Data Management: DataServer for ODBC](#), [OpenEdge Data Management: DataServer for Oracle](#)).

Starting the multi-user server or broker as a Windows service

Before you can run multi-user OpenEdge as a Windows service, you must create an entry in the registry to enable OpenEdge to run as a Windows service. Use OpenEdge Management or OpenEdge Explorer to create an entry in the registry. The following sections describe these methods.

Using OpenEdge Management or OpenEdge Explorer to start the multi-user server or broker

Start OpenEdge Management or OpenEdge Explorer and establish a connection to one or more AdminServers. See [Chapter 10, “Configuration,”](#) for more information about starting OpenEdge Management or OpenEdge Explorer, and see the OpenEdge Management or OpenEdge Explorer online help for detailed instructions on using OpenEdge Management or OpenEdge Explorer to start a multi-user server process.

Starting OpenEdge on UNIX platforms

OpenEdge startup commands differ with certain operating systems, user interfaces, and network software. UNIX provides a series of scripts to run the OpenEdge executables, such as `proserve` to start broker/servers and `mprow` to start multi-user interactive clients. These scripts are tailored for your particular software environment. For information on the script executed by each command, see the description of the command in [Table 63](#).

It is important that you observe the following conventions when you enter a command:

- Use lowercase characters for commands on UNIX
- Enter parameters on UNIX exactly as shown in the syntax descriptions
- Values can be case sensitive on UNIX, for example, names of UNIX files are case sensitive

Startup and shutdown commands

Startup commands start an OpenEdge session and connect you to a database. [Table 63](#) and [Table 64](#) summarize the startup commands for each operating system and their functions. For detailed information on these commands and their parameters, see the descriptions of the commands following the tables. [Table 63](#) describes each of the command components.

Table 63: OpenEdge command components

Component	Description
command	On UNIX, the command runs a script that executes an OpenEdge executable with appropriate parameters
db-name	Name of the database you want to connect to
parameter, qualifier	Operating criteria for the command
value	Numeric value or file specification for the parameter

Table 64 summarizes the tasks you can perform and the related startup and shutdown commands to use on UNIX systems.

Table 64: UNIX startup and shutdown commands

Task	Command
Start a UNIX character Procedure Editor and connect to a single-user database	<code>pro db-name</code>
Start a UNIX character Procedure Editor and connect to a multi-user database	<code>mpro db-name</code>
Start a UNIX character client session in batch mode and connect to a single-user database	<code>bpro db-name</code> <code>-p procedure-name</code>
Start a UNIX OpenEdge character client session in batch mode and connect to a multi-user database	<code>mbpro db-name</code> <code>-p procedure-name</code>
Start an OpenEdge server-group	<code>proserve</code> <code>-servergroup server-group-name</code>
Start a server or broker for a multi-user OpenEdge database	<code>proserve db-name</code> <code>-S service-name</code> <code>-H host-name</code> <code>-N network-type</code>
Shut down a multi-user server or broker for an OpenEdge database	<code>proshut db-name</code>
Start a remote OpenEdge DataServer broker	<code>probrkr -S service-name</code> <code>-H host-name</code> <code>-N network-type</code>
Start an asynchronous page writer (APW) for a database ¹	<code>proapw db-name</code>
Start a before-image writer (BIW) ¹	<code>probiw db-name</code>
Start an after-image writer (AIW) ¹	<code>proaiw db-name</code>
Start the OpenEdge Watchdog utility ¹	<code>prowdog db-name</code>
Shut down a remote OpenEdge DataServer	<code>proshut db-name</code> <code>-S service-name</code> <code>-H host-name</code> <code>-N network-type</code>
Shut down an APW, AIW, BIW, or Watchdog process ¹	<code>proshut db-name</code> Choose option 1 (Disconnect a User) to disconnect the process.

1. Option available only on Enterprise product.

Starting single-user OpenEdge in interactive mode

To start single-user OpenEdge, enter the following command:

```
pro [ db-name ] [ parameters ]
```

db-name

Specifies the database you want to start (-db is implicit but can be specified).

parameters

Specifies the startup parameters you want to use to describe system and application characteristics. For a detailed description of the OpenEdge startup parameters, see [OpenEdge Deployment: Startup Command and Parameter Reference](#) and [OpenEdge Data Management: Database Administration](#).

Starting single-user OpenEdge in batch or background mode

Batch or background processing is convenient for large-scale database updates or procedures that you can run unattended (for example, at night).

To start single-user OpenEdge in batch or background mode, enter the following command:

```
bpro [ db-name ] -p procedure-name  
[ parameters ] > output-file
```

db-name

Specifies the database you want to start.

-p procedure-name

Specifies the procedure to run at startup.

parameters

Specifies the startup parameters you want to use.

output-file

Specifies the name of the file that receives all output to the default stream.

Redirecting Output

On UNIX you can redirect batch job input and output with the greater than (>) and less than (<) redirection symbols. You can also use the pipe symbol (|) to put an OpenEdge batch run in a command pipeline. See the Batch (-b) startup parameter in [OpenEdge Deployment: Startup Command and Parameter Reference](#) for more information.

The example that follows starts in batch or background mode against the sports database and automatically runs the sportsbat startup procedure. In addition, the system directs output (not otherwise directed) with an OUTPUT TO statement to the file named `errlist`, as shown:

```
bpro sports -p sportsbat.p > errlist
```

Starting the multi-user server or broker

Before you can run multi-user OpenEdge, you must start the multi-user server process. The server process coordinates all the database requests from all the users using a single database. Enter the following command to start the multi-user server process:

```
proserve db-name [ parameters ]
```

db-name

Specifies the database you want to start OpenEdge against (-db is implicit).

parameters

Specifies the startup parameters for the broker/server.

The main database server is called the *broker*. The broker process manages shared resources and starts servers for remote users, if necessary. For more information, see the [“Shared-memory configurations”](#) section on page 506.

Running OpenEdge clients and servers on a network

After the database is set up on the network, you are ready to run OpenEdge. The procedure for running clients and servers on a network of systems is similar to the procedure for running them on a single system. First, you must start all required database servers and application servers, and then start the client sessions that connect to them.

Using network startup parameters

To connect network clients, servers, and application servers, you might have to use a variety of startup parameters to establish and manage network communications among them. The requirements and use of these parameters vary on different operating systems and network environments. For more information of using startup parameters, see [OpenEdge Data Management: Database Administration](#).

Client network parameters

[Table 65](#) lists the parameters used to supply OpenEdge clients with necessary network information.

Table 65: Client network parameters

Parameter	Syntax
Host name ¹	-H
Message buffer size	-Mm
Network type	-N
Network version	-Nv
Service name	-S

1. For the TCP network type, this required parameter specifies the machine name (address) where the server runs.

Server network parameters

[Table 66](#) lists the parameters used to supply OpenEdge brokers and servers with necessary network information. In an OpenEdge AppServer configuration, use the same parameters to pass information to AppServer brokers and application servers.

Table 66: Server network parameters (1 of 2)

Parameter	Syntax
Host name	-H
Manual server	-m2
Secondary login broker	-m3
Maximum clients per server	-Ma

Table 66: Server network parameters (2 of 2)

Parameter	Syntax
Minimum clients per server	-Mi
Maximum dynamic server	-maxport
Minimum dynamic server	-minport
Message buffer size	-Mm
AppServer maximum maintained prestart counter	-Mms
Maximum servers	-Mn
Maximum servers per protocol	-Mp
Maximum servers per broker	-Mpb
AppServer maximum prestart counter	-Ms
Network type	-N
Service name	-S

For more information on the syntax and values for each parameter, see the OpenEdge DataServer Guides ([OpenEdge Data Management: DataServer for Microsoft SQL Server](#), [OpenEdge Data Management: DataServer for ODBC](#), and [OpenEdge Data Management: DataServer for Oracle](#)) and [OpenEdge Deployment: Managing ABL Applications](#).

Specifying the network type (-N)

Each OpenEdge executable has a default network type determined by the operating system on which it runs. [Table 67](#) lists the default network type that Progress uses on each supported operating system.

Table 67: Default network types

Operating system (executable)	Default network type
Windows (client or server)	TCP
UNIX (client or server)	TCP

Network addressing (-S and -H)

In all network environments, you use the Service Name (-s) startup parameter to assign a name to an OpenEdge broker/server. You then address this broker/server from a remote client by using the same value for -s as a startup or database connection parameter. Depending on your network type, you might also have to specify additional addressing criteria for remote OpenEdge clients. In terms of OpenEdge addressing, the TCP protocol uses host addressing.

The TCP protocol requires a remote client to explicitly address the database server machine (or host) on which the server runs. In a TCP network, you must use the Host Name (-H) startup parameter to specify the host address. The -H value is the name assigned to the database server machine in your TCP/IP hosts file.

Note: For more information on network addressing, see the [“Support for IPv6”](#) section on page 232.

Starting applications on a network

This section describes the procedures for starting applications on a network.



To start an OpenEdge application on a network:

1. Start each broker or server on its database server machine or application server machine.
2. Start the client applications on the application workstations.

Starting network brokers and servers

You can start most network brokers and servers using either OpenEdge Management or OpenEdge Explorer or the PROSERVE command for your database server machine. To use OpenEdge Management or OpenEdge Explorer, double-click the OpenEdge Management or OpenEdge Explorer icon and follow the directions in the online help for starting brokers and servers. See [Chapter 6, “Administration Utilities,”](#) for more information about starting OpenEdge Management or OpenEdge Explorer.

Alternatively, in Windows and on UNIX systems you can enter the following command to start brokers for two databases (sports and news) using the TCP network type:

```
proserve sports -S spsrv -H localhost -N TCP -db news -S nwsrv
-H localhost -N TCP
```

Starting TCP/IP clients in Windows

You can start most network clients using the MPRO command for your application workstation. You can do this by either modifying the Client properties or by entering a command at the command line.

To modify the Client icon, display the properties and modify the shortcut target. Modify the shortcut target with the following parameters to start a client application named spapp.p for two databases (sports and news) managed on a host named dbmach using the TCP network type:

```
prowin32.exe sports -p spapp.p -S sportssv -H dbmach -N TCP
-db news -S newssv -H dbmach -N TCP
```

To use the command line in Windows, enter the following command to start a client application named `spapp.p` for two databases (sports and news) managed on a host named `dbmach` using the TCP network type:

```
prowin32 sports -p spapp.p -S sportssv -H dbmach -N TCP
      -db news -S newssv -H dbmach -N TCP
```

Starting TCP/IP clients on UNIX

You can start most network clients using the `MPRO` command for your application workstation. For example, on UNIX machines you can enter the following command to start a client application named `spapp.p` for two databases (sports and news) managed on a host named `dbmach`:

```
mpro sports -p spapp.p -S sportssv -H dbmach -db news -S newssv -H dbmach
```

Starting multiple brokers using the same protocol

You can use either OpenEdge Management or OpenEdge Explorer or the command-line interface to start multiple brokers that use the same protocol. The `-Mn` parameter and a new parameter, Maximum Servers per Broker (`-Mpb`), determine the number of servers a broker can start. You can use OpenEdge Management or OpenEdge Explorer to manage and configure server groups.

Using the OpenEdge Management or OpenEdge Explorer to start multiple brokers

You can use OpenEdge Management or OpenEdge Explorer to start multiple brokers that use the same protocol. Start OpenEdge Management or OpenEdge Explorer by double-clicking OpenEdge Management or OpenEdge Explorer Tool icon in the OpenEdge program group. Follow the instructions in the OpenEdge Management or OpenEdge Explorer online help to start and configure brokers. See [Chapter 10, "Configuration,"](#) for more information about starting OpenEdge Management or OpenEdge Explorer.

Using the command-line interface to start multiple brokers

Use the following commands to start two brokers that use TCP and start multiple servers each:

```
proserve db-name -S server-name -N network-type -H host-name -Mn n -Mpb n
proserve db-name -S server-name -N network-type -H host-name -Mpb n -m3
```

`db-name`

Specifies the database you want to start. If the database is not in the current directory, you must specify the full pathname of the database.

`-S service-name`

Specifies the database server or broker process service name. You must specify the service name in a TCP network.

`-N network-type`

Specifies the network protocol, which is TCP.

`-H host-name`

Specifies the machine where the database server runs.

`-Mn n`

Specifies the maximum number of remote client servers and login brokers that the broker process can start.

`-Mpb n`

Specifies the number of servers that the login broker can start to serve remote users. This applies to the login broker that is being started.

`-m3`

Starts the secondary login broker.

To start two brokers that use TCP and start four servers each, use the following commands:

```
proserve db -S demosv1 -N tcp -H myhost -Mn 9 -Mpb 4
proserve db -S demosv2 -N tcp -H myhost -Mpb 4 -m3
```

As the example shows, the `-Mn` value must be large enough to account for each additional broker and all servers. If you do not specify `-Mpb`, the value of `-Mn` becomes the default.

You must include the `-m3` parameter with every secondary broker startup command. While the `-Mpb` sets the number of servers a broker can start, the `-m3` parameter actually starts the secondary broker.

If you start multiple brokers, you should also run the Progress Watchdog process (`PROWDG`). `PROWDG` enables you to restart a dead secondary broker without shutting down the database server.

Accessing a server behind a firewall

OpenEdge allows you to use the Minimum Dynamic Server Port (`-minport`) and the Maximum Dynamic Server Port (`-maxport`) server startup parameters to provide client access to an OpenEdge server that is behind a firewall. This communication is possible only when the access to the server can be limited. You supply this limit when you specify a group of port numbers with the `-minport` and `-maxport` parameters.

For example, suppose you start the following two login brokers:

```
proserve db -S demosv1 -N tcp -H myhost -minport 4000 -maxport 4040
proserve db -S demosv2 -N tcp -H myhost -minport 4041 -maxport 4080 -m3
```

A client requesting a connection from the first broker, `demoSV1`, is assigned a port number in the range of 4000–4040. The 4000–4040 range limits access to the server by limiting communication to just 40 ports.

The default for `-minport` is 1025 for all platforms. Ports lower than 1025 are usually reserved for system TCP and UDP. The default for `-maxport` is 2000 for all platforms. Remember that some operating systems choose transient client ports in the 32768–65535 range. Choosing a port in this range might produce unwanted results.

Starting and running multi-user OpenEdge in interactive mode in Windows

Enter the following command to start and run Progress in interactive mode:

```
prowin32 [ db-name ] [ parameters ]
```

db-name

Specifies the database you want to start. If the database is not in the current directory, you must specify the full pathname of the database.

parameters

Specifies the startup parameters you want to use.

The database you name when starting multi-user Progress must be in the current directory, or you must specify the full pathname of the database. For example, if you are using UNIX and you log in as `sue`, the login directory is `/usr/sue`.

Starting and running multi-user OpenEdge in interactive mode on UNIX

Enter the following command to start and run OpenEdge in interactive mode:

```
mpro [ db-name ] [ parameters ]
```

db-name

Specifies the database you want to start. If the database is not in the current directory, you must specify the full pathname of the database.

parameters

Specifies the startup parameters you want to use.

On UNIX, the `MPRO` command starts either a local or remote client. If the Host Name (`-H`) and Service Name (`-S`) parameters are supplied, OpenEdge starts a remote client—a client that is assigned to a server. Otherwise, OpenEdge starts a local self-service client. Note that specifying `-H` and `-S` when starting a client on the local host machine actually produces a “local remote client” (a local process that accesses the database through a server).

The database you name when starting multi-user OpenEdge must be in the current directory, or you must specify the full pathname of the database. For example, if you are using UNIX and you log in as sue, the login directory is `/usr/sue`.

Starting and running multi-user OpenEdge clients in batch or background mode in Windows

Before you can start a multi-user OpenEdge batch or background job, you must start the server for the database you want to use. You can start the server either by modifying the Client icon properties or by typing a command at the command line.

Using the Client to start multi-user OpenEdge in batch or background mode

You can modify the Client properties to start multi-user OpenEdge in batch or background mode. To modify the Client icon, display the properties and modify the shortcut target with the following parameters:

```
prowin32 [ database-name ] -N network-type  
-S service-name -p procedure-name  
[ parameters ]
```

Using the command-line interface to start multi-user OpenEdge in batch or background mode

To use the command-line interface to start multi-user OpenEdge in batch or background mode, enter the following command:

```
prowin32 [ database-name ] -N network-type  
-S service-name -p procedure-name  
[ parameters ]
```

database-name

Specifies the database you want to start.

-p procedure-name

Specifies the procedure to run at startup.

parameters

Specifies the startup parameters you want to use.

error-file

Specifies the file where error messages are sent.

Using OpenEdge Explorer to start multi-user OpenEdge in batch or background mode

You can use OpenEdge Management or OpenEdge Explorer to create a task that starts multi-user OpenEdge in batch or background mode. See [Chapter 10, "Configuration,"](#) for information on starting the OpenEdge Management or OpenEdge Explorer. See the OpenEdge Management or OpenEdge Explorer online help for instructions on starting multi-user OpenEdge in batch or background mode.

Starting and running multi-user OpenEdge clients in batch or background mode on UNIX

Before you can start a multi-user OpenEdge batch or background job, you must start the server for the database you want to use.

To start multi-user OpenEdge in batch or background mode, enter the following command:

```
mbpro db-name -p procedure-name [ parameters ] > error-file
```

db-name

Specifies the database you want to start.

-p procedure-name

Specifies the procedure to run at startup.

parameters

Specifies the startup parameters you want to use.

error-file

Specifies the file where error messages are sent.

Part 3

OpenEdge Products and Components

[Chapter 12, OpenEdge Installation Products and Components in Windows](#)

[Chapter 13, OpenEdge Installation Products and Components on UNIX](#)

[Appendix A, Preinstallation Checklist for Windows](#)

[Appendix B, Preinstallation Checklist for UNIX](#)

[Appendix C, Command and Utility Reference](#)

[Appendix D, OpenEdge National Language Support](#)

[Appendix E, NameServer and NameServer Load Balancing Details](#)

[Appendix F, Configuration Models](#)

[Appendix G, AdminServer Authorization and Authentication](#)

[Appendix H, Third Party Acknowledgements](#)

OpenEdge Installation Products and Components in Windows

When you install OpenEdge you can choose from two installation options: complete or custom. This chapter provides you with a list of the components and subcomponents that you install for each product when you choose a complete installation, as described in the following sections:

- [OpenEdge installation options](#)
- [OpenEdge product components and subcomponents](#)

OpenEdge installation options

You can choose between two options when installing OpenEdge: complete or custom. These installation options allow you to choose the option that is best suited for you, depending on how many products you are installing, which product components are mandatory, and which are optional, and whether all the products reside on the same system.

Complete installation option

When you choose the Complete installation option and specify the products you want to install, all mandatory, recommended, and optional components and subcomponents are installed automatically. For this reason, a Complete installation usually requires more disk space than a custom installation requires.

Custom installation option

When you choose the custom installation option, all mandatory products and subcomponents are installed, but you can selectively install the recommended and optional components and subcomponents on a product-by-product basis. A Custom installation provides more advanced users, for whom this method is recommended, a means to distribute OpenEdge components on different machines, the ability to select product components to suit their business needs, and allows for working around issues such as disk space limitations.

Caution: Removing recommended product components and/or subcomponents can affect the functionality of a product.

The mandatory, recommended, and optional components and subcomponents for each OpenEdge product are listed, by product, in the [“OpenEdge product components and subcomponents”](#) section on page 337.

For a description of the steps to follow when installing OpenEdge, see the OpenEdge online installation help system. Online help is accessible from all supported platforms in Windows and on UNIX.

OpenEdge product components and subcomponents

The tables in the following sections list the components and subcomponents that are installed for each product.

4GL Development System

[Table 68](#) lists the 4GL Development System components and subcomponents. Choosing the Complete option results in the installation of all components and subcomponents listed.

Table 68: 4GL Development System components and subcomponents (1 of 4)

Component	M/R/O ¹	Subcomponent	M/R/O
4GL Client	M	Base Client—4GL	M
		Crypto Tools	M
		Graphical Client	M
		ICU PSC	M
		Java Server	M
		XML	M
ActiveX Control Support	M	ActiveX Control Development Support	M
		ActiveX Control Runtime Support	M
ADE Source Code	M	ADE Common Source	M
		ADM Source	M
		DB Administration Source	M
		Editor Source	M
		ProTools Source Code	M
Application Debugger	R	Application Debugger	R
		Remote Debugging	M
Character Base Tools	O	ADM Runtime—CHAR	O
		Base ADE	M
		Compile Tool—CHAR	M
		Procedure Editor—CHAR	M
Character Database Admin Tools	O	—	—
Character Image—Dev	O	—	—
Character Runtime Client—Dev	M	—	—

Table 68: 4GL Development System components and subcomponents (2 of 4)

Component	M/R/O ¹	Subcomponent	M/R/O
Client-Side Web Service	R	Client-Side Security	R
		Security Common	M
		Web Services Basic	R
		WSDL Analyzer	R
		Web Services Schema	R
4GL utilities	R	XSD-4GL	R
Common Files	M	Common Files	M
		WebSpeed Common	M
Database Administration Tools	M	4GL Database	M
		Auditing Policy Maintenance	M
		Base ADE	M
		Database Utilities	M
		Graphical Administration	M
Database Server Component	M	4GL Database	M
		4GL Server	M
		Database Server	M
		Database Tools	M
		ICU PSC	M
		SQL Server	M
DataDirect ODBC Driver Support	R	—	—
Graphical Base Tools	M	ADM Runtime—GUI	M
		Base ADE	M
		Compile Tool—GUI	M
		Desktop	M
		Procedure Editor—GUI	M
Name Server	M	—	—
NetSetup	O	—	—
OE Build Utility	R	—	—

Table 68: 4GL Development System components and subcomponents (3 of 4)

Component	M/R/O ¹	Subcomponent	M/R/O
Open Client Adapter Options	R	AppServer Internet Adapter	R
		Common Broker	M
		DotNET Client Support	R
		Java Class Tailoring	M
		Java Client Support	R
		Java Ext	M
		Java Server	M
		OpenEdge Adapter for Sonic MQ	R
		OpenEdge Adapter for Sonic ESB	R
		Proxy Generator	M
		Web Services Adapter Common	M
		Web Services Admin Enable	R
		Web Service Schema	R
Oracle DataServer Client	O	–	–
OpenEdge Management or OpenEdge Explorer Tools	M	Administration Server	M
		Common Broker	M
		Explorer Tools	M
		Java Ext	M
		Java Server	M
		Ubroker Tools	M
		WebSpeed Tools	M
Progress Messages (PROMSGS)	M	Language subset	O
OpenEdge ESQL/C Clients	O	Database Tools	M
		ESQL Client	M
		ICU PSC	M
		SQL Server	M
		SQL Common	M
OpenEdge SQL JDBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL Common	M
		SQL Server	M
		SQL JBDC Client	M

Table 68: 4GL Development System components and subcomponents (4 of 4)

Component	M/R/O ¹	Subcomponent	M/R/O
OpenEdge SQL ODBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL Common	M
		SQL ODBC Client	M
		SQL Server	M
Report Engine	M	—	—
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common	M
Secure Server	M	Perl	M
		Security Common	M
		Server-Side Security	M
SQL Database Server	O	4GL Server	M
		Database Server	M
		Database Tools	M
		Database Utilities	M
		ICU PSC	M
		JDK	M
		Progress Databases	M
		SQL Server	M
Toolkit	M	—	—

1. M=Mandatory, R=Recommended, O=Optional

AppServer Internet Adapter (AIA)

[Table 69](#) lists the AppServer Internet Adapter components and subcomponents. Choosing the Complete option results in the installation of all components and subcomponents listed.

Table 69: AppServer Internet Adapter (AIA) components and subcomponents (1 of 2)

Component	M/R/O ¹	Subcomponent	M/R/O
AppServer Internet Adapter	M	—	—
Common Broker	M	—	—
Common Files (minimum)	M	—	—
Java Server	M	Java Server	M
Name Server	M	—	—

Table 69: AppServer Internet Adapter (AIA) components and subcomponents (2 of 2)

Component	M/R/O ¹	Subcomponent	M/R/O
Progress Messages (PROMSGS)	M	Language subset	O
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common	M
Secure Server	M	Perl	M
		Security Common	M
		Server-Side Security	M

1. M=Mandatory, R=Recommended, O=Optional

Client Networking

[Table 70](#) lists the Client Networking components and subcomponents. Choosing the Complete option results in the installation of all components and subcomponents listed.

Table 70: Client Networking components and subcomponents (1 of 3)

Component	M/R/O ¹	Subcomponent	MR/O
ActiveX Control Support	M	ActiveX Control Runtime Support	M
Character Base Tools—Optional	O	ADM Runtime—CHAR	O
		Base ADE	M
		Compile Tool—CHAR	M
		Procedure Editor—CHAR	M
Character Database Admin Tools	O	—	—
Character Image—Dev	O	—	—
Character Runtime Client—Dev	M	—	—
Client Side Web Services Deploy	R	Client-Side Security	R
		Security Common	M
		Web Services Basic	R
		Web Services Schema	R
Common Files	M	Common Files	M
		WebSpeed Common	M
DataDirect ODBC Driver Support	R	—	—
Database Administration Tools	M	4GL Database	M
		Auditing Policy Maintenance	M
		Base ADE	M
		Database Utilities	M
		Graphical Administrations	M

Table 70: Client Networking components and subcomponents*(2 of 3)*

Component	M/R/O ¹	Subcomponent	MR/O
Graphical Base Tools—Client	M	ADM Runtime—GUI	M
		Base ADE	M
		Compile Tool—GUI	M
		Desktop	M
		Procedure Editor—GUI	M
Name Server	M	—	—
NetSetup	O	—	—
OE Build Utility	R	—	—
Open Client Adapter Options	R	AppServer Internet Adapter	R
		Common Broker	M
		DotNET Client Support	R
		Java Ext	M
		Java Client Support	R
		Java Server	M
		Java Class Tailoring	M
		OE Adapter for Sonic MQ	R
Oracle DataServer Client	O	—	—
OpenEdge ESQL/C Clients	O	Database Tools	M
		ESQL Client	M
		ICU PSC	M
		SQL Server	M
		SQL Common	M
OpenEdge SQL JDBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL JDBC Client	M
		SQL Common	M
		SQL Server	M
OpenEdge SQL ODBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL Common	M
		SQL Server	M
		SQL ODBC Client	M
Progress Messages (PROMSGS)	M	Language subset	O
Report Engine	M	—	—
Remote Debugging	M	—	—

Table 70: Client Networking components and subcomponents (3 of 3)

Component	M/R/O ¹	Subcomponent	MR/O
Runtime Client	M	Base Client—RT	M
		Crypto Tools	M
		Graphical Client	M
		ICU PSC	M
		Java Server	M
		XML	M
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common	M

1. M=Mandatory, R=Recommended, O=Optional

NameServer

[Table 71](#) lists the NameServer components and subcomponents. Choosing the Complete option results in the installation of all components and subcomponents listed.

Table 71: NameServer components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
NameServer	M	—	—
OpenEdge Management or OpenEdge Explorer Tools	M	Administration Server	M
		Common Broker	M
		Explorer Tools	M
		Java Ext	M
		Java Server	M
		Ubroker Tools	M
		WebSpeed Tools	M
Progress Messages (PROMSGS)	M	Language subset	O

1. M=Mandatory, R=Recommended, O=Optional

NameServer Load Balancer

Table 72 lists the NameServer Load Balancer components and subcomponents. Choosing the Complete option results in the installation of all components and subcomponents listed.

Table 72: NameServer Load Balancer components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Common files	M	Common Files	M
		WebSpeed Common	M

1. M=Mandatory, R=Recommended, O=Optional

OpenEdge Adapter for Sonic ESB

Table 73 lists the OpenEdge Adapter for Sonic ESB components and subcomponents. Choosing the Complete option results in the installation of all components and subcomponents listed.

Table 73: OpenEdge Adapter for Sonic ESB components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
Java Class Tailoring	M	—	—
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common	M
OpenEdge Adapter for Sonic ESD	M	—	—

1. M=Mandatory, R=Recommended, O=Optional

OpenEdge Application Server—Basic

Table 74 lists the OpenEdge Application Server Basic components and subcomponents. Choosing the Complete option results in the installation of all components and subcomponents listed.

Table 74: OpenEdge Application Server—Basic components and subcomponents (1 of 2)

Component	M/R/O ¹	Subcomponent	M/R/O
AppServer Runtime Client	M	—	—
Basic Server Option	R	ADE Common Source	M
		ADM Runtime—GUI	R
		ADM Runtime—Char	M
		AppServer—Basic	R
		Auditing Policy Maintenance	M
		Base Client—4GL	M
		Base ADE	M
		Character Client—WebSpeed	R
		Common Broker	M
		Crypto Tools	M
		Editor Source	M
		Graphical Client	M
		ICU PSC	M
		NameServer	R
		Procedure Editor—Char	M
		Progress Databases	M
		SQL Server	M
		Transaction Server—Basic	R
		Web Static	M
		WebSpeed Messenger	R
		WebSpeed Run-time	M
		WebSpeed Tools	M
		XML	M
Client-Side Web Services Deploy	R	Client-Side Security	R
		Security Common	M
		Web Services Basic	R
		Web Services Schema	R
Common Files	M	Common Files	M
		WebSpeed Common	M
OE Build Utilities	R	—	—

Table 74: OpenEdge Application Server—Basic components and subcomponents (2 of 2)

Component	M/R/O ¹	Subcomponent	M/R/O
OE Perl	M	—	—
Progress Messages (PROMSGS)	M	Language	O
Remote Debugging	M	—	—
Open Client Adapter Options—Basic	R	OpenEdge Adapter for Sonic MQ	R
		AppServer Internet Adapter	R
		Common Broker	M
		DotNET Client Support	R
		Java Client Support	R
		Java Ext	M
		Java Class Tailoring	M
		Java Server	M
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common	M
Secure Server	M	Perl	M
		Security Common	M
		Server-Side Security	M
Server Data Source Options	O	DataDirect ODBC Driver Support	O
		Database Tools	M
		ICU PSC	M
		Oracle Client	O
		ESQL Client	M
		SQL Common	M
		SQL JDBC Client	M
		SQL ODBC Client	M
		SQL Server	M
Server Admin and Configuration	M	Administration Server	M
		Common Broker	M
		Explorer Tools	M
		Java Ext	M
		Java Server	M
		Name Server	R
		Ubroker Tools	M
		WebSpeed Tools	M

1. M=Mandatory, R=Recommended, O=Optional

OpenEdge Application Server—Enterprise

Table 75 lists the OpenEdgeApplication Server Enterprise components and subcomponents. Choosing the Complete option results in the installation of all components and subcomponents listed.

Table 75: OpenEdge Application Server—Enterprise components and subcomponents

(1 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
AppServer Runtime Client	M	–	–
Client-Side Web Services Deploy	R	Client-Side Security	R
		Security Common	M
		Web Services Basic	R
		Web Services Schema	R
Common Files	M	Common Files	M
		WebSpeed Common	M
Enterprise Server Options	R	AppServer—Enterprise	R
		ADE Common Source	M
		ADM Runtime—Char	M
		ADM Runtime—GUI	R
		Base Client—4GL	M
		Auditing Policy Maintenance	M
		Base ADE	M
		Character Client—WebSpeed	R
		Client-Side Security	R
		Common Broker	M
		Crypto Tools	M
		Editor Source	M
		Graphical Client	M
		ICU PSC	M
		NameServer	R
		Procedure Editor—Char	M
		Progress Databases	M
		Security Common	M
		SQL Server	M
		Transaction Server—Enterprise	R

Table 75: OpenEdge Application Server—Enterprise components and subcomponents

(2 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
Enterprise Server Options (continued)	R	Web Static	M
		WebSpeed Messenger	R
		WebSpeed Run-time	M
		WebSpeed Tools	M
		XML	M
OE Build Utility	R	—	—
OE Perl	M	—	—
Progress Messages (PROMSGS)	M	Language subset	O
Remote Debugging	M	—	—
Open Client Adapter Options—Enterprise	R	AppServer Internet Adapter	R
		Common Broker	M
		DotNET Client Support	R
		Java Class Tailoring	M
		Java Client Support	R
		Java Ext	M
		Java Server	M
		OpenEdge Adapter for Sonic MQ	R
		OpenEdge Adapter for Sonic ESB	R
		Web Services Adapter Common	M
		Web Services Admin Enable	R
		Web Services Schema	R
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common	M
Secure Server	M	Perl	M
		Security Common	M
		Server-Side Security	M

Table 75: OpenEdge Application Server—Enterprise components and subcomponents*(3 of 3)*

Component	M/R/O ¹	Subcomponent	M/R/O
Server Data Source Options	O	DataDirect ODBC Driver Support	O
		Database Tools	M
		Oracle Client	O
		ESQL Client	M
		ICU PSC	M
		SQL Common	M
		SQL JDBC Client	M
		SQL ODBC Client	M
		SQL Server	M
Server Admin and Configuration	M	Administration Server	M
		Common Broker	M
		Explorer Tools	M
		Java Ext	M
		Java Server	M
		Name Server	R
		Ubroker Tools	M
		WebSpeed Tools	M

1. M=Mandatory, R=Recommended, O=Optional

Progress Developer Studio for OpenEdge

Table 76 lists the Progress Developer Studio for OpenEdge components and subcomponents. When you choose the Complete installation option and install Progress Developer Studio for OpenEdge, all components and subcomponents listed are installed.

Table 76: Progress Developer Studio for OpenEdge components and subcomponents

(1 of 5)

Component	M/R/O ¹	Subcomponent	M/R/O
Application Server options	R	4GL Database	M
		4GL Server	M
		ADM Runtime—GUI	R
		ADM Runtime—CHAR	M
		AppServer—Dev	R
		Base Client—4GL	M
		Character Client—WebSpeed	R
		Common Broker	M
		Crypto Tools	M
		Database Server	M
		Database Tools	M
		ICU PSC	M
		NameServer	R
		Procedure Editor—CHAR	R
		Progress Databases	M
		SQL Server	M
		Transaction Server—Dev	R
		WebSpeed Messenger	R
		Web Static	M
		WebSpeed Run-time	M
		WebSpeed Tools	M
		XML	M
Client-Side Web Services	R	Client-Side Security	R
		Web Services Basic	R
		WSDL Analyzer	R
		Web Services Schema	R
		Security Common	M
4GL utilities	R	XSD-4GL	R

Table 76: Progress Developer Studio for OpenEdge components and subcomponents

(2 of 5)

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
Development Data Source Option	O	4GL Server	M
		DataDirect ODBC Driver Support	O
		Database Utilities	M
		Database Tools	M
		Database Server	M
		ESQL Client	M
		ICU PSC	M
		Oracle Client	O
		JDK	M
		SQL Server	M
		Progress Databases	M
		SQL ODBC Client	M
		SQL JDBC Client	M
		SQL Common	M
Progress Developer Studio for OpenEdge Development	R	4GL Server	M
		ActiveX Control Development	M
		ActiveX Control Runtime	M
		ADM GUI Runtime	R
		ADM Runtime CHAR	M
		Auditing Policy Maintenance	M
		Base Client—4GL	M
		Character Client—RT	O
		Apache Tomcat	M

Table 76: Progress Developer Studio for OpenEdge components and subcomponents

(3 of 5)

Component	M/R/O ¹	Subcomponent	M/R/O
Progress Developer Studio for OpenEdge Development (cont.)	R	Character Image	O
		Crypto Tools	M
		Database Server	M
		Database Tools	M
		Graphical Client	M
		ICU PSC	M
		Progress Developer Studio for OpenEdge	R
		Java Client Support	R
		Java Ext	M
		Java Server	M
		JDK	M
		Procedure Editor—GUI	M
		Progress Databases	M
		Proxy Generator	M
		Remote Debugging	M
		Web Static	M
		WebClient Assembler Utility	R
		WebClient Client	M
		WebSpeed Runtime	M
		XML	M
		Application Debugger	R
		Progress Dynamics	O
		Progress Dynamics RT	O
Progress Developer Studio for OpenEdge AppBuilder	R	Advanced Editing	M
		APPBuilder Core	M
		Base ADE	M
		Compile Tool—CHAR	O
		WebSpeed Workshop-Dev	R
		Compile Tool—GUI	R
NetSetup	O	—	—
OEBuild Utility	R	—	—

Table 76: Progress Developer Studio for OpenEdge components and subcomponents*(4 of 5)*

Component	M/R/O ¹	Subcomponent	M/R/O
Open Client Adapter Options	R	AppServer Internet Adapter	R
		Common Broker	M
		Java Class Tailoring	M
		DotNET Client support	R
		Java Client Support	R
		Java Ext	M
		Java Server	M
		OpenEdge Adapter for Sonic ESB	R
		OpenEdge Adapter for Sonic MQ	R
		Proxy Generator	M
		Web Services Adapter Common	M
		Web Services Admin Enable	R
		Web Services Schema	R
Other Options	O	Base ADE	M
		Client-Side Security	R
		Report Builder Engine	M
		Results (Graphical)	O
		Security Common	M
Progress Messages (PROMSGS)	M	Language subset	O
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common	M
Secure Server	M	Perl	M
		Security Common	M
		Server-Side Security	M

Table 76: Progress Developer Studio for OpenEdge components and subcomponents

(5 of 5)

Component	M/R/O ¹	Subcomponent	M/R/O
Studio Admin and Configuration	R	4GL Database	M
		Administration Server	M
		Auditing Policy Maintenance	M
		Base ADE	M
		Character Administration	R
		Database Utilities	M
		NameServer	R
		Common Broker	M
		UBroker Tools	M
		WebSpeed Tools	M
		NameServer	R
		ToolKit	R
		Administration Server	M
		Explorer Tools	M
		Graphical Administration	M
		Java Ext	M
		Java Server	M

1. M=Mandatory, R=Recommended, O=Optional

OpenEdge DataServer for MS SQL Server

Table 77 lists the OpenEdge DataServer for MS SQL Server components and subcomponents. The DataServer for Microsoft SQL Server is compatible with Microsoft SQL Server 2000 and later. Choosing the Complete option results in the installation of all components and subcomponents listed.

Table 77: OpenEdge DataServer for MS SQL Server components and subcomponents

(1 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
ActiveX Control support	M	ActiveX Control Runtime Support	M
Character Base Tools	O	ADM Runtime—CHAR	O
		Base ADE	M
		Compile Tool—CHAR	M
		Procedure Editor—CHAR	M
Character Database Admin Tools	O	—	—
Character Image—Dev	O	—	—

Table 77: OpenEdge DataServer for MS SQL Server components and subcomponents

(2 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
Character Runtime Client	O	–	–
Common files	M	Common Files	M
		WebSpeed Common	M
Database Administration Tools	M	4GL Database	M
		Auditing Policy Maintenance	M
		Base ADE	M
		Database Utilities	M
		Graphical Administration	M
Graphical Base Tools	M	ADM Runtime—GUI	M
		Base ADE	M
		Compile Tool—GUI	M
		Desktop	M
		Procedure Editor—GUI	M
MS SQL Server DataServer	M	Broker	M
		MS SQL Server DataServer	M
Name Server	M	–	–
NetSetup	O	–	–
OE Build Utility	R	–	–
Open Client Adapter Options Basic	R	AppServer Internet Adapter	R
		Common Broker	M
		DotNET Client Support	R
		Java Client Support	R
		Java Ext	M
		Java Server	M
		Java Class Tailoring	M
		OpenEdge Adapter for SonicMQ	R
OpenEdge Management or OpenEdge Explorer Tools	M	Administration Server	M
		Common Broker	M
		Explorer Tools	M
		Java Ext	M
		Java Server	M
		Ubroker Tools	M
		WebSpeed Tools	M
Progress Messages (PROMSGS)	M	Language subset	O
Remote Debugging	M	–	–

Table 77: OpenEdge DataServer for MS SQL Server components and subcomponents

(3 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
Runtime Client	M	Base Client—RT	M
		Crypto Tools	M
		Graphical Client	M
		ICU PSC	M
		Java Server	M
		XML	M
Schema Holder and Server	M	4GL Server	M
		Database Server	M
		Database Tools	M
		ICU PSC	M
		SQL Server	M

1. M=Mandatory, R=Recommended, O=Optional

OpenEdge DataServer for ODBC

[Table 78](#) lists the OpenEdge DataServer for ODBC components and subcomponents. Choosing the Complete option results in the installation of all components and subcomponents listed.

Table 78: OpenEdge DataServer for ODBC components and subcomponents

(1 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
ActiveX Control Support	M	ActiveX Control Runtime Support	M
Character Base Tools	O	ADM Runtime—CHAR	O
		Base ADE	M
		Compile Tool—CHAR	M
		Procedure Editor—CHAR	M
Character Database Admin Tools	O	—	—
Character Image—Dev	O	—	—
Character Runtime Client—Dev	O	—	—
Common files	M	Common Files	M
		WebSpeed Common	M

Table 78: OpenEdge DataServer for ODBC components and subcomponents (2 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
Database Administration Tools	M	4GL Database	M
		Auditing Policy Maintenance	M
		Base ADE	M
		Database Utilities	M
		Graphical Administration	M
Graphical BaseTools	M	ADM Runtime—GUI	M
		Base ADE	M
		Compile Tools—GUI	M
		Desktop	M
		Procedure Editor—GUI	M
Name Server	M	—	—
NetSetup	O	—	—
ODBC DataServer	M	Broker	M
		ODBC DataServer	M
ODBC DataServer Drivers	M	—	—
OE Build Utility	R	—	—
Open Client Adapter Options Basic	R	AppServer Internet Adapter	R
		Common Broker	M
		DotNET Client Support	R
		Java Client Support	R
		Java Ext	M
		Java Server	M
		Java Class Tailoring	M
		OpenEdge Adapter for SonicMQ	R
OpenEdge Management or OpenEdge Explorer Tools	M	Administration Server	M
		Common Broker	M
		Explorer Tools	M
		Java Ext	M
		Java Server	M
		Ubroker Tools	M
		WebSpeed Tools	M
Progress Messages (PROMSGS)	M	All Languages	O
Remote Debugging	M	—	—

Table 78: OpenEdge DataServer for ODBC components and subcomponents (3 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
Runtime Client	M	Base Client—RT	M
		Crypto Tools	M
		Graphical Client	M
		ICU PSC	M
		Java Server	M
		XML	M
Secure Clients	M	ClientSide Security	R
		Perl	M
		Security Common	M
Schema Holder and Server	M	4GL Server	M
		Database Server	M
		Database Tools	M
		ICU PSC	M
		SQL Server	M

1. M=Mandatory, R=Recommended, O=Optional

OpenEdge DataServer for Oracle

[Table 79](#) lists the OpenEdge DataServer for Oracle components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents listed.

Table 79: OpenEdge DataServer for Oracle components and subcomponents (1 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
ActiveX Control Support	M	ActiveX Control Runtime Support	M
Character BaseTools	O	ADM Runtime—CHAR	O
		Base ADE	M
		Compile Tool—CHAR	M
		Procedure Editor—CHAR	M
Character Database Admin Tools	O	—	—
Character Image—Dev	O	—	—
Character Runtime Client	O	—	—
Common Files	M	Common Files	M
		WebSpeed Common	M

Table 79: OpenEdge DataServer for Oracle components and subcomponents (2 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
Database Administration Tools	M	4GL Database	M
		Auditing Policy Maintenance	M
		Base ADE	M
		Database Utilities	M
		Graphical Administration Tools	M
Graphical Base Tools	M	ADM Runtime—GUI	M
		Base ADM	M
		Compile Tools—GUI	M
		Desktop	M
		Procedure Editor—GUI	M
Name Server	M	—	—
NetSetup	O	—	—
OE Build Utility	R	—	—
Open Client Adapter Options	R	AppServer Internet Adapter	R
		Common Broker	M
		DotNET Client Support	R
		Java Client Support	R
		Java Ext	M
		Java Class Tailoring	M
		Java Server	M
		OpenEdge Adapter for SonicMQ	R
Oracle DataServer	M	Broker	M
		Oracle DataServer	M
Oracle DataServer Client	O	—	—
OpenEdge Management or OpenEdge Explorer Tools	M	Administration Server	M
		Common Broker	M
		Explorer Tools	M
		Java Ext	M
		Java Server	M
		Ubroker Tools	M
		WebSpeed Tools	M
Progress Messages (PROMSGS)	M	Language subset	O
Remote Debugging	M	—	—

Table 79: OpenEdge DataServer for Oracle components and subcomponents (3 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
Runtime Client	M	Base Client—RT	M
		Crypto Tools	M
		Graphical Client	M
		ICU PSC	M
		Java Server	M
		XML	M
Schema Holder and Server	M	4GL Server	M
		Database Server	M
		Database Tools	M
		ICU PSC	M
		SQL Server	M

1. M=Mandatory, R=Recommended, O=Optional

OpenEdge Development Server

[Table 80](#) lists the OpenEdge Development Server components and subcomponents. Choosing the Complete option results in the installation of all components and subcomponents listed.

Table 80: OpenEdge Development Server components and subcomponents (1 of 4)

Component	M/R/O ¹	Subcomponent	M/R/O
Administration and Configuration	M	Administration Server	M
		Auditing Policy Maintenance	M
		Graphical Administration	M
		Base ADE	M
		Character Administration	M
		Common Broker	M
		Database Utilities	M
		4GL Database	M
		Explorer Tools	M
		Java Ext	M
		Java Server	M
		Name Server	R
		Ubroker Tools	M
		WebSpeed Tools	M
Client Side Web Services	R	Client-Side Security	R

Table 80: OpenEdge Development Server components and subcomponents (2 of 4)

Component	M/R/O ¹	Subcomponent	M/R/O
		Security Common	M
		Web Services Basic	R
		WSDL Analyzer	R
		Web Services Schema	R
4GL utilities	R	XSD—4GL	R
Common Files	M	Common Files	M
		WebSpeed Common	M
Development Data Source Options	O	4GL Server	M
		Data Direct ODBC Driver Support	O
		Database Server	M
		Database Tools	M
		Database Utilities	M
		ESQL Client	M
		ICU PSC	M
		JDK	M
		Oracle Client	O
		Progress Databases	M
		SQL Common	M
		SQL JDBC Client	M
		SQL ODBC Client	M
		SQL Server	M
NetSetup	O	—	—
OE Build Utility	R	—	—
OE Perl	M	—	—

Table 80: OpenEdge Development Server components and subcomponents (3 of 4)

Component	M/R/O ¹	Subcomponent	M/R/O
Development Server Options	R	ADM Run-time GUI	R
		ADM Run-time CHAR	M
		AppBuilder Core	M
		AppServer—Dev	R
		4GL Database	M
		4GL Server	M
		Graphical Client	M
		ICU PSC	M
		Name Server	R
		Procedure Editor—Char	M
		Progress Databases	M
		Security Common	M
		Base Client—4GL	M
		Base ADE	M
		Character Client—WebSpeed	R
		Client-Side Security	R
		Common Broker	M
		Crypto Tools	M
		Database Server	M
		Database Tools	M
		DB Administration Source	M
		Desktop	M
		Editor Source	M
		SQL Server	M
		WebSpeed Messenger	R
		Web Static	M
		WebSpeed Run-time	M
		WebSpeed Tools	M
		XML	M
		Transaction Server-Dev	R
		Progress Dynamics RT	R

Table 80: OpenEdge Development Server components and subcomponents (4 of 4)

Component	M/R/O ¹	Subcomponent	M/R/O
Open Client Adapter Options	R	AppServer Internet Adapter	R
		Common Broker	M
		Java Class Tailoring	M
		Java Client Support	R
		Java Ext	M
		Java Server	M
		DotNET Client Support	R
		Proxy Generator	M
		OpenEdge Adapter for SonicMQ	R
		OpenEdge Adapter for Sonic ESB	R
		Web Services Adapter	M
		Web Services Admin Enable	R
Progress Messages (PROMSGS)	M	Language Subset	O
Server Source Code Options	R	ADE Common Source	O
		ProTools Source Code	M
		Editor Source	O
Toolkit	M	–	–
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common	M
Secure Server	M	Perl	M
		Security Common	M
		Server-Side Security	M
Other Development Server Options	R	ADM Runtime—CHAR	O
		Application Debugger	R
		Character Image	O
		Character Client—Runtime	O
		Compile Tool—CHAR	O
		Crypto Tools	M
		Procedure Editor—CHAR	O
		Remote Debugging	M

1. M=Mandatory, R=Recommended, O=Optional

OpenEdge Enterprise RDBMS

Table 81 lists the OpenEdge Enterprise RDBMS components and subcomponents. Choosing the Complete option results in the installation of all components and subcomponents listed.

Table 81: OpenEdge Enterprise RDBMS components and subcomponents (1 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
4GL	O	4GL Server	M
		ActiveX Control Runtime Support	M
		ADM Runtime—CHAR	O
		ADM Runtime—GUI	O
		Auditing Policy Maintenance	M
		Base ADE	M
		Base Client—DA	M
		Character Admin	O
		Character Client—4GL	O
		Character Image	O
		Compile Tool—GUI	O
		Compile Tool—CHAR	O
		Crypto Tools	M
		Desktop	M
		Graphical Administration	M
		Graphical Client	M
		Oracle Client	O
		Procedure Editor—CHAR	O
		Procedure Editor—GUI	M
		Report Engine	M
		SQL Server	M
		ICU PSC	M
		XML	M
Client Side Web Services	R	Client-Side Security	R
		Security Common	M
		Web Services Basic	R
		Web Services Schema	R
Common Files	M	Common Files	M
		WebSpeed Common	M
Failover Clusters	R	Cluster Common	M

Table 81: OpenEdge Enterprise RDBMS components and subcomponents (2 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
NameServer	M	—	—
NetSetup	O	—	—
OE Build Utility	R	—	—
OE Perl	M	—	—
Open Client Adapter Options Basic	R	AppServer Internet Adapter	R
		Common Broker	M
		DotNET Client Support	R
		Java Client Support	R
		Java Ext	M
		Java Server	M
		Java Class Tailoring	M
		OpenEdge Adapter for SonicMQ	R
DB Management	M	4GL Server	M
		Database Server	M
		Database Tools	M
		Database Utilities	M
		ICU PSC	M
		Progress Databases	M
		Legacy 83 Utilities	M
		Legacy 91 Utilities	M
		OE 11.3 DB Utilities	M
		SQL Server	M
OpenEdge Management or OpenEdge Explorer Tools — DB	M	Administration Server	M
		Common Broker	M
		Explorer Tools	M
		Java Ext	M
		Java Server	M
Progress Messages (PROMSGS)	M	All Language subset	O
OpenEdge ESQL/C Clients	O	Database Tools	M
		ESQLClient	M
		ICU PSC	M
		SQLCommon	M
		SQL Server	M

Table 81: OpenEdge Enterprise RDBMS components and subcomponents (3 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
OpenEdge SQL JDBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL Common	M
		SQL Server	M
		SQL JDBC Client	M
OpenEdge SQL ODBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL ODBC Client	M
		SQL Server	M
		SQL Common	M
Remote Debugging	M	—	—
Secure Server	M	Perl	M
		Security Common	M
		Server-Side Security	M
SQL	O	Database Tools	M
		ESQL Client	M
		ICU PSC	M
		SQL Server	M
		SQL Common	M
		SQL JDBC Client	M
		SQL ODBC Client	M

1. M=Mandatory, R=Recommended, O=Optional

OpenEdge Personal RDBMS

Table 82 lists the OpenEdge Personal RDBMS components and subcomponents. Choosing the Complete option results in the installation of all components and subcomponents listed.

Table 82: OpenEdge Personal RDBMS components and subcomponents (1 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
4GL ²	O	4GL Server	M
		ActiveX Control Runtime Support	M
		ADM Runtime—CHAR	O
		ADM Runtime—GUI	O
		Base ADE	M
		Auditing Policy Maintenance	M
		Base Client—DA	M
		Character Administration	O
		Character Client—4GL	O
		Character Image	O
		Compile Tool—CHAR	O
		Compile Tool—GUI	O
		Crypto Tools	M
		Desktop	M
		Graphical Administration	M
		Graphical Client	M
		Oracle Client	O
		Procedure Editor—CHAR	O
		Procedure Editor—GUI	M
		Report Engine	M
		ICU PSC	M
		SQL Server	M
		XML	M
Client-Side Web Services	R	Client-Side Security	R
		Security Common	M
		Web Services Basic	R
		Web Services Schema	R
Common files	M	Common Files	M
		WebSpeed Common	M

Table 82: OpenEdge Personal RDBMS components and subcomponents (2 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
DB Management	M	4GL Server	M
		Database Server	M
		Database Tools	M
		Database Utilities	M
		ICU PSC	M
		Progress Databases	M
		Legacy 83 Utilities	M
		Legacy 91 Utilities	M
		OE 11.3 DB Utilities	M
		SQL Server	M
Name Server	M	—	—
NetSetup	O	—	—
OE Build Utility	R	—	—
OE Perl	M	—	—
Open Client Adapter Options Basic	R	AppServer Internet Adapter	R
		Common Broker	M
		DotNET Client Support	R
		Java Client Support	R
		Java Ext	M
		Java Server	M
		Java Class Tailoring	M
		OpenEdge Adapter for SonicMQ	R
OpenEdge ESQL/C Clients	O	Database Tools	M
		ESQL Client	M
		ICU PSC	M
		SQL Server	M
		SQL Common	M
OpenEdge SQL JDBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL Common	M
		SQL Server	M
		SQL JDBC Client	M

Table 82: OpenEdge Personal RDBMS components and subcomponents (3 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
OpenEdge SQL ODBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL Common	M
		SQL Server	M
		SQL ODBC Client	M
OpenEdge Management or OpenEdge Explorer Tools—DB	M	Administration Server	M
		Common Broker	M
		Explorer Tools	M
		Java Ext	M
		Java Server	M
Progress Messages (PROMSGS)	M	Language subset	O
Remote Debugging	M	—	—
Secure Server	M	Perl	M
		Security Common	M
		Server-Side Security	M
SQL	O	Database Tools	M
		ESQL Client	M
		ICU PSC	M
		SQL Common	M
		SQL JDBC Client	M
		SQL ODBC Client	M
		SQL Server	M

1. M=Mandatory, R=Recommended, O=Optional

2. The 4GL component of the OpenEdge Personal RDBMS includes the Client Networking functionality.

OpenEdge Replication

[Table 83](#) lists the OpenEdge Replication components and subcomponents. Choosing the Complete Installation option results in the installation of all components and subcomponents listed.

Table 83: OpenEdge Replication components and subcomponents (1 of 2)

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
Progress Messages (PROMSGS)	M	Language subset	O

Table 83: OpenEdge Replication components and subcomponents (2 of 2)

Component	M/R/O ¹	Subcomponent	M/R/O
Replication	M	Replication Common	M
		Replication Installation	M
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common	M
Secure Server	M	Perl	M
		Security Common	M
		Server-Side Security	M

1. M=Mandatory, R=Recommended, O=Optional

OpenEdge Replication Plus

[Table 84](#) lists the OpenEdge Replication Plus components and subcomponents. Choosing the Complete Installation option results in the installation of all components and subcomponents listed.

Table 84: OpenEdge Replication Plus components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
Progress Messages (PROMSGS)	M	Language subset	O
Replication	M	Replication Common	M
		Replication Installation	M
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common	M
Secure Server	M	Perl	M
		Security Common	M
		Server-Side Security	M

1. M=Mandatory, R=Recommended, O=Optional

OpenEdge SQL Client Access

Table 85 lists the OpenEdge SQL Client Access components and subcomponents. Choosing the Complete Installation option results in the installation of all components and subcomponents listed.

Table 85: OpenEdge SQL Client Access components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
Name Server	M	—	—
OpenEdge ESQL/C Clients	O	Database Tools	M
		ESQL Client	M
		ICU PSC	M
		SQL Server	M
		SQL Common	M
OpenEdge SQL JDBC clients	O	Database Tools	M
		ICU PSC	M
		SQL Common	M
		SQL Server	M
		SQL JDBC Client	M
OpenEdge SQL ODBC clients	O	Database Tools	M
		ICU PSC	M
		SQL Common	M
		SQL Server	M
		SQL ODBC Client	M
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common	M

1. M=Mandatory, R=Recommended, O=Optional

OpenEdge Studio

[Table 86](#) lists the OpenEdge Studio components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents.

Table 86: OpenEdge Studio components and subcomponents (1 of 5)

Component	M/R/O ¹	Subcomponent	M/R/O
Application Server Options	R	4GL Database	M
		4GL Server	M
		ADM Runtime—GUI	R
		ADM Runtime—CHAR	M
		AppServer—Dev	R
		Base Client—4GL	M
		Character Client—WebSpeed	R
		Common Broker	M
		Crypto Tools	M
		Database Server	M
		Database Tools	M
		Editor Source	M
		ICU PSC	M
		NameServer	R
		Procedure Editor—CHAR	R
		Progress Databases	M
		SQL Server	M
		Transaction Server—Dev	R
		WebSpeed Messenger	R
		Web Static	M
		WebSpeed Run-time	M
		WebSpeed Tools	M
		XML	M
Client-Side Web Services	R	Client-Side Security	R
		Security Common	M
		Web Services Basic	R
		WSDL Analyzer	R
		Web Services Schema	R
4GL utilities	R	XSD-GL	R

Table 86: OpenEdge Studio components and subcomponents (2 of 5)

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
Development Data Source Option	O	4GL Server	M
		DataDirect ODBC Driver Support	O
		Database Utilities	M
		Database Tools	M
		Database Server	M
		ESQL Client	M
		ICU PSC	M
		Oracle Client	O
		JDK	M
		SQL Server	M
		Progress Databases	M
		SQL ODBC Client	M
		SQL JDBC Client	M
		SQL Common	M
Development Source Code Option	R	ADE Common Source	M
		ADM Source	R
		DB Administration Source	O
		Editor Source	O
		ProTools Source Code	M
NetSetup	O	–	–
OE Build Utility	R	–	–

Table 86: OpenEdge Studio components and subcomponents*(3 of 5)*

Component	M/R/O ¹	Subcomponent	M/R/O
Open Client Adapter Options	R	AppServer Internet Adapter	R
		Common Broker	M
		Java Class Tailoring	M
		DotNET Client Support	R
		Java Client Support	R
		Java Ext	M
		Java Server	M
		OpenEdge Adapter for Sonic ESB	R
		OpenEdge Adapter for Sonic MQ	R
		Proxy Generator	M
		Web Services Adapter	M
		Web Services Admin Enable	R
		Web Services Schema	R
Progress Messages (PROMSGS)	M	Language subset	O
Studio Admin and Configuration	R	Administration Server	M
		Character Administration	R
		4GL Database	M
		Auditing Policy Maintenance	M
		Base ADE	M
		Common Broker	M
		Database Utilities	M
		Graphical Administration	M
		Java Ext	M
		Java Server	M
		Explorer Tools	M
		Name Server	R
		Ubroker Tools	M
		Toolkit	R
		WebSpeed Tools	M

Table 86: OpenEdge Studio components and subcomponents*(4 of 5)*

Component	M/R/O ¹	Subcomponent	M/R/O
Studio Development	R	4GL Server	M
		ActiveX Control Development	M
		ActiveX Control Runtime	M
		ADM Runtime—CHAR	M
		ADM Run-time GUI	R
		Advanced Editing	M
		AppBuilder Core	M
		Application Debugger	R
		Auditing Policy Maintenance	M
		Base Client—4GL	M
		Character Client—Runtime	O
		Character Image	O
		Compile Tool—CHAR	O
		Compile Tool—GUI	R
		Crypto Tools	M
		Database Server	M
		Database Tools	M
		DB Admin Source	M
		Desktop	M
		Editor Source	M

Table 86: OpenEdge Studio components and subcomponents (5 of 5)

Component	M/R/O ¹	Subcomponent	M/R/O
Studio Development (cont.)	R	Graphical Client	M
		ICU PSC	M
		Java Ext	M
		Java Server	M
		Base ADE	M
		JDK	M
		Java Client Support	R
		Procedure Editor—CHAR	O
		Procedure Editor—GUI	M
		Progress Dynamics	R
		Progress Dynamics RT	R
		Progress Databases	M
		Proxy Generator	M
		Remote Debugging	M
		Web Static	M
		WebClient Assembler Utility	R
		WebSpeed Runtime	M
		WebClient Client	M
		WebSpeed Workshop—Dev	R
		XML	M
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common	M
Secure Server	M	Perl	M
		Security Common	M
		Server-Side Security	M
Other Options	O	Base ADE	M
		Client-Side Security	R
		Report Builder Engine	M
		Results (Graphical)	O
		Security Common	M

1. M=Mandatory, R=Recommended, O=Optional

OpenEdge Ultra Controls

lists the OpenEdge Ultra Controls components and subcomponents. Choosing the Complete option results in the installation of all components and subcomponents list.

Table 87: OpenEdge Ultra Controls components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
OpenEdge Ultra Controls	M	Infragistics Controls	M

1. M=Mandatory, R=Recommended, O=Optional

OpenEdge Workgroup RDBMS

[Table 88](#) lists the OpenEdge Workgroup RDBMS components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents.

Table 88: OpenEdge Workgroup RDBMS components and subcomponents (1 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
4GL	O	4GL Server	M
		ActiveX Control Runtime	M
		ADM Runtime—CHAR	O
		ADM Runtime—GUI	O
		Auditing Policy Maintenance	M
		Character Image	O
		Compile Tool—CHAR	O
		Compile Tool—GUI	O
		Crypto Tools	M
		Desktop	M
		Graphical Administration	M
		Graphical Client	M
		Oracle Client	O
		Procedure Editor—CHAR	O
		Procedure Editor—GUI	M
		Report Engine	M
		ICU PSC	M
		Base ADE	M
		Base Client—DA	M
		Character Administration	O

Table 88: OpenEdge Workgroup RDBMS components and subcomponents (2 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
4GL (cont.)	O	Character Client—4GL	O
		SQL Server	M
		XML	M
Client-Side Web Services	R	Client-Side Security	R
		Security Common	M
		Web Services Basic	R
		Web Services Schema	R
Common files	M	Common Files	M
		WebSpeed Common	M
Name Server	M	—	—
NetSetup	O	—	—
OE Build Utility	R	—	—
OE Perl	M	—	—
DB Management	M	Database Server	M
		Database Tools	M
		Database Utilities	M
		ICU PSC	M
		4GL Server	M
		Progress Databases	M
		SQL Server	M
		Legacy 83 Utilities	M
		Legacy 91 Utilities	M
		OE 10.1A DB Utilities	M
Open Client Adapter Options Basic	R	AppServer Internet Adapter	R
		Common Broker	M
		DotNET Client Support	R
		Java Client Support	R
		Java Ext	M
		Java Server	M
		Java Class Tailoring	M
		OpenEdge Adapter for SonicMQ	R

Table 88: OpenEdge Workgroup RDBMS components and subcomponents (3 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
OpenEdge Management or OpenEdge Explorer Tools—DB	M	Administration Server	M
		Common Broker	M
		Explorer Tools	M
		Java Ext	M
		Java Server	M
Progress Messages (PROMSGS)	M	Language subset	O
OpenEdge SQL JDBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL Common	M
		SQL JDBC Client	M
OpenEdge SQL ODBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL Common	M
		SQL ODBC Client	M
OpenEdge ESQL/C Clients	O	Database Tools	M
		ICU PSC	M
		SQL Common	M
		ESQL Client	M
Remote Debugging	M	—	—
Secure Server	M	Perl	M
		Security Common	M
		Server-Side Security	M
SQL	O	Database Tools	M
		ICU PSC	M
		SQL Common	M
		ESQL Client	M
		SQL JDBC Client	M
		SQL ODBC Client	M
		SQL Server	M

1. M=Mandatory, R=Recommended, O=Optional

Query/Results

Table 89 lists the Query/Results components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents.

Table 89: Query/Results components and subcomponents

(1 of 2)

Component	M/R/O ¹	Subcomponent	M/R/O
ActiveX Control Support	M	ActiveX Control Runtime Support	M
Character Base Tools	O	ADM Runtime—CHAR	O
		Base ADE	M
		Compile Tool—CHAR	M
		Procedure Editor—CHAR	M
Character Database Admin Tools	O	—	—
Character Image—Dev	O	—	—
Character Runtime Client—Dev	M	—	—
Client-Side Web Services Deploy	R	Client-Side Security	R
		Security Common	M
		Web Services Basic	R
		Web Services Schema	R
Common Files	M	Common Files	M
		WebSpeed Common	M
Database Administration Tools	O	4GL Database	M
		Auditing Policy Maintenance	M
		Base ADE	M
		Database Utilities	M
		Graphical Administration Tools	M
Graphical Base Tools	M	ADM Runtime—GUI	M
		Base ADE	M
		Compile Tool—GUI	M
		Desktop	M
		Procedure Editor—GUI	M
NetSetup	O	—	—
OE Build Utility	R	—	—

Table 89: Query/Results components and subcomponents (2 of 2)

Component	M/R/O ¹	Subcomponent	M/R/O
Open Client Adapter Options Basic	R	AppServer Internet Adapter	R
		Common Broker	M
		DotNET Client Support	R
		Java Client Support	R
		Java Ext	M
		Java Server	M
		Java Class Tailoring	M
		OpenEdge Adapter for SonicMQ	R
Oracle DataServer Client	O	—	—
Progress Messages (PROMSGS)	M	Language subset	O
Query Client	M	Base Client—Query	M
		Crypto Tools	M
		Graphical Client	M
		ICU PSC	M
		Java Server	M
		XML	M
Remote Debugging	M	—	—
Report Engine	M	—	—
Results	M	Base ADE	M
		Results (Graphical)	M
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common	M

1. M=Mandatory, R=Recommended, O=Optional

Mobile REST Management Agent

Table 91 lists the Mobile REST Management Agent components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents listed.

Table 90: Mobile REST Management Agent components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
REST Management Agent	M	REST Management Common	M
		Web Services Adapter Install	

1. M=Mandatory, R=Recommended, O=Optional

Translation Manager

Table 91 lists the Translation Manager components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents listed.

Table 91: Translation Manager components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
ActiveX Control support	M	ActiveX Control Runtime	M
Common Files	M	Common Files	M
		WebSpeed Common	M
NetSetup	O	—	—
Translation Manager	M	Base ADE	M
		Translation Manager	M
Visual Translator	M	Translation Manager	M

1. M=Mandatory, R=Recommended, O=Optional

Visual Translator

Table 92 lists the Visual Translator components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents.

Table 92: Visual Translator components and subcomponents

(1 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
4GL Client	M	Base Client—4GL	M
		Crypto Tools	M
		Graphical Client	M
		ICU PSC	M
		Java Server	M
		XML	M
ActiveX Control Support	M	ActiveX Control Runtime Support	M

Table 92: Visual Translator components and subcomponents*(2 of 3)*

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
Database Server Component	M	4GL Database	M
		4GL Server	M
		Database Server	M
		Database Tools	M
		ICU PSC	M
		SQL Server	M
Graphical Base Tools	M	ADM Runtime—GUI	M
		Base ADE	M
		Compile Tool—GUI	M
		Desktop	M
		Procedure Editor—GUI	M
NetSetup	O	—	—
Open Client Adapter Options Basic	R	AppServer Internet Adapter	R
		Common Broker	M
		DotNET Client Support	R
		Java Client Support	R
		Java Ext	M
		Java Server	M
		Java Class Tailoring	M
		OpenEdge Adapter for SonicMQ	R
Progress Messages (PROMSGS)	M	Language subset	O
Remote Debugging	M	—	—
Report Engine	M	—	—
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common	M

Table 92: Visual Translator components and subcomponents (3 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
SQL Database Server	O	4GL Server	M
		Database Server	M
		Database Tools	M
		Database Utilities	M
		ICU PSC	M
		JDK	M
		Progress Databases	M
		SQL Server	M
Visual Translator	M	–	–

1. M=Mandatory, R=Recommended, O=Optional

Web Services Adapter

[Table 93](#) lists the Web Services Adapter (WSA) components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents.

Table 93: Web Services Adapter components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Name Server	M	–	–
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common	M
Web Services Adapter	M	Web Services Adapter	M
		Web Services Admin	R

1. M=Mandatory, R=Recommended, O=Optional

WebSpeed Messenger

Table 94 lists the WebSpeed Messenger components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents.

Table 94: WebSpeed Messenger components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Messenger component	M	Java Server	M
		Messenger Runtime Support	M
		WebSpeed Tools	M
Progress Messages (PROMSGS)	M	Language subset	O
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common	M
Web static	M	—	—
WebSpeed Common	M	—	—

1. M=Mandatory, R=Recommended, O=Optional

WebSpeed Workshop

Table 95 lists the WebSpeed Workshop components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents.

Table 95: WebSpeed Workshop components and subcomponents (1 of 4)

Component	M/R/O ¹	Subcomponent	M/R/O
Administration and Configuration	M	Administration Server	M
		Auditing Policy Maintenance	M
		Base ADE	M
		Character Administration	M
		Common Broker	M
		Database Utilities	M
		Explorer Tools	M
		4GL Database	M
		Graphical Administration	M
		Java Ext	M
		Java Server	M
		Name Server	R
		Ubroker Tools	M
		WebSpeed Tools	M
Client-Side Web Services	R	Client-Side Security	R
		Security Common	M
		Web Services Basic	R
		WSDL Analyzer	R
		Web Services Schema	R
4GL utilities	R	XSD-4GL	R
Common files	M	Common Files	M
		WebSpeed Common	M
Development Data Source	O	4GL Server	M
		Database Server	M
		Database Tools	M
		Database Utilities	M
		DataDirect ODBC Driver Support	O
		ESQL Client	M
		ICU PSC	M
		JDK	M
		Progress Databases	M

Table 95: WebSpeed Workshop components and subcomponents*(2 of 4)*

Component	M/R/O ¹	Subcomponent	M/R/O
Development Data Source (cont.)	O	Oracle Client	O
		SQL Common	M
		SQL JDBC Client	M
		SQL ODBC Client	M
		SQL Server	M
Development Server Options	R	ADM Runtime—GUI	R
		ADM Runtime—CHAR	M
		Advanced Editing	M
		APPBuilder Core	M
		AppServer—Dev	R
		Base ADE	M
		Base Client—4GL	M
		Character Client—WebSpeed	R
		Client-Side Security	R
		Common Broker	M
		Compile Tool GUI	R
		Crypto Tools	M
		Database Server	M
		Database Tools	M
		DB Administration Source	M
		Desktop	M
		Editor Source	M
		4GL Database	M
		4GL Server	M
		Graphical Client	M
		ICU PSC	M
		NameServer	R
		Procedure Editor—CHAR	M
		Procedure Editor—GUI	M
		Progress Databases	M
		Security Common	M
		SQL Server	M
		Transaction Server—Dev	R
		WebSpeed Messenger	R
		WebSpeed Runtime	M

Table 95: WebSpeed Workshop components and subcomponents (3 of 4)

Component	M/R/O ¹	Subcomponent	M/R/O
Development Server Options (cont.)	R	WebSpeed Tools	M
		Web Static	M
		Workshop	M
		XML	M
		Progress Dynamics RT	R
Development Source Code Options	R	ADE Common Source	M
		ADM Source Code	O
		DB Administration Source	O
		Editor Source	O
		ProTools Source Code	M
NetSetup	O	–	–
OE Build Utility	R	–	–
Open Client Adapters Options	R	AppServer Internet Adapter	R
		Common Broker	M
		Java Class Tailoring	M
		Java Client Support	R
		Java Ext	M
		Java Server	M
		DotNET Client Support	R
		OpenEdge Adapter for Sonic ESB	R
		OpenEdge Adapter for SonicMQ	R
		Proxy Generator	M
		Web Services Adapter	M
		Web Services Admin Enable	R
		Web Services Schema	R
Progress Messages (PROMSGS)	M	Language subset	O
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common	M
Secure Server	M	Perl	M
		Security Common	M
		Server-Side Security	M
Toolkit	M	–	–

Table 95: WebSpeed Workshop components and subcomponents*(4 of 4)*

Component	M/R/O ¹	Subcomponent	M/R/O
WebSpeed Development	R	ActiveX Control Development	M
		4GL Database	M
		ActiveX Control Runtime Support	M
		ADM Runtime CHAR	M
		ADM Runtime GUI	R
		Application Debugger	R
		Advanced Editing	M
		AppBuilder Core	M
		Base ADE	M
		Base Client—4GL	M
		Character Client—4GL	O
		Compile Tool GUI	R
		Crypto Tools	M
		Progress Databases	M
		Database Server	M
		Database Tools	M
		DB Administration Source	M
		Desktop	M
		Editor Source	M
		Graphical Client	M
		ICU PSC	M
		Web Static	M
		4GL Server	M
		Remote Debugging	M
		Procedure Editor—CHAR	O
		SQL Server	M
		WebSpeed Runtime	M
		WebSpeed Workshop Dev	R
		XML	M
		Procedure Editor—GUI	M

1. M=Mandatory, R=Recommended, O=Optional

OpenEdge Management SE

Table 96 lists the OpenEdge Management SE components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents listed.

Table 96: OpenEdge Management SE components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
Fathom	M	Administration Server	M
		Client-Side Security	R
		Fathom Cmn	M
		Fathom Common	M
		Fathom Doc	M
		Fathom Tailor	M
		Fathom Class Tailoring	M
Progress Messages (PROMSGS)	M	Language subset	O

1. M=Mandatory;R=Recommended;O=Optional

OpenEdge Business Process Modeler

Table 97 lists the OpenEdge Business Process Modeler components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents listed.

Table 97: OpenEdge Business Process Modeler components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
BP Modeler	M	BP Modeler Runtime	M
		JDK	M

1. M=Mandatory;R=Recommended;O=Optional

OpenEdge Business Process Server

Table 98 lists the OpenEdge Business Process Server components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents listed.

Table 98: OpenEdge Business Process Server components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
Business Process Server	M	Business Process Server Runtime	M
		JDK	M
		JBOSS Community Edition	M
Fathom	M	Administration Server	M
		Client-Side Security	R
		Fathom Cmn	M
		Fathom Common	M
		Fathom Tailor	M
		Java Class Tailoring	M
		Perl	M
OpenEdge SQL JDBC Client	O	Database Tools	M
		ICU PSC	M
		SQL Common	M
		SQL JDBC Client	M
		SQL Server	M

1. M=Mandatory;R=Recommended;O=Optional

SNMP Adapter

[Table 99](#) lists the SNMP Adapter components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents listed.

Table 99: SNMP Adapter components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
SNMP Adapter	M	SNMP Common	M

1. M=Mandatory;R=Recommended;O=Optional

OpenEdge TDE

[Table 100](#) lists the OpenEdge TDE components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents listed.

Table 100: OpenEdge TDE components and subcomponents

Components	M/R/O ¹	Subcomponents	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M

1. M=Mandatory;R=Recommended;O=Optional.

OpenEdge Multi-tenant Table

[Table 101](#) lists the OpenEdge Multi-tenant Table components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents listed.

Table 101: OpenEdge Multi-tenant Table components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M

1. M=Mandatory;R=Recommended;O=Optional\

OpenEdge Installation Products and Components on UNIX

This chapter presents a comprehensive list of the components and subcomponents that comprise each OpenEdge product. The information attempts to be a reference for all UNIX/Linux platforms on which you can install. However, depending on the specific UNIX/Linux platform, there can be some minor differences. In a few instances, some platform variations can affect component and subcomponent availability as noted.

Refer to the product details in this chapter when planning and performing either a Complete or Custom installation.

The topics in this chapter are described in the following sections:

- [OpenEdge installation options](#)
- [OpenEdge product components and subcomponents](#)

Note: With the exception of Solaris SPARC 32- and 64-bit platforms, all supported UNIX/Linux platforms require at least the entry level JDK and JRE versions installed to run OpenEdge products. For more information about this Java prerequisite, see the [“Requirements for using Java”](#) section on page 54.

OpenEdge installation options

You can choose between two options when installing OpenEdge: complete or custom. These two options allow you to choose the option that is best suited for you, depending on how many products you are installing, which product components are mandatory and which are optional, and whether all the products reside on the same system.

Complete installation option

When you choose the Complete installation option and specify the products you want to install, all mandatory, recommended, and optional components and subcomponents are installed automatically. For this reason, a Complete installation usually requires more disk space than a custom installation requires.

Custom installation option

When you choose the Custom installation option, all mandatory products and subcomponents are installed, but you can selectively install the recommended and optional components and subcomponents on a product-by-product basis. A Custom installation provides more advanced users, for whom this method is recommended, a means to distribute OpenEdge components on different machines, the ability to select product components to suit their business needs, and allows for working around issues such as disk space limitations.

Caution: Removing recommended product components and/or subcomponents can affect the functionality of a product.

The mandatory, recommended, and optional components and subcomponents for each OpenEdge product are listed, by product, in the tables found in the [“OpenEdge product components and subcomponents”](#) section on page 397.

OpenEdge product components and subcomponents

The tables in the following sections list the components and subcomponents that are installed for each product.

4GL Development System

[Table 102](#) lists the 4GL Development System components and subcomponents. Choosing the Complete installation results in the installation of all components and subcomponents listed.

Table 102: 4GL Development System components and subcomponents (1 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
4GL Client	M	Base Client—GL	M
		Character Client	M
		Crypto Tools	M
		ICU PSC	M
		Java Server	M
		XML	M
4GL utilities	R	XSD—4GL	R
ADE Source Code	R	ADE Common Source	M
		ADM Source	M
		DB Administration Source	M
		Editor Source	M
Application Debugger	R	Application Debugger	R
		Remote Debugging	M
Character Base Tools	M	ADM Runtime—CHAR	M
		Base ADE	M
		Compile Tool—CHAR	M
		Procedure Editor—CHAR	M
Client-Side Web Services	R	Client-Side security	R
		Security Common ²	M
		Web Services Basic	R
		WSDL Analyzer	R
		Web Services Schema	R
Common Files	M	Common Files	M
		WebSpeed Common	M

Table 102: 4GL Development System components and subcomponents*(2 of 3)*

Component	M/R/O ¹	Subcomponent	M/R/O
Database Administration Tools	M	4GL Database	M
		Audit Policy Maintenance	M
		Base ADE	M
		Character Administration	M
		Database Utilities	M
Database Server Component	M	4GL Database	M
		4GL Server	M
		Database Server	M
		Database Tools	M
		ICU PSC	M
		SQL Server	M
Name Server	M	—	—
OE Build Utility	R	—	—
Open Client Adapter Options	R	AppServer Internet Adapter	R
		Common Broker	M
		Java Class Tailoring	M
		Java Client Support	R
		Java Ext	M
		Java Server	M
		OpenEdge Adapter for SonicMQ	R
		OpenEdge Adapter for Sonic ESB	R
		Proxy Generator	M
		Web Services Adapter Common	M
		Web Services Admin Enabler	R
		Web Services Schema	R
OpenEdge ESQL/C Clients	O	Database Tools	M
		ESQL Client	M
		ICU PSC	M
		SQL Common	M
		SQL Server	M
OpenEdge SQL JDBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL Common	M
		SQL JDBC Client	M
		SQL Server	M

Table 102: 4GL Development System components and subcomponents (3 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
OpenEdge SQL ODBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL Common	M
		SQL ODBC Client	M
		SQL Server	M
Oracle DataServer Client	O	—	—
OpenEdge Management or OpenEdge Explorer Tools	M	Administration Server	M
		Common Broker	M
		Java Ext	M
		Java Server	M
		Ubroker Tools	M
Progress Messages (PROMSGS)	M	Language subset	O
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common ²	M
Secure Server	M	Perl	M
		Security Common ²	M
		Server-Side Security	M
SQL Database Server	O	4GL Server	M
		Database Server	M
		Database Tools	M
		Database Utilities	M
		ICU PSC	M
		JDK	M
		Progress Databases	M
		SQL Server	M
Toolkit	M	—	—

1. M=Mandatory, R=Recommended, O=Optional

2. The Security Common subcomponent is not supported on a Linux PowerPC platform.

AppServer Internet Adapter (AIA)

Table 103 lists the AppServer Internet Adapter components and subcomponents. Choosing the Complete installation results in the installation of all components and subcomponents listed.

Table 103: AppServer Internet Adapter components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
AppServer Internet Adapter	M	—	—
Common Broker	M	—	—
Common Files (minimum)	M	—	—
Java Server	M	Java Server	M
Name Server	M	—	—
Progress Messages (PROMSGS)	M	Language subset	O
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common ²	M
Secure Server	M	Perl	M
		Security Common ²	M
		Server-Side Security	M

1. M=Mandatory, R=Recommended, O=Optional

2. The Security Common subcomponent is not supported on a Linux PowerPC platform.

Client Networking

Table 104 lists the Client Networking components and subcomponents. When you choose the Complete installation option and install Client Networking, all components and subcomponents listed are installed.

Table 104: Client Networking components and subcomponents

(1 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
Character Base Tools	M	ADM Runtime—CHAR	M
		Base ADE	M
		Compile Tool—CHAR	M
		Procedure Editor—CHAR	M
Client-Side Web Services Deploy	R	Client-Side Security	R
		Security Common ²	M
		Web Services Basic	R
		Web Services Schema	R

Table 104: Client Networking components and subcomponents*(2 of 3)*

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
Database Administration Tools	M	4GL Database	M
		Audit Policy Maintenance	M
		Base ADE	M
		Character Administration	M
		Database Utilities	M
Name Server	M	—	—
OE Build Utility	R	—	—
Open Client Adapter Options Basic	R	AppServer Internet Adapter	R
		Common Broker	M
		Java Client Support	R
		Java Ext	M
		Java Server	M
		Java Class Tailoring	M
		Open Edge Adapter for SonicMQ	R
OpenEdge ESQL/C Clients	O	Database Tools	M
		ESQL CLient	M
		ICU PSC	M
		SQL Common	M
		SQL Server	M
OpenEdge SQL JDBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL Common	M
		SQL JDBC Client	M
		SQL Server	M
OpenEdge SQL ODBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL Common	M
		SQL ODBC Client	M
		SQL Server	M
Oracle DataServer Client	O	—	—
Progress Messages (PROMSGS)	M	Language subset	O
Remote Debugging	M	—	—

Table 104: Client Networking components and subcomponents (3 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
Runtime Client	M	Base Client—RT	M
		Character Client	M
		Crypto Tools	M
		ICU PSC	M
		Java Server	M
		XML	M
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common ²	M

1. M=Mandatory;R=Recommended;O=Optional

2. The Security Common subcomponent is not supported on a Linux PowerPC platform.

OpenEdge Replication

[Table 105](#) lists the OpenEdge Replication components and subcomponents. Choosing the Complete installation results in the installation of all components and subcomponents listed.

Table 105: OpenEdge Replication components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
Progress Messages (PROMSGS)	M	Language subset	O
Replication	M	Replication Common	M
		Replication Installation	M
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common ²	M
Secure Server	M	Perl	M
		Security Common ²	M
		Server-Side Security	M

1. M=Mandatory, R=Recommended, O=Optional

2. The Security Common subcomponent is not supported on a Linux PowerPC platform.

OpenEdge Replication Plus

Table 106 lists the OpenEdge Replication Plus components and subcomponents. Choosing the Complete installation results in the installation of all components and subcomponents listed.

Table 106: OpenEdge Replication Plus components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
Progress Messages (PROMSGS)	M	Language subset	O
Replication	M	Replication Common	M
		Replication Installation	M
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common ²	M
Secure Server	M	Perl	M
		Security Common ²	M
		Server-Side Security	M

1. M=Mandatory, R=Recommended, O=Optional

2. The Security Common subcomponent is not supported on a Linux PowerPC platform.

NameServer

Table 107 lists the NameServer components and subcomponents. Choosing the Complete installation results in the installation of all components and subcomponents listed.

Table 107: NameServer components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
NameServer	M	—	—
OpenEdge Management or OpenEdge Explorer Tools	M	Administration Server	M
		Common Broker	M
		Java Ext	M
		Java Server	M
		Ubroker Tools	M
Progress Messages (PROMSGS)	M	Language subset	O

1. M=Mandatory, R=Recommended, O=Optional

NameServer Load Balancer

Table 108 lists the NameServer Load Balancer components and subcomponents. Choosing the Complete installation results in the installation of all components and subcomponents listed.

Table 108: NameServer Load Balancer components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M

1. M=Mandatory, R=Recommended, O=Optional

OpenEdge Adapter for Sonic ESB

Table 109 lists the OpenEdge Adapter for Sonic ESB components and subcomponents. Choosing the Complete installation results in the installation of all components and subcomponents listed.

Table 109: OpenEdge Adapter for Sonic ESB components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
Java Class Tailoring	M	—	—
OpenEdge Adapter for Sonic ESB	M	—	—
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common ²	M

1. M=Mandatory, R=Recommended, O=Optional

2. The Security Common subcomponent is not supported on a Linux PowerPC platform.

OpenEdge Application Server—Basic

Table 110 lists the OpenEdge Application Server Basic components and subcomponents. Choosing the Complete installation results in the installation of all components and subcomponents listed.

Table 110: OpenEdge Application Server—Basic components and subcomponents (1 of 2)

Component	M/R/O ¹	Subcomponent	M/R/O
AppServer Runtime Client	M	–	–
Basic Server Options	R	SQL Server	M
		AppServer—Basic	R
		ADE Common Source	M
		ADM Runtime CHAR	M
		Audit Policy Maintenance	M
		Base ADE	M
		Base Client—4GL	M
		Character Client	M
		Common Broker	M
		Crypto Tools	M
		ICU PSC	M
		Name Server	R
		Procedure Editor CHAR	M
		Progress Databases	M
		Transaction Server—Basic	R
		Editor Source	M
		Web Static	M
		WebSpeed Messenger	R
		WebSpeed Runtime	M
		XML	M
Client-Side Web Services Deploy	R	Client-Side Security	R
		Security Common ²	M
		Web Services Schema	R
		Web Services Basic	R
Common Files	M	Common Files	M
		WebSpeed Common	M
OE Build Utility	R	–	–
OE Perl	M	–	–

Table 110: OpenEdge Application Server—Basic components and subcomponents (2 of 2)

Component	M/R/O ¹	Subcomponent	M/R/O
OpenClient Adapter Options Basic	R	AppServer Internet Adapter	R
		Common Broker	M
		Open Edge Adapter for Sonic MQ	R
		Java Client Support	R
		Java Server	M
		Java Ext	M
		Java Class Tailoring	M
Progress Messages (PROMSGS)	M	All languages	O
Remote Debugging	M	–	–
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common ²	M
Secure Server	M	Perl	M
		Security Common	M
		Server-Side Security	M
Server Admin and Configuration	M	Administration Server	M
		Common Broker	M
		Java Ext	M
		Java Server	M
		Name Server	R
		Ubroker Tools	M
Server Data Source Options	O	Database Tools	M
		ESQL Client	M
		ICU PSC	M
		Oracle Client	O
		SQL Common	M
		SQL JDBC Client	M
		SQL ODBC Client	M
		SQL Server	M

1. M=Mandatory, R=Recommended, O=Optional

2. The Security Common subcomponent is not supported on a Linux PowerPC platform.

OpenEdge Application Server—Enterprise

Table 111 lists the OpenEdge Application Server Enterprise components and subcomponents. Choosing the Complete installation results in the installation of all components and subcomponents listed.

Table 111: OpenEdge Application Server—Enterprise components and subcomponents

(1 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
AppServer Runtime Client	M	—	—
Client-Side Web Services Deploy	R	Client-Side Security	R
		Security Common ²	M
		Web Services Basic	R
		Web Services Schema	R
Common Files	M	Common Files	M
		WebSpeed Common	M
Enterprise Server Options	R	AppServer—Enterprise	R
		ADE Common Source	M
		ADM Runtime CHAR	M
		Base ADE	M
		Base Client—4GL	M
		Character Client	M
		Client-Side Security	R
		Common Broker	M
		Crypto Tools	M
		Editor Source	M
		ICU PSC	M
		Auditing Policy Maintenance	M
		NameServer	R
		Procedure Editor CHAR	M
		Progress Databases	M
		Security Common ²	M
		SQL Server	M
		Transaction Server—Enterprise	R
		Web Static	M
		WebSpeed Messenger	R
		WebSpeed Run-time	M
		XML	M

Table 111: OpenEdge Application Server—Enterprise components and subcomponents

(2 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
OE Build Utilities	R	—	—
OE Perl	M	—	—
Progress Messages (PROMSGS)	M	All Languages	O
Remote Debugging	M	—	—
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common ²	M
Secure Server	M	Perl	M
		Security Common ²	M
		Server-Side Security	M
Open Client Adapter Options Enterprise	R	AppServer Internet Adapter	R
		Common Broker	M
		Java Class Tailoring	M
		Java Client Support	R
		Java Ext	M
		Java Server	M
		OpenEdge Adapter for SonicMQ	R
		OpenEdge Adapter for Sonic ESB	R
		Web Services Adapter	M
		Web Services Admin Enabler	R
		Java Class Tailoring	M
		Web Services Schema	R
Server Admin and Configuration	M	Administration Server	M
		Common Broker	M
		Java Ext	M
		Java Server	M
		Name Server	R
		Ubroker Tools	M

Table 111: OpenEdge Application Server—Enterprise components and subcomponents

(3 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
Server Data Source Options	O	Database Tools	M
		ESQL Client	M
		ICU PSC	M
		Oracle Client	O
		SQL Common	M
		SQL JDBC Client	M
		SQL ODBC Client	M
		SQL Server	M

1. M=Mandatory, R=Recommended, O=Optional

2. The Security Common subcomponent is not supported on a Linux PowerPC platform.

OpenEdge DataServer for Oracle

Table 112 lists the OpenEdge DataServer for Oracle components and subcomponents. Choosing the Complete installation results in the installation of all components and subcomponents listed.

Table 112: OpenEdge DataServer for Oracle components and subcomponents

(1 of 2)

Component	M/R/O ¹	Subcomponent	M/R/O
Character Base Tools	M	ADM Runtime—CHAR	M
		Base ADE	M
		Compile Tool—CHAR	M
		Procedure Editor—CHAR	M
Common Files	M	Common Files	M
		WebSpeed Common	M
Database Administration Tools	M	4GL Database	M
		Audit Policy Maintenance Tool	M
		Base ADE	M
		Character Administration	M
		Database Utilities	M
Name Server	M	—	—
OE Build Utility	R	—	—

Table 112: OpenEdge DataServer for Oracle components and subcomponents (2 of 2)

Component	M/R/O ¹	Subcomponent	M/R/O
OpenClient Adapter Options Basic	R	AppServer Internet Adapter	R
		Common Broker	M
		Java Client Support	R
		Java Ext	M
		Java Server	M
		OpenEdge Adapter for SonicMQ	R
		Java Class Tailoring	M
Oracle DataServer	M	Broker	M
		Oracle DataServer	M
Oracle DataServer Client	O	—	—
OpenEdge Management or OpenEdge Explorer Tools	M	Administration Server	M
		Common Broker	M
		Java Ext	M
		Java Server	M
		Ubroker Tools	M
Progress Messages (PROMSGS)	M	Language subset	O
Remote Debugging	M	—	—
Runtime Client	M	Base Client—RT	M
		Character Client	M
		Crypto Tools	M
		ICU PSC	M
		Java Server	M
		XML	M
Schema Holder and Server	M	4GL Server	M
		Database Server	M
		Database Tools	M
		ICU PSC	M
		SQL Server	M

1. M=Mandatory, R=Recommended, O=Optional

OpenEdge Development Server

Table 113 lists the OpenEdge Development Server components and subcomponents. Choosing the Complete installation results in the installation of all components and subcomponents listed.

Table 113: OpenEdge Development Server components and subcomponents (1 of 4)

Component	M/R/O ¹	Subcomponent	M/R/O
Administration and Configuration	M	4GL Database	M
		Administration Server	M
		Auditing Policy Maintenance	M
		Base ADE	M
		Character Administration	M
		Common Broker	M
		Database Utilities	M
		Java Ext	M
		Java Server	M
		Name Server	R
		Ubroker Tools	M
Client-Side Web Services	R	Client-Side Security	R
		Security Common ²	M
		Web Services Basic	R
		WSDL Analyzer	R
		Web Services Schema	R
Common Files	M	Common Files	M
		WebSpeed Common	M
Development Server Options	R	4GL Database	M
		4GL Server	M
		ADM Runtime—CHAR	M
		AppServer—Development	R
		Base ADE	M
		Base Client—4GL	M
		Character Client	M

Table 113: OpenEdge Development Server components and subcomponents (2 of 4)

Component	M/R/O ¹	Subcomponent	M/R/O
Development Server Options (cont.)	R	Client-Side Security	R
		Common Broker	M
		Crypto Tools	M
		Database Server	M
		Database Tools	M
		ICU PSC	M
		NameServer	R
		Procedure Editor—CHAR	M
		Progress Databases	M
		Security Common ²	M
		SQL Server	M
		WebSpeed Messenger	R
		Transaction Server—Development	M
		Web Static	M
		WebSpeed Runtime	M
		XML	M
		Editor Source	M
		Progress Dynamics RT	R
Development Data Source Option	O	4GL Server	M
		Database Server	M
		Database Tools	M
		Database Utilities	M
		ESQL Client	M
		ICU PSC	M
		JDK	M
		Oracle Client	O
		Progress Databases	M
		SQL Common	M
		SQL JDBC Client	M
		SQL ODBC Client	M
		SQL Server	M

Table 113: OpenEdge Development Server components and subcomponents (3 of 4)

Component	M/R/O ¹	Subcomponent	M/R/O
Open Client Adapter Options	R	Web Services Schema	R
		AppServer Internet Adapter	R
		Common Broker	M
		Java Class Tailoring	M
		Java Client Support	R
		Java Ext	M
		Java Server	M
		OpenEdge Adapter for Sonic ESB	R
		OpenEdge Adapter for SonicMQ	R
		Proxy Generator	M
		Web Services Adapter Comm	M
		Web Services Admin Enable	R
OE Build Utility	R	–	–
OE Perl	M	–	–
Progress Messages (PROMSGS)	M	Language Subset	O
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common ²	M
Secure Server	M	Perl	M
		Security Common ²	M
		Server-Side Security	M
Server Source Code Options	R	ADE Common Source	O
		Editor Source	O

Table 113: OpenEdge Development Server components and subcomponents (4 of 4)

Component	M/R/O ¹	Subcomponent	M/R/O
Toolkit	M	—	—
Other Development Server Options	R	ADM Runtime—CHAR	O
		Application Debugger	R
		Character Client—Runtime	O
		Compile Tool—CHAR	O
		Crypto Tools	M
		Procedure Editor—CHAR	O
		Remote Debugging	M
4GL Utilities	R	XSD—4GL	R

1. M=Mandatory, R=Recommended, O=Optional

2. The Security Common subcomponent is not supported on a Linux PowerPC platform.

OpenEdge Enterprise RDBMS

Table 114 lists the OpenEdge Enterprise RDBMS components and subcomponents. Choosing the Complete installation results in the installation of all components and subcomponents listed.

Table 114: OpenEdge Enterprise RDBMS components and subcomponents (1 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
4GL	O	4GL Server	M
		ADM Runtime—CHAR	O
		Audit Policy Maintenance	M
		Base ADE	M
		Base Client—DA	M
		Character Administration	M
		Character Client	M
		Compile Tool—CHAR	O
		Crypto Tools	M
		ICU PSC	M
		Oracle Client	O
		Procedure Editor—CHAR	M
		SQL Server	M
		XML	M

Table 114: OpenEdge Enterprise RDBMS components and subcomponents (2 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
Client-Side Web Services	R	Client-Side Security	R
		Security Common ²	M
		Web Services Basic	R
		Web Services Schema	R
Common Files	M	Common Files	M
		WebSpeed Common	M
Database Management	M	4GL Server	M
		Database Server	M
		Database Tools	M
		Database Utilities	M
		ICU PSC	M
		Legacy 83 Utilities	M
		Legacy 91 Utilities	M
		OE 11.3 DB Utilities	M
		Progress Databases	M
		SQL Server	M
Failover Clusters	R	Cluster Common	M
Name Server	M	–	–
OE Build Utility	R	–	–
OE Perl	M	–	–
Open Client Adapter Options Basic	R	AppServer Internet Adapter	R
		Common Broker	M
		Java Client Support	R
		Java Ext	M
		Java Server	M
		Java Class Tailoring	M
		OpenEdge Adapter for SonicMQ	R
OpenEdge ESQL/C Clients	O	Database Tools	M
		ESQL Client	M
		ICU PSC	M
		SQL Server	M
		SQL Common	M

Table 114: OpenEdge Enterprise RDBMS components and subcomponents (3 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
OpenEdge SQL JDBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL JDBC Client	M
		SQL Server	M
		SQL Common	M
OpenEdge SQL ODBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL ODBC Client	M
		SQL Server	M
		SQL Common	M
OpenEdge Management or OpenEdge Explorer Tools—DB	M	Administration Server	M
		Common Broker	M
		Java Ext	M
		Java Server	M
Progress Messages (PROMSGS)	M	Language subset	O
Remote Debugging	M	—	—
Secure Server	M	Perl	M
		Security Common ²	M
		Server-Side Security	M
SQL	O	Database Tools	M
		ESQL Client	M
		ICU PSC	M
		SQL Common	M
		SQL JDBC Client	M
		SQL ODBC Client	M
		SQL Server	M

1. M=Mandatory, R=Recommended, O=Optional

2. The Security Common subcomponent is not supported on a Linux PowerPC platform.

OpenEdge Personal RDBMS

Table 115 lists the Personal RDBMS components and subcomponents. Choosing the Complete installation results in the installation of all components and subcomponents listed.

Table 115: OpenEdge Personal RDBMS components and subcomponents (1 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
4GL	O	4GL Server	M
		ADM Runtime—CHAR	O
		Audit Policy Maintenance	M
		Base ADE	M
		Base Client—DA	M
		Character Administration	M
		Character Client	M
		Compile Tool—CHAR	O
		Crypto Tools	M
		ICU PSC	M
		Oracle Client	O
		Procedure Editor—CHAR	M
		SQL Server	M
		XML	M
Client-Side Web Services Deploy	R	Client-Side Security	R
		Security Common ²	M
		Web Services Basic	R
		Web Services Schema	R
Common Files	M	Common Files	M
		WebSpeed Common	M
Database Management	M	4GL Server	M
		Database Server	M
		Database Tools	M
		Database Utilities	M
		ICU PSC	M
		Legacy 83 Utilities	M
		Legacy 91 Utilities	M
		OE 11.3 DB Utilities	M
		Progress Databases	M
		SQL Server	M
Name Server	M	—	—

Table 115: OpenEdge Personal RDBMS components and subcomponents (2 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
OE Build Utility	R	—	—
OE Perl	M	—	—
Open Client Adapter Options Basic	R	AppServer Internet Adapter	R
		Common Broker	M
		Java Client Support	R
		Java Ext	M
		Java Server	M
		Java Class Tailoring	M
		OpenEdge Adapter for SonicMQ	R
OpenEdge ESQL/C Clients	O	Database Tools	M
		ESQL Client	M
		ICU PSC	M
		SQL Server	M
		SQL Common	M
OpenEdge SQL JDBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL JDBC Client	M
		SQL Server	M
		SQL Common	M
OpenEdge SQL ODBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL ODBC Client	M
		SQL Server	M
		SQL Common	M
OpenEdge Management or OpenEdge Explorer Tools —DB	M	Administration Server	M
		Common Broker	M
		Java Ext	M
		Java Server	M
Progress Messages (PROMSGS)	M	Language subset	O
Remote Debugging	M	—	—

Table 115: OpenEdge Personal RDBMS components and subcomponents (3 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
Secure Server	M	Perl	M
		Security Common ²	M
		Server-Side Security	M
SQL	O	Database Tools	M
		ESQL Client	M
		ICU PSC	M
		SQL Common	M
		SQL JDBC Client	M
		SQL ODBC Client	M
		SQL Server	M

1. The 4GL component of the OpenEdge Personal RDBMS includes the Client Networking functionality.

2. The Security Common subcomponent is not supported on a Linux PowerPC platform.

OpenEdge Workgroup RDBMS

Table 116 lists the OpenEdge Workgroup RDBMS components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents listed.

Table 116: OpenEdge Workgroup RDBMS components and subcomponents (1 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
4GL	O	4GL Server	M
		ADM Run-time—CHAR	O
		Auditing Policy Maintenance	M
		Base ADE	M
		Base Client—DA	M
		Character Administration	M
		Character Client	M
		Compile Tool—CHAR	O
		Crypto Tools	M
		ICU PSC	M
		Oracle Client	O
		Procedure Editor—CHAR	M
		SQL Server	M
		XML	M

Table 116: OpenEdge Workgroup RDBMS components and subcomponents (2 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
Client-Side Web Services	R	Client-Side Security	R
		Security Common ²	M
		Web Services Basic	R
		Web Services Schema	R
Common Files	M	Common Files	M
		WebSpeed Common	M
Database Management	M	4GL Server	M
		Database Server	M
		Database Tools	M
		Database Utilities	M
		ICU PSC	M
		Legacy 83 Utilities	M
		Legacy 91 Utilities	M
		OE 11.3 DB Utilities	M
		Progress Databases	M
		SQL Server	M
Name Server	M	—	—
OE Build Utility	R	—	—
OE Perl	M	—	—
Open Client Adapter Options Basic	R	AppServer Internet Adapter	R
		Common Broker	M
		Java Client Support	R
		Java Ext	M
		Java Server	M
		Java Class Monitoring	M
		Java Class Tailoring	M
		OpenEdge Adapter for SonicMQ	R
OE ESQL/C Clients	O	Database Tools	M
		ESQL Client	M
		ICU PSC	M
		SQL Common	M
OE SQL JDBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL JDBC Client	M
		SQL Common	M

Table 116: OpenEdge Workgroup RDBMS components and subcomponents (3 of 3)

Component	M/R/O ¹	Subcomponent	M/R/O
OE SQL ODBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL ODBC Client	M
		SQL Common	M
OpenEdge Management or OpenEdge Explorer Tools—DB	M	Administration Server	M
		Java Ext	M
		Java Server	M
		Common Broker	M
Progress Messages (PROMSGS)	M	Language subset	O
Remote Debugging	M	—	—
Secure Server	M	Perl	M
		Security Common ²	M
		Server-Side Security	M
SQL	O	Database Tools	M
		ESQL Client	M
		ICU PSC	M
		SQL Common	M
		SQL JDBC Client	M
		SQL ODBC Client	M
		SQL Server	M

1. M=Mandatory, R=Recommended, O=Optional

2. The Security Common subcomponent is not supported on a Linux PowerPC platform.

OpenEdge SQL Client Access

[Table 117](#) lists the SQL Client Access components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents listed.

Table 117: OpenEdge SQL Client Access components and subcomponents (1 of 2)

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
Name Server	M	—	—

Table 117: OpenEdge SQL Client Access components and subcomponents (2 of 2)

Component	M/R/O ¹	Subcomponent	M/R/O
OpenEdge ESQL/C Clients	O	Database Tools	M
		ESQL Client	M
		ICU PSC	M
		SQL Server	M
		SQL Common	M
OpenEdge SQL JDBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL JDBC Client	M
		SQL Server	M
		SQL Common	M
OpenEdge SQL ODBC Clients	O	Database Tools	M
		ICU PSC	M
		SQL ODBC Client	M
		SQL Server	M
		SQL Common	M
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common ²	M

1. M=Mandatory, R=Recommended, O=Optional

2. The Security Common subcomponent is not supported on the Linux Power PC platform.

Query/Results

Table 118 lists the Query/Results components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents listed.

Table 118: Query/Results components and subcomponents (1 of 2)

Component	M/R/O ¹	Subcomponent	M/R/O
Character Base Tools	M	ADM Runtime—CHAR	M
		Base ADE	M
		Compile Tool—CHAR	M
		Procedure Editor—CHAR	M

Table 118: Query/Results components and subcomponents (2 of 2)

Component	M/R/O ¹	Subcomponent	M/R/O
Client-Side Web Services Deploy	R	Client-Side Security	R
		Security Common ²	M
		Web Services Basic	R
		Web Services Schema	R
Common Files	M	Common Files	M
		WebSpeed Common	M
Database Administration Tools	O	4GL Database	M
		Audit Policy Maintenance	M
		Base ADE	M
		Character Administration	M
		Database Utilities	M
OE Build Utility	R	—	—
Open Client Adapter Options Basic	R	AppServer Internet Adapter	R
		Common Broker	M
		Java Client Support	R
		Java Ext	M
		Java Server	M
		Java Class Tailoring	M
		OpenEdge Adapter for SonicMQ	R
Oracle DataServer Client	O	—	—
Progress Messages (PROMSGS)	M	Language Subset	O
Query Client	M	Base Client—Query	M
		Character Client	M
		Crypto Tools	M
		ICU PSC	M
		Java Server	M
		XML	M
Remote Debugging	M	—	—
Results	M	Base ADE	M
		Results (Char)	M
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common ²	M

1. M=Mandatory, R=Recommended, O=Optional

2. The Security Common subcomponent is not supported on a Linux PowerPC platform.

WebSpeed Messenger

Table 119 lists the WebSpeed Messenger components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents listed.

Table 119: WebSpeed Messenger components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Messenger Component	M	Java Server	M
		Messenger Runtime Support	M
Progress Messages (PROMSGS)	M	Language subset	O
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common ²	M
Web Static	M	—	—
WebSpeed Common	M	—	—

1. M=Mandatory;R=Recommended;O=Optional

2. The Security Common subcomponent is not supported on a Linux PowerPC platform.

Web Services Adapter

Table 120 lists the Web Services Adapter components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents listed.

Table 120: Web Services Adapter components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Name Server	M	—	—
Secure Clients	M	Client-Side Security	R
		Perl	M
		Security Common ²	M
Web Services Adapter	M	Web Services Adapter	M
		Web Services Admin Enable	R

1. M=Mandatory;R=Recommended;O=Optional

2. The Security Common subcomponent is not supported on the Linux Power PC platform.

OpenEdge Management SE

[Table 121](#) lists the OpenEdge Management SE components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents listed.

Table 121: OpenEdge Management SE components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M
Fathom	M	Administration Server	M
		Client-Side Security	R
		Fathom Cmn	M
		Fathom Common	M
		Fathom Doc	M
		Fathom Tailor	M
		Fathom Class Tailoring	M
Progress Messages (PROMSGS)	M	Language subset	O

1. M=Mandatory;R=Recommended;O=Optional

SNMP Adapter

[Table 122](#) lists the SNMP Adapter components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents listed.

Table 122: SNMP Adapter components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
SNMP Adapter	M	SNMP Common	M

1. M=Mandatory;R=Recommended;O=Optional

OpenEdge TDE

[Table 123](#) lists the OpenEdge TDE components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents listed.

Table 123: OpenEdge TDE components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M

1. M=Mandatory;R=Recommended;O=Optional

OpenEdge Multi-tenant Table

[Table 124](#) lists the OpenEdge Multi-tenant Table components and subcomponents. Choosing the Complete installation option results in the installation of all components and subcomponents listed.

Table 124: OpenEdge Multi-tenant Table components and subcomponents

Component	M/R/O ¹	Subcomponent	M/R/O
Common Files	M	Common Files	M
		WebSpeed Common	M

1. M=Mandatory;R=Recommended;O=Optional

Preinstallation Checklist for Windows

The checklist is a tool to help technical personnel determine and record product installation choices **before** running the OpenEdge Release 11.3 Installation Utility. Familiarize yourself with the checklist's content and then work through each topic. Refer to this completed checklist when you start the installation.

Note: This checklist covers both the Windows 32-bit and Windows 64-bit platforms. Unless specifically identified, sections apply to both platforms.

Before you start

Obtain these documents to complete the checklist:

- ☐ *License Addendum* and *OpenEdge Release Notes* from the OpenEdge package

Note: For information on installing OpenEdge components using an electronic License Addendum File, refer to the [“Using an Electronic License Addendum file”](#) section on page 116.

- ☐ Windows Installation online help available for download from the .tar file from the Progress Download Center at <http://www.progress.com/esd>
- ☐ “Chapter 12, “OpenEdge Installation Products and Components in Windows”

Products to install

- ☐ Obtain serial numbers and control codes from the *License Addendum*. Control codes are not case sensitive and can be entered in any order.

Note: When you upload the *License Addendum*, all the products are populated the **Product(s) to be Installed** text area in the **Serial Number and Control Codes** dialog box. You can remove unwanted products from the list by selecting them and then clicking the **Remove** button.

Prerequisite third-party software (Windows 32-bit only)

If your planned installation meets each of the following conditions, you are prompted to accept a Microsoft .NET Framework software installation which will run after the OpenEdge installation concludes:

- ☐ At least one product for which you enter a control code requires Microsoft .NET Framework 4.0 software. The OpenEdge 11.3 development products that require .NET Framework are:

4GL Development System	OpenEdge Ultra Controls
Progress Developer Studio for OpenEdge	OpenEdge Development Server
OpenEdge Studio	—

For deployment products, you can optionally install the Microsoft .NET Framework. If your application uses the OpenEdge Ultra Controls, the Framework is necessary for your application to run correctly.

- ☐ The Microsoft .NET Framework is not currently installed on the machine to which you are installing OpenEdge.

Note: The OpenEdge installation will install the English version of the Microsoft .NET Framework. If you require a different language, you must install the Framework **before** running the OpenEdge installation. Frameworks in additional languages are available from the Progress Download Center at <http://www.progress.com/esd>.

The Microsoft .NET framework installation must be performed interactively. If you are planning a silent installation, you must install the Microsoft .NET framework first.

If your planned installation includes the OpenEdge Ultra Controls:

- ☐ You must also install, or be adding additional products to an installation that contains one of the following products:

Progress Developer Studio for OpenEdge	OpenEdge Studio
4GL Development System	OpenEdge Development Server

- ☐ You are prompted to acknowledge that you have a license for the Microsoft Office 2007 UI Capabilities.

Values from your existing OpenEdge installation (Windows 32-bit only)

If the Installation Program detects a prior OpenEdge Release 11.0 installation on the machine to which you are installing OpenEdge 11.3, you are prompted to use the field values defined for the previous installation as default field values in the OpenEdge 11.3 installation. Proceed as follows:

- ☐ Choose **Yes** to accept the values as default field values. A reminder **Initial values used from a previous installation** displays at the bottom of each dialog box affected.
- ☐ Choose **No** to decline the option. Only the Installation Program default field values appear.

Installation and working directories

For OpenEdge:

- ☐ Accept the default installation directory location provided, **C:\Progress\OpenEdge**, or define another location: _____ . **Do not** use the pathname where other versions of Progress or OpenEdge products are already installed on your system.
- ☐ Accept the default working directory location provided, **C:\OpenEdge\WRK**, in which your applications, databases, and log files will reside, or define another location: _____ . **Do not** make your working directory a subdirectory under the OpenEdge installation path.

For OpenEdge Management and OpenEdge Explorer:

- ☐ Accept the default installation directory location provided, **C:\Progress\oemgmt**, or define another location: _____ . **Do not** use the pathname where other versions of Progress or OpenEdge products are already installed on your system.
- ☐ Accept the default working directory location provided, **C:\OpenEdge\wrk_oemgmt**, in which your OpenEdge Management or OpenEdge Explorer files will reside, or define another location: _____ . **Do not** make your working directory a subdirectory under the OpenEdge installation path.

See the [“OpenEdge Management or OpenEdge Explorer”](#) section on page 90 for more information.

Installation type

- ☐ Complete (default option)
- ☐ Custom

If you chose a custom installation, use [Chapter 12, “OpenEdge Installation Products and Components in Windows”](#) to determine which Optional and Recommended components to install.

Recommended and optional components

Depending on the products for which you enter control codes, you can be prompted to identify components for the purposes indicated in the sections that follow.

WebSpeed with local Web server, OpenEdge Adapter for Sonic ESB, and/or OpenEdge Explorer

Each is a recommended component that must be configured during the installation.

For example, the OpenEdge Adapter for Sonic ESB is a recommended component of OpenEdge Application Server Enterprise Edition and OpenEdge Studio. Many OpenEdge products required access to at least one of the recognized Web server types. See the [“Web server”](#) section on page 433 for a list of products.

OpenEdge Explorer is a browser-based tool that allows you to set configuration properties for various OpenEdge resources, as well as to start and stop them, view their status, and view their log file data.

Proceed with this choice based on these options:

- ☐ Accept the check mark, or select the check box associated with a recommended component to identify it for configuration. During the installation, you may be prompted to specify the component's configuration details.

To note configuration details for the OpenEdge Adapter for Sonic ESB see the [“OpenEdge Adapter for Sonic ESB”](#) section on page 432. To note configuration details for the Web Server type see the [“Web server”](#) section on page 433. There are no additional configuration details for OpenEdge Explorer.

Caution: You must select the OpenEdge Explorer option to be able to work with the server functionality in Progress Developer Studio for OpenEdge. If you do not select this option, you can manually configure OpenEdge Explorer after installation to work with the server functionality. For more information, see the Progress Developer Studio for OpenEdge online help.

- ☐ Deselect the check box, or leave the check box blank if you do not plan to configure the recommended component. (If a check box is grayed out, the component is not available.)

Note: The OpenEdge installation program does not automatically install OpenEdge Sonic ESB samples. For more information on installing these components, see [OpenEdge Development: Messaging and ESB](#).

Progress Dynamics (Windows 32-bit only)

An optional component that can be installed with OpenEdge products such as OpenEdge Studio and Progress Developer Studio for OpenEdge. When performing a Complete installation, determine how to proceed with this choice based on these options:

- ☐ Accept the check mark associated with Progress Dynamics when installing OpenEdge Studio, or uncheck it to bypass installing this optional component.
- ☐ If you are installing Progress Developer Studio for OpenEdge, the Progress Dynamics component is **unchecked** by default. The Progress Dynamics component supports the AppBuilder functionality within Progress Developer Studio for OpenEdge. You must check this option in this dialog box to install it so that you can enable Progress Dynamics in Progress Developer Studio for OpenEdge.

To note configuration details for the Progress Dynamics option see the [“Progress Dynamics \(Windows 32-bit only\)”](#) section on page 434.

Database

If you are installing a database, the database server connection to support queries written in ABL is installed by default:

- ☐ Accept the system-default to install the SQL database server connection to support queries written in SQL, or deselect the option.

Note: If you are installing a product that requires SQL access, such as Progress Developer Studio for OpenEdge, this prompt is not provided, and the SQL engine is automatically installed.

OpenEdge Adapter for Sonic ESB

Respond to the following points if you are configuring the OpenEdge Adapter for Sonic ESB as a recommended component or as a stand-alone product (downloadable from the Progress Download Center):

- ☐ Accept the default host name value provided in the **Container Name** field or define a different container name; this name must be unique to the Sonic management broker: _____.
- ☐ Accept the default values provided in the **Domain Name**, **Connection URL**, **User Name** and **Password** fields, or define unique values for any of these fields:

_____.

- ☐ The OpenEdge Adapter for Sonic ESB requires the configuration of the Sonic Management Console to administer OpenEdge Services. Proceed as shown in the following table:

If the Sonic Domain Manager is installed ...	Then ...
On a machine on which you are installing all other OpenEdge products	During the installation, select the local directory in which the Domain Manager resides. Specify your choice in the Sonic ESB Home Directory field: _____. Note: If your Sonic Domain Manager is not running at the time of installation, you must perform an offline configuration, as described in OpenEdge Application Server: Administration .
On a machine remote from all other OpenEdge products you are installing	As a postinstallation task, you must manually configure the remote system's Sonic Management Console, using the installation instructions available in OpenEdge Application Server: Administration .

Options to install your Progress Developer Studio for OpenEdge plug-ins to additional targets

If you are installing Progress Developer Studio for OpenEdge, during the installation process, you can choose to link your Progress Developer Studio for OpenEdge plug-ins to additional environments, provided the framework version is 3.7.1 and 3.8.0.

Proceed as follows:

- ☐ Check **Enter additional Eclipse workbench install targets on the next dialog** to specify an additional installation location for your Progress Developer Studio for OpenEdge plug-ins. In the subsequent dialog identify the target directories:

Note: Your additional target directories must contain the `eclipse.exe` file. When you install 32-bit Windows installer for OpenEdge, you must use 32-bit Eclipse. When you install 64-bit Windows installer for OpenEdge, you must use 64-bit Eclipse.

- ☐ Leave **Enter additional Eclipse workbench install targets on the next dialog** unchecked to only install the Progress Developer Studio for OpenEdge plug-ins in the current installation destination.

Web server

You must have access to at least one of the recognized Web server types if you plan to install any of the following OpenEdge products:

- The WebSpeed Messenger, Secure AppServer Internet Adapter (AIA) or Web Services Adapter (WSA) products to develop and/or deploy Web applications.
- OpenEdge Development Server if you plan to use Progress Dynamics functionality.
- WebSpeed as a component or subcomponent of another OpenEdge product, if you plan to use this functionality. (If you do not plan to use WebSpeed functionality and the check box is available, leave the check box blank to bypass these Web server choices during the installation process.)

If you are configuring a Web server, respond to the following points for each Web server you intend to use. Refer to the installation online help topic “Selecting a Web Server Type,” as needed:

- ☐ Identify the type of Web server:
 - Microsoft Internet Information System (IIS) or ISAPI-compatible
 - Sun Web server or NSAPI-compatible
 - CGI-compatible
- ☐ Note the Web server’s machine location:
 - On the same machine where you plan to install your OpenEdge products

- On a machine that is different from the machine on which you are installing your OpenEdge products
- ☐ For a Microsoft Internet Information System (IIS) or ISAPI-compatible Web server:
 - Accept the default directories provided that appear in the **Web Server Script directory** and the separate **Web Server Document Root directory fields**, respectively. (These default directories are independent of each other.) **Or**, you can select a different location for either or both directories: _____ . (If you select an alternative Web server script directory, it must be an existing directory that your Web server references.)
 - Deselect the **Copy static HTML files to Document Root directory** checkbox and select the **Create virtual directory for static HTML files** checkbox. This step enables OpenEdge to create an alias that points to the WebSpeed HTML files in the OpenEdge installation directory.
- ☐ For either a Sun Web server an NSAPI-compatible Web server, **or** a CGI-compatible Web server:
 - Define the directory path for the **Web Server Scripts directory** field: _____ and the **Web Server Document Root directory** field: _____ .
 - Select the Web server's document root directory for the **Copy static HTML to the Document Root directory** field so that during the installation, the WebSpeed Workshop HTML files are copied to this location: _____ .

Progress Dynamics (Windows 32-bit only)

Respond to the following points, preparing the values you enter in the **Dynamics Options** dialog box:

- ☐ Select the **Install/upgrade Dynamics repository** check box to create or upgrade the `icfdb` repository file in the default location **C:\OpenEdge\WRK\databases**, **or** select another directory location: _____ .
- ☐ If you plan to develop Web applications with this product, proceed with **one** of these tasks:
 - Select the **Copy the Progress Dynamics static HTML files to the Document Root Directory** option to copy the static Dynamic Web files to your Web server's document root directory.
 - Create a virtual directory on your Web server that points to the location of the static files. (The static files physically reside in `install\tty\icf\ry`.)

Language in which online messages appear

The Progress Messages file `PROMSGS` contains error and information messages that appear when you are working in OpenEdge. Respond to these points related to `PROMSGS`:

- ☐ Identify the default language in which error and information messages appear:
_____.
- ☐ Identify other languages to install (optional):
_____.

For a complete list of languages shipped with OpenEdge 11.3 and supplemental `PROMSGS` translations that are available for download from the Progress Download Center at <http://www.progress.com/esd>, see [Appendix D, "OpenEdge National Language Support."](#)

Character set, date, and number formats

- ☐ Select the format in which the character set, date, and number appear from the lists associated with each option:
 - **Character Set** (including **Collation** and **Case**):
_____.
 - **Date Format**:
_____.
 - **Number Format**:
_____.

The software uses the settings you choose to tailor your OpenEdge startup parameter file, `startup.pf`.

Web Services Adapter (WSA)

If you are installing the Web Services Adapter (WSA) product or a product that contains the WSA, respond to these points:

- ☐ Accept the default URL provided in the **URL** field **`http://<machinename:80>/wsa/wsa1`**, or change it to:
_____.

Consider your WSA configuration and your Web Server or Java Servlet engine. The URL field defines the location for the sample Web Services `wsa1`. (When you deploy a Web service, you deploy it to a WSA instance, which defines the root URL used to access the Web service and handles all of its client communications. Each WSA instance manages its own set of deployed Web services.)

- ☐ Retain **or** change the WSA authentication feature.

The default security setting to perform administrative tasks requires a valid username and password each time a connection to a WSA instance is initiated. If you select the check box associated with the **Disable Authentication** option, you disable this default security setting, eliminating any authorization requirement to administer the WSA.

Options to secure your AdminServer

Set up AdminServer authorization options either during **or** after the installation process:

- ☐ **During the installation process** — You can optionally set up both, one, or none of the AdminServer authorization options as shown in the following table:

This AdminServer security option ...	Requires you to ...
User (individual) Authorization option	<ul style="list-style-type: none"> Select the Require a Username and Password checkbox Set up a valid individual user name and password at the operating system level for each individual to whom you are granting individual privileges
Group¹ Authorization option	<ul style="list-style-type: none"> Select the Enable Group Checking checkbox Accept the initial default group provided, PSC Admin, which you can then use as a template for each unique group that you define

1. For details about the guidelines, naming conventions, and restrictions concerning the **Group Authorization** option, refer to the Installation online help topic "Establishing AdminServer Authorization Options."

- ☐ **As a postinstallation task** — To create groups or to create additional groups, you must reinstall your OpenEdge medium and perform the **Group Authorization** option tasks at that time.

OpenEdge Business Process Server

Respond to the following options if you are installing the OpenEdge Business Process Server as a recommended component or as a stand-alone product (downloadable from the Progress Download Center):

- ☐ Accept the default or update the port values provided in the **Business Process Application Server** dialog box. Select **Advanced** to specify additional JBOSS port details. _____.

Note: You cannot enter duplicate port numbers or leave the port fields empty. The range for the port numbers is 1024 to 49151.

-
- ☐ Accept the default values or update the provided value in the **Portal URL** in the **Business Process Server Portal URL** dialog.
-

Note: The **Portal URL** is generated using the host name and the domain name of your machine where you are installing Business Process Server. The valid format is **http://<host_name>.<domain_name>:<port_number>**. For example, <http://localhost.apac.progress.com:18256>.

- ☐ The OpenEdge Business Process Server requires you to create a database for OpenEdge. Proceed as shown in the following table in the **Business Process Server Database Details** dialog box:

Option	Requires you to...
Server Details	Enter the port number. Note: If you are installing all the components, Database name displays oebps because the name of the OpenEdge database you use during development is oebps.db. If you are installing only OpenEdge Progress Developer Studio and have selected OpenEdge Business Process Server for PDS check box in the Configuring/Installing Components dialog box, it displays oebpsdev because the name of the OpenEdge database you use during development is oebpsdev.db. We recommend you not to change the Database name .
OpenEdge Database	Verify the Location for storing the database files. Note: We recommend you not to change the Location .
Create Database	Select to create the database.

- ☐ Accept the email details provided in the **Email Configuration** dialog box or update values as:

Option	Requires you to...
Outgoing Email Server	Enter the address used for sending email from Business Process Server.
Administrator Email	Enter the details for configuring the Business Process Server email properties.
Incoming Server	Enter the email properties for receiving email sent to Business Process Server if you want different email servers for incoming and outgoing mail.

Preinstallation Checklist for UNIX

The checklist is a tool to help technical personnel determine and record product installation choices **before** running the OpenEdge Release 11.3 Installation Utility. Familiarize yourself with the checklist's content and then work through each topic. Refer to this completed checklist when you start the installation.

Before you start

Obtain these documents to complete the checklist:

- ☐ *License Addendum* and *OpenEdge Release Notes* from the OpenEdge package

Note: For information on installing OpenEdge components using the License Addendum File, refer to the “[Obtaining an Electronic License Addendum file](#)” section on page 72.

- ☐ UNIX Installation online help available for download from the `.tar` file from the Progress Download Center at <http://www.progress.com/esd>
- ☐ [Chapter 13, “OpenEdge Installation Products and Components on UNIX”](#)

Java platform requirements

If your installation requires either the JDK or the JRE. Check the Java-specific requirements for your supported platform as identified in the “[Chapter 2, “UNIX Systems Installation Requirements.”](#)”

Products to install

- ☐ Obtain serial numbers and control codes from the *License Addendum*. Control codes are not case sensitive and can be entered in any order.

Values from your existing OpenEdge installation

If the Installation Program detects an earlier OpenEdge Release 10 installation on the machine to which you are installing OpenEdge 11.3, you are prompted to use the values defined for the prior installation as default values in the OpenEdge 11.3 installation. Proceed as follows:

- ☐ Choose **Yes** to accept the values as default values. A reminder **Initial values used from a previous installation** displays at the bottom of each dialog box affected.
- ☐ Choose **No** to decline the option. Only the Installation Program default values appear.

Installation and working directories

- ☐ Accept the default destination directory location provided, **/usr/dlc**, or define another location: _____. **Do not** use the pathname where other versions of Progress or OpenEdge products are already installed on your system.

Note: Here dlc refers to the **OpenEdge installation directory**.

- ☐ Accept the default working directory location provided, **/usr/wrk**, in which your applications, databases, and log files will reside, **or** define another location: _____. **Do not** make your working directory a subdirectory under the OpenEdge installation path.

For OpenEdge Management and OpenEdge Explorer:

- ☐ Accept the default installation directory location provided, `/usr/oemgmt`, or define another location: _____. **Do not** use the pathname where other versions of Progress or OpenEdge products are already installed on your system.
- ☐ Accept the default working directory location provided, `/usr/wrk_oemgmt`, in which your OpenEdge Management or OpenEdge Explorer files will reside, or define another location: _____. **Do not** make your working directory a subdirectory under the OpenEdge installation path.

See the [“OpenEdge Management or OpenEdge Explorer”](#) section on page 90 for more information.

Installation type

- ☐ Complete (default option)
- ☐ Custom

If you chose a custom installation, use [Chapter 13, “OpenEdge Installation Products and Components on UNIX”](#) to determine which Optional and Recommended components to install.

Database

If you are installing a database, the database server connection to support queries written in ABL is installed by default:

- ☐ Accept the system-default to install the SQL database server connection to support queries written in SQL, or deselect the option.

OpenEdge Management or OpenEdge Explorer

If you are installing a database or server/broker installation, you have the option to configure OpenEdge Management or OpenEdge Explorer. OpenEdge Management or OpenEdge Explorer is a browser-based tool that allows you to set configuration properties for various OpenEdge resources, as well as to start and stop them, view their status, and view their log file data.

- ☐ Choose **Yes** to configure OpenEdge Management or OpenEdge Explorer as part of your installation.
- ☐ Choose **No** to decline the option.

See the [“OpenEdge Management or OpenEdge Explorer”](#) section on page 90 for more information.

OpenEdge Adapter for Sonic ESB

Respond to the following points if you are installing the OpenEdge Adapter for Sonic ESB in either of the following situations:

- As a stand-alone product (downloadable from the Progress Download Center).
- As a component of another product, such as the OpenEdge Application Server Enterprise Edition and 4GL Development System, and you want to install the adapter. (If you do not want to install the adapter, choose **No** to bypass the adapter choices during the installation process.)

Note: The OpenEdge installation program does not automatically install OpenEdge Sonic ESB samples. For more information on installing these components, see [OpenEdge Development: Messaging and ESB](#).

- ☐ Accept the host name default value provided for the **Container Name** field **or** define a different container name; it must be unique to the Sonic management broker: _____.
- ☐ Accept the default values provided for the **Domain Name**, **Connection URL**, **User Name** and **Password** fields, **or** define unique values for any of these fields: _____.

The OpenEdge Adapter for Sonic ESB requires the configuration of the Sonic Management Console to administer OpenEdge Services. Proceed as shown in the following table:

If the Sonic Domain Manager is installed ...	Then ...
On a machine on which you are installing all other OpenEdge products	During the installation, select the local directory in which the Domain Manager resides. Specify your choice in the SonicESB Home Directory field: _____.
On a machine remote from all other OpenEdge products you are installing	As a postinstallation task, you must manually configure the remote system's Sonic Management Console, using the installation instructions available in OpenEdge Application Server: Administration .

Web server

You must have access to at least one of the recognized Web server types if you plan to install any of the following OpenEdge products:

- The WebSpeed Messenger, Secure AppServer Internet Adapter (AIA), or Web Services Adapter (WSA) products to develop and/or deploy Web applications.
- WebSpeed as a component or subcomponent of another OpenEdge product, if you plan to use this functionality. (If you do not plan to use WebSpeed functionality, choose **No** to bypass these Web server choices during the installation process.)

Respond to the following points for each Web server you intend to use, referring to the Installation online help topic “Selecting a Web Server Type,” as needed:

- ☐ Identify Web server type:
 - Sun Web server or NSAPI-compatible Web server
 - CGI-compatible Web server
- ☐ Identify the Web server’s location:
 - On the same machine where you plan to install your OpenEdge products
 - On a machine that is different from the machine on which you are installing your OpenEdge products
- ☐ For either a Sun Web server, or NSAPI-compatible Web server, or CGI-compatible Web server:
 - Identify the directory you will define as your default value for the **Web Server Script directory** field: _____.
 - Identify the Web server’s document root directory you will define for the **Copy static HTML to the docroot** directory field to enable the WebSpeed Workshop HTML files to be copied to this location during the installation:
_____.

Language in which messages appear

The Progress Messages file `PROMSGS` contains error and information messages that appear when you are working in OpenEdge. Respond to these points related to `PROMSGS`:

- ☐ Identify the default language in which error and information messages appear:
_____.
- ☐ Identify other languages to install (optional):
_____.

For a complete list of languages shipped with OpenEdge 11.3 and supplemental `PROMSGS` translations available for download from the Progress Download Center at <http://www.progress.com/esd>, see [Appendix D, “OpenEdge National Language Support.”](#)

Character set, date, and number formats

- ☐ Select the format in which the character set, date, and number appear from the lists associated with each option:

- **Character Set** (including **Collation** and **Case**):

_____.

- **Date Format:**

_____.

- **Number Format:**

_____.

The software uses the settings you choose to tailor your OpenEdge startup parameter file, `startup.pf`.

Web Services Adapter (WSA)

If you are installing the WSA or a product that contains the WSA:

- ☐ Accept the default URL provided in the **URL** field **http://<machinename:80>/wsa/wsa1**, or change it to: _____.

Consider your WSA configuration and your Web Server or Java Servlet engine. The URL field defines the location of the sample Web Services `wsa1`. (When you deploy a Web service, you deploy it to a WSA instance, which defines the root URL used to access the Web service and handles all of its client communications. Each WSA instance manages its own set of deployed Web services.)

- ☐ Type **Y** to remove the default authorization requirement or type **N** to retain the default authorization requirement.

For the WSA authentication option, the default security setting to perform administrative tasks requires a valid username and password each time a connection to a WSA instance is initiated.

OpenEdge Business Process Server

Respond to the following options if you are installing the OpenEdge Business Process Server as a recommended component or as a stand-alone product (downloadable from the Progress Download Center):

- ☐ Retain the default or update the port values provided in the **Business Process Application Server** dialog box. _____.

Note: You cannot enter duplicate port numbers or leave the port fields empty. The range for the port numbers is 1024 to 49151.

-
- ☐ Retain the default values or update the provided value in the **Portal URL** in the **Business Process Server Portal URL** dialog.
-

Note: The **Portal URL** is generated using the host name and the domain name of your machine where you are installing Business Process Server. The valid format is **http://<host_name>.<domain_name>:<port_number>**. For example, <http://localhost.apac.progress.com:18256>.

- ☐ The OpenEdge Business Process Server requires you to create a database for OpenEdge. Proceed as shown in the following table in the **Business Process Server Database Details** dialog box:

Option	Requires you to...
Database Name	Displays oebps because the name of the OpenEdge database you use during development is oebps.db .
OpenEdge Database	Verify the Location for storing the database files.
Database Location	Retain or update the location for storing the database files. Note: We recommend you not to change the Location
Create Database	Select to create the database.

- ☐ Retain the email details provided in the **Email Configuration** dialog box or update values as:

Option	Requires you to...
Outgoing Email Server	Enter the email address used for sending email from Business Process Server.
Administrator Email	Enter the email details of the Business Process Server administrator.
Incoming Server	Enter the email address for receiving email sent to Business Process Server if you want different email servers for incoming and outgoing mail.

OpenEdge product scripts and program modules

- ☐ Determine whether you want all users to have access to OpenEdge product scripts and program modules. Proceed as follows:
 - Type **Y** to place OpenEdge scripts in **/usr/bin** and the destination pathname you defined in the “[Installation and working directories](#)” section on page 440.

- Type **N** to indicate that you want the Installation Utility to place the OpenEdge scripts only in the destination pathname you defined in the [“Installation and working directories”](#) section on page 440.

Identical file exists in the installation directory

- ☐ Identify what you want the Installation Utility to do if an identical file already exists in the installation directory:
 - Type **Y** to delete the file.
 - Type **N** to retain the file.

C

Command and Utility Reference

This appendix provides a reference to a select number of commands and utilities that are useful when performing tasks or understanding information presented in earlier chapters of this document.

The categories these command and utilities refer to are described in the following sections:

- [Administering and configuring Unified Broker products](#)
- [Installing and managing keys and digital certificates](#)

Administering and configuring Unified Broker products

This section highlights the following utilities and commands that you can use to manage various Unified Broker products and their properties:

- [ASBMAN](#)
- [DBMAN](#)
- [Mergeprop](#)
- [NSMAN](#)
- [PROADSV](#)
- [WTBMAN](#)

ASBMAN

Starts, stops, adds AppServer agents, trims AppServer agents, and queries the status for an AppServer instance and its AppServer agent.

Operating system	Syntax
UNIX Windows	<pre>asbman { { -name AppServer-name -all { -kill -start -stop -query -addservers number-to-start -trimservers number-to-trim } [-host host-name -user user-name -user user-name] [-port port-number] } -help }</pre>

-name AppServer-name

This parameter is optional. It specifies the name of an AppServer.

-all

This parameter is optional. This parameter informs the utility to apply the request to all AppServer entries in ubroker.properties. Using this parameter in combination with the *-stop* parameter, you can stop all the AppServers running at a particular time.

-kill

Stops and removes the AppServer from memory, no matter what it is doing.

-start

Starts an AppServer.

-stop

Tells the AppServer to stop itself.

Note: The AppServer stops only after completing any active client requests.

-query

Queries an AppServer for its status.

-addservers number-to-start

Specifies the number of additional servers to start.

-trimservers number-to-trim

Specifies the number of additional servers to trim.

`-host host-name`

Specifies the name of the machine where the AdminServer is running. If a host name is not specified, it defaults to the local host name.

`-user user-name`

Specifies a user name and prompts for a password. A user name and password are required only when you use the `-host` parameter and specify a remote host name. If you specify a remote host name with the `-host` parameter but do not specify a user name with the `-user` parameter, you receive a prompt for a username and password.

Windows supports three different formats for *user-name*:

- A user name as a simple text string, such as "mary", implies a local user whose user account is defined on the local Windows server machine, which is the same machine that runs the AdminServer.
- A user name as an explicit local user name, in which the user account is defined on the same machine that runs the AdminServer, except the user name explicitly references the local machine domain, for example ".\mary".
- A user name as a user account on a specific Windows domain. The general format is *Domain\User*, in which the *User* is a valid user account defined within the domain and the *Domain* is any valid Windows Server, including the one where the AdminServer is running.

`-port port-number`

Specifies the port number of the machine on which the AdminServer is running. If a port number is not specified, it defaults to 20931.

`-help`

Displays command-line help.

`-agentdetail`

Displays the procedure run by an AppServer agent. You can select the specific AppServer agent by specifying the process ID (PID) of an agent process.

`-agentkill`

Stops a specific AppServer agent forcefully. The PID specified to the `asbman` command must be an agent process for the specified AppServer. Use the `kill` command to stop a specific AppServer agent on UNIX. Use the `taskkill` command to stop a specific AppServer agent on Windows XP Professional Edition. This utility is not supported by the Windows Home Edition.

`-agentstop`

Stops an AppServer agent gracefully.

Example Use the following command to display the .r procedure run by an AppServer agent. It displays the port and connection state of an AppServer agent:

```
asbman -name <name> -agentdetail <pid>
```

Example Use the following command to stop an AppServer agent forcefully:

```
asbman -name <name> -agentkill <pid>
```

Example Use the following command to stop an AppServer agent gracefully:

```
asbman -name <name> -agentstop <pid>
```

DBMAN

Starts, stops, or queries a database. Before you can use the DBMAN command-line utility, you must use the OpenEdge Management or OpenEdge Explorer Database Configuration Tool to create the database configuration and store it in the `conmgr.properties` file.

Operating system	Syntax
UNIX Windows	<pre>dbman [-host <i>host-name</i> -port <i>port-number</i> <i>service-name</i> -user <i>user-name</i>] -database <i>db-name</i> [-config <i>config-name</i> -start -stop -query]</pre>

`-database db-name`

Specifies the name of the database you want to start. It must match the name of a database in the `conmgr.properties` file.

`-config config-name`

Specifies the name of the configuration with which you want to start the database.

`-start`

Starts the database *db-name* as defined by the configuration *config-name*.

`-stop`

Stops the database *db-name*.

`-query`

Queries the Connection Manager for the status of the database *db-name*.

`-host host-name`

Identifies the host machine where the AdminServer is running. The default is the local host. If your AdminServer is running on a remote host, you must use the `-host host-name` parameter to identify the host where the remote AdminServer is running.

`-port port-number|service-name`

Identifies the port that the AdminServer is listening on. If your AdminServer is running on a remote host, you must use the `-port port-number` parameter to identify the port on which the remote AdminServer is listening. The default port number is 20931.

`-user user-name`

If your AdminServer is running on a remote host, you must use the `-user user-name` parameter to supply a valid user name for that host. You will be prompted for the password.

Note: For more information, see [OpenEdge Data Management: Database Administration](#).

Mergeprop

Provides a consistent means to manage and maintain the content of property files, either by direct user action or programmatically. Property files store configuration information that identify and control the behavior of various components. The **mergeprop** program is located in the *OpenEdge-Install-Directory\bin*.

Presented through a command-line interface, the **mergeprop** utility is an alternative, fully supported tool by which you can update a property file when either OpenEdge Management or OpenEdge Explorer tool is not available or you choose to use this approach.

The following table presents the mergeprop syntax:

Operating system	Syntax
UNIX Windows	<pre>mergeprop -type file_type [-action operation_type [group_name]] [-target target_file] [-delta delta_file] [-validate] [-nobackup] [-silent] [-recurse]</pre>

See [Table 125](#) for the details about valid values for argument variables.

All of the command switches identified in the previous syntax and presented in more detail in [Table 125](#) can occur in any sequence following the **mergeprop** command.

Command switches and arguments

[Table 125](#) summarizes the syntax elements used with the **mergeprop** command. Refer to the preceding section and the “[Mergeprop utility overview](#)” section on page 291 for additional descriptive information.

Table 125: Command line input to the mergeprop command

Switch	Arguments	Notes
<code>-type</code> (required)	<code>ubroker</code> <code>database</code> <code>tools</code> <code>plugin</code> <code>none</code>	Each argument (other than <code>none</code>) implies a specific target file in the <code>properties</code> directory. See the “ File type ” section on page 295.
<code>-action</code> ¹	<code>update</code> <code>create</code> <code>delete</code> <code>list group_name</code> <code>listall group_name</code>	If no action is specified, <code>update</code> is assumed by default. The <code>list</code> and <code>listall</code> actions require an additional argument, the name of the property group to be displayed (for example, <code>ubroker.AS.asbroker1</code>). Do not include the square brackets ([]) that enclose the group name in the <code>ubroker.properties</code> file. On <code>update</code> and <code>create</code> actions, groups listed with no properties in the delta file are ignored.
<code>-target</code> (optional)	Path to the property file to be modified	If you are updating a property file that is in the <code>OpenEdge-Install-Dir/properties</code> subdirectory, you can omit this option. Only use this option when the property file you plan to update exists in a location other than the <code>OpenEdge-Install-Dir/properties</code> subdirectory.
<code>-delta</code>	Path to the <code>delta</code> file containing changes to be made	File containing create, update, or delete changes.
<code>-validate</code>	None	Performs all processing without actually making changes to the target file. This option lets you test for errors.
<code>-nobackup</code>	None	Does not create a backup to the target file before making changes. Unless you invoke this option, mergeprop saves a copy of the original target file in the same directory. The backup copy has a system-generated unique string appended to the name (for example, <code>ubroker.properties (31420040644533)</code>).
<code>-silent</code>	None	Suppresses all messages.
<code>-recurse</code>	None	Lists or deletes all groups, server groups, and configurations associated with the specified database.

1. Command switches can occur in any order following the mergeprop command.

NSMAN

Allows you to start and stop a NameServer and check the operational status of a NameServer that is located on either a local or remote NameServer instance. Unlike OpenEdge Management or OpenEdge Explorer, the NSMAN utility does not support a means to view log files or delete configured NameServer instances.

Operating system	Syntax
UNIX Windows	<pre>{ { -name name-server { -kill -start -stop -query } [-host host-name -user user-name -user user-name] [-port port-number] } -help }</pre>

`-name name-server`

This parameter is required. It specifies the name of the NameServer.

`-kill`

Stops and removes the NameServer from memory, no matter what it is doing.

`-start`

Starts the NameServer.

`-stop`

Tells the NameServer to stop itself.

`-query`

Queries the NameServer for its status.

`-host host-name`

Specifies the name of the machine where the AdminServer is running. If a host name is not specified, it defaults to the local host name.

`-user user-name`

Specifies a user name and prompts for a password. A user name and password are required only when you use the `-host` parameter and specify a remote host name. If you specify a remote host name with the `-host` parameter but do not specify a user name with the `-user` parameter, you receive a prompt for a user-name and password.

Windows supports three different formats for *user-name*:

- A user name as a simple text string, such as "mary", implies a local user whose user account is defined in the local Windows server machine, which is the same machine that runs the AdminServer.
- A user name as an explicit local user name, in which the user account is defined on the same machine that runs the AdminServer except the user name explicitly references the local machine domain, for example "\mary".
- A user name as a user account on a specific Windows domain. The general format is *Domain\User*, in which the *User* is a valid user account defined within the domain and the *Domain* is any valid Windows Server, including the one where the AdminServer is running.

`-port port-number`

Specifies the port number of the machine on which the AdminServer is running. If a port number is not specified, it defaults to 20931.

`-help`

Displays command-line help.

PROADSV

Supports various activities including starting up, shutting down, and querying the status of the current installation of an AdminServer.

Operating system	Syntax
UNIX	<pre>proadsv { { { -start { [-adminport port-number] } stop -query } [-all][[-keepservers][[-port port-number]] } -help }</pre>

-start

Starts the AdminServer.

-admingroup *groups*

Identifies a list of group names separated by a colon.

-adminport *port-number*

Specifies the port number used by the AdminServer for database broker communication. If a port number is not specified, the `adminport` defaults to port 7838.

-f *pluginsFile*

Points to an AdminServerPlugins.properties file by default. If a file is not defined for the `-f`, then this default is used.

-propertyfile *filename*

Database configuration information. The default value is `OpenEdge-Install-Dir/properties/commgr.properties`.

-requireusername

Indicates that at least one user ID is required to be resolved for each AdminServer operation before each operation can be executed.

-stop

Stops the AdminServer.

-query

Displays the AdminServer status.

`-all`

Displays additional information about the servers the AdminServer has started. These additional servers include nameserver, AppServers, and WebSpeed servers. It also lists any running databases that it started.

`-keepservers`

Allows the AdminServer to shutdown itself, while the servers it started continue running.

`-port port-number`

Specifies the listening port number. If a port number is not specified, the port defaults to 20931.

`-user username`

User who has been assigned AdminServer process privileges. The default is the current user.

`-password password`

The password associated with the `-user`.

`-help`

Displays the command-line help.

[Table 126](#) shows several options that you can use with `proadsv` to accomplish the corresponding tasks. Note that the examples use the port number 9999.

Table 126: proadsv command-line options

AdminServer task	Commands	Examples
Start	<code>-start</code>	<code>proadsv -start</code>
Specify the listening port	<code>-port port-number</code>	<code>proadsv -port 9999 -start</code>
Specify the database broker port	<code>-adminport port-number</code>	<code>proadsv -adminport 9998</code>
Stop	<code>-stop</code>	<code>proadsv -stop</code>
Query	<code>-query</code>	<code>proadsv -query</code>
Help	<code>-help</code>	<code>proadsv -help</code>

Notes: The port numbers specified with the `-port` and `-adminport` options must be different.

If you are running multiple AdminServers, you must override both the default `port` and the default `adminport` settings.

WTBMAN

Controls the operation of a configured WebSpeed Transaction Server. The utility allows you to start one or more than one Transaction Servers, query their status, start and stop additional WebSpeed Agents, trim by a certain number of agents, and shut down the Transaction Server (WebSpeed only).

Operating system	Syntax
UNIX Windows	<pre>wtbman { { -name transaction-server-name { -kill -start -stop -query -addagents number-to-start -trimagents number-to-trim } [-host host-name -user user-name -user user-name] [-port port-number] } -help }</pre>

`-name transaction-server-name`

Specifies the name of one or more than one Transaction Servers, you want to modify. You can specify multiple servers by separating them with a colon (:).

`-kill`

Stops and removes the Transaction Server from memory, no matter what it is doing.

`-start`

Starts the Transaction Server.

`-stop`

Stops the Transaction Server.

`-query`

Queries the Transaction Server for its status.

`-addagents number-to-start`

Specifies the number of additional agents to start.

`-trimagents number-to-trim`

Specifies the number of additional agents to trim.

`-host host-name`

Specifies the name of the machine where the AdminServer is running. If a host name is not specified, it defaults to the local host name.

`-user user-name`

Specifies a user name and prompts for a password when logging in to a remote machine. A user name and password are required only when you use the `-host` parameter and specify a remote host name. If you specify a remote host name with the `-host` parameter but do not specify a user name with the `-user` parameter, you receive a prompt for a user name and password.

`-port port-number`

Specifies the port number of the machine on which the AdminServer controlling the WebSpeed Transaction Server is running. If a port number is not specified, it defaults to 20931.

`-help`

Displays command-line help.

`-agentdetail`

Displays the procedure run by a WebSpeed agent. You can select the specific WebSpeed agent by specifying the process ID (PID) of an agent process.

Example Use the following command to display the .r procedure run by the WebSpeed agent. It displays the port and connection state of a WebSpeed agent:

```
wtbman -name <name> -agentdetail <pid>
```

`-agentkill`

Stops a specific WebSpeed agent forcefully. The PID specified to the `wtbman` command must be an agent process for the specified WebSpeed. Use the `kill` command to stop a specific WebSpeed agent on UNIX. Use the `taskkill` command to stop a specific WebSpeed agent on Windows XP Professional Edition. This utility is not supported by the Windows Home Edition.

Example Use the following command to stop a WebSpeed agent forcefully:

```
wtbman -name <name> -agentkill <pid>
```

Example Use the following command that stops the WebSpeed servers, a, b, and c, leaving the remaining servers unaffected:

```
wtbman -name a:b:c -stop
```

Installing and managing keys and digital certificates

This section identifies each OpenEdge utility that allows you to install and manage keys and digital certificates (in key stores and certificate stores) so the components can use them. For Open Clients and clients of Progress Web services, OpenEdge provides utilities for some clients or relies on utilities provided by the client platform to manage the required certificate stores.

The utilities presented in this section are:

- [certutil](#)
- [genpassword](#)
- [mkhashfile](#)
- [pkiutil](#)

certutil

Provides all the functions necessary to install and manage root certificates from any Certification Authority (CA) as entries in the root certificate store of an OpenEdge client machine (located in *OpenEdge-Install-Dir\certs*).

Operating system	Syntax
<div> <div>UNIX</div> <div>Windows</div> </div>	<pre> certutil [-brief -verbose] { [-format { DER PEM }] -display cert-file [-format { DER PEM }] -import cert-file -list [alias ...] -remove alias ... } </pre>

`-brief`

Provides less information or as specified for the function.

`-verbose`

Provides more information or as specified for the function.

`-format { DER | PEM }`

Specifies the certificate format for the `-import` and `-display` functions. The default input format for a certificate is Privacy Enhanced Mail (PEM). Because some CAs issue root digital certificates in a binary format (DER), you must specify `-format DER` to import these certificates.

`-display cert-file`

Displays the digital certificate file information contained in the operating system disk file, *cert-file*. You must specify *cert-file* as a fully qualified operating system file path name. The `-verbose` option displays complete certificate information, and the `-brief` option displays less certificate information for each certificate store entry.

`-import cert-file`

Imports a trusted CA digital certificate from the disk file *cert-file*. The *cert-file* argument must specify a fully qualified operating system file pathname. The function creates a root certificate store entry with a generated alias name and displays that alias name for view (specified by *alias* for other functions of this command).

Note: All root digital certificate store entry alias names are exactly eight hexadecimal characters in length and have a .0 (dot-zero) file extension. All other files in the root certificate store are ignored.

`-list [alias ...]`

Displays a list of certificate store entries identified by each alias name (*alias*). You can specify multiple aliases, but you cannot use wild cards. If you specify no alias, `certutil` displays all entries in the certificate store. The `-verbose` option displays complete certificate information and the `-brief` option displays less certificate information per key store entry.

`-remove alias ...`

Removes one or more certificate store entries that you specify by their *alias*. You cannot use wild cards. Moves each specified certificate store entry into the `backup` subdirectory and overwrites any previous `backup` subdirectory entry with the same alias name.

For more information on managing root certificates in the OpenEdge root certificate store, see [Chapter 9, “Managing OpenEdge Key and Certificate Stores.”](#)

genpassword

Accepts the clear-text value of a password and generates the encoded and encrypted form for the specified password.

Operating system	Syntax
UNIX Windows	<code>genpassword -password <i>text</i> [-verify <i>encrypted-password</i>]</code>

`-password text`

Where *text* is the clear-text value of the real password. When you specify the `-password` option alone, the tool displays a string of characters that represent the encrypted password value. You can then use this value directly to manually specify a required password as a property or command-line parameter value when manually configuring an OpenEdge client or server.

`-verify encrypted-password`

Where *encrypted-password* is the value of the encrypted password. When you specify the `-verify` option, the tool displays a message indicating if the real password and the encrypted password value match one another.

mkhashfile

Provides a simple way to install a root certificate in the OpenEdge root certificate store of a client machine. Such a certificate can be authorized by your own internal-use Certification Authority (CA) or by any CA that can provide you with a PEM-encoded certificate.

Operating system	Syntax
UNIX Windows	<code>mkhashfile PEM-certificate-pathname</code>

PEM-certificate-pathname

Pathname of a PEM-encoded certificate file (typically with a `.pem` extension) containing a root certificate that you want to store in an OpenEdge root certificate store. The command creates a copy of this file with a hashed filename and places it in the `OpenEdge-install-dir/certs` directory of the client machine. The generated filename becomes the alias for the root certificate store entry.

You can use `certutil` to manage the root certificates that you install with this utility. For more information on managing root certificates in the OpenEdge root certificate store, see the “[certutil](#)” section on page 464 and [Chapter 9, “Managing OpenEdge Key and Certificate Stores.”](#)

pkiutil

Provides all of the functions necessary to create and manage key store entries for OpenEdge SSL servers. It creates these entries from pairs of private keys and digital certificates that it stores in the OpenEdge server key store (located in *OpenEdge-Install-Dir\keys*).

Note: You must submit a public-key certificate request that is generated for each new key store entry that you want to create a Certification Authority (CA) with this utility. The CA then returns the necessary server (public-key) certificate for you to import and completes creation of the new key store entry.

Operating system	Syntax
UNIX Windows	<pre>pkiutil [-brief -verbose] { [-format { DER PEM }] -display cert-file [-format { DER PEM }] -import alias cert-file -list [alias ...] [-keysize size] -newreq alias -print alias -remove alias ... }</pre>

-brief

Provides less information or as specified for the function.

-verbose

Provides more information or as specified for the function.

-format { DER | PEM }

Specifies the certificate format for the -import and -display functions. The default input format for a certificate is Privacy Enhanced Mail (PEM). Because some CAs issue public-key certificates in a binary format (DER) you must specify -format DER to import these certificates.

-display cert-file

Displays the digital certificate file information contained in the operating system disk file, *cert-file*. You must specify *cert-file* as a fully qualified operating system file pathname. The -verbose option displays complete certificate information, and the -brief option displays less certificate information for each key store entry.

```
-import alias cert-file
```

Imports a CA-issued SSL server digital (public-key) certificate from the disk file, *cert-file*, pairs it with the `-newreq`-generated private key identified by the specified alias name (*alias*), and places the pair in the key store as a new entry identified by *alias*. The function prompts for the same password used to generate the public-key certificate request for this entry.

```
-list [ alias ... ]
```

Displays a list of key store entries identified by each alias name (*alias*). You can specify multiple aliases, but you cannot use wild cards. If you specify no alias, `pkiutil` displays all entries in the key store. The `-verbose` option displays complete certificate information, and the `-brief` option displays less certificate information per key store entry.

```
[ -keysize size ] -newreq alias
```

Generates a new private/public-key pair and a corresponding public-key certificate request (suitable for submission to a CA), stored under the alias name specified by *alias*, and placed in the *OpenEdge-Install-Dir\keys\requests* directory.

You must specify an *alias* name between 5 and 39 characters long and use only the following characters:

- "0" to "9"
- "a" to "z"
- "A" to "Z"
- "_" and "-"

Note: The character "-" cannot be used as the first character.

The function prompts for a password with a minimum of four characters using any printable ASCII character. You must use this same password later to create and allow access to the key store entry generated from this certificate request.

When `pkiutil` generates the keys and certificate request for this function, by default it generates keys using the RSA asymmetric encryption algorithm with a 1024-bit key size. If you require a different key size, you can specify the number of bits to generate using the `-keysize` option (valid key sizes must be 512, 1024, or 2048 bits).

`-print alias`

Displays the public-key certificate request identified by *alias*.

`-remove alias ...`

Removes one or more entries from the key store that you specify by their *alias*. You cannot use wild cards. Moves each specified key store entry into the `backup` subdirectory of the key store and overwrites any key store entry previously stored in the `backup` subdirectory with the same *alias*.

OpenEdge National Language Support

This appendix provides information about Progress messages in different languages, as described in the following sections:

- [Packaging](#)
- [Directory structure](#)
- [Contents of each directory](#)
- [Implementing regional support](#)
- [International databases](#)
- [Progress messages](#)
- [Environment variables of the SQL client](#)
- [Regional parameter files](#)
- [Progress.ini file and the Windows registry](#)

Packaging

OpenEdge users can select the language for OpenEdge error and informational messages. The OpenEdge message file, `PROMSGS`, is translated into several different languages.

Some languages are shipped with OpenEdge and are selectable from the Language Choice dialog box during the installation. Additional languages are available to download from the Progress Download Center Web site available at <http://www.progress.com/esd>.

Table 127 identifies the `PROMSGS` translations shipped to all OpenEdge users.

Table 127: PROMSGS translations shipped with OpenEdge

Supported languages shipped with OpenEdge Release 11.3	
Czech (CZE)	Polish (POL)
Dutch (DUT)	Portuguese (POR)
English-American (AME)	Portuguese-Brazilian (BRZ)
English-International (ENG)	Spanish (SPA)
French (FRE)	Spanish-Latin American (SPL)
German (GER)	Swedish (SWE)
Italian (ITA)	Chinese-Simplified (SCH)

Table 128 identifies the additional languages in which `PROMSGS` is translated. These languages are available to download from the Progress Download Center at <http://www.progress.com/esd>.

Note: The Web site requires a valid account that your company must establish with Progress Software Corporation to access this information.

Table 128: Supplemental PROMSGS translations available for download

Supported supplemental languages available to download	
Arabic (ARB)	Lithuanian (LIT)
Chinese-Traditional (TCH)	Norwegian (NOR)
Croatian (HRV)	Persian
Danish (DAN)	Romanian (ROM)
Finnish (FIN)	Russian (RUS)
Greek (GRE)	Serbian (SRB)
Hebrew (HBR)	Slovak (SVK)
Hungarian (HUN)	Slovenian (SVN)
Icelandic (ISL)	Thai (TAI)
Japanese (JPN)	Turkish (TUR)
Korean (KOR)	

Directory structure

The *OpenEdge-install-dir*\prolang\Readme file lists the subdirectories of the \prolang directory by language. Also included is helpful information about code-page tables in convmap.dat.

Contents of each directory

The `prolang` directory contains a subdirectory for each national language that you have chosen to install. Each language subdirectory can contain several files. [Table 129](#) highlights and briefly describes the more important file types contained in a language subdirectory.

Table 129: National language file descriptions

Filename	Description
<code>empty.db</code>	An empty, language-specific OpenEdge directory containing databases of various block sizes. The database is initialized with an appropriate code page and collation for your language.
<code>promsgs.lang¹</code>	A translated OpenEdge-related run-time messages file.
<code>lang.pf¹</code>	A file containing the parameters used to start up OpenEdge with the appropriate settings for your region. For example, <code>engus.pf</code> contains parameters associated with English-American (AME). Also, the <code>startup.pf</code> file, which contains conventions used when your OpenEdge installation is started up, is located in this language subdirectory.
<code>lang.df¹</code>	A data definition file that can be loaded into an empty OpenEdge database to create a language-specific database. A database created by loading this file is identical to the empty database provided in this directory. You can use this file to create sort ordering variations in the database. For example, <code>ame88591.df</code> identifies an English-American (AME) data definition file. In the Asian directories, the file is named <code>_tran.df</code> .
<code>progress.ini</code>	A file containing the parameters used to start up OpenEdge with the appropriate regional settings. For example, the Japanese <code>progress.ini</code> contains Japanese font specifications. The <code>progress.ini</code> file is only installed in the language subdirectory that is identified as the default, or primary, language during the installation process. For information about establishing a language choice, see the “Language Choice” help topic in the Windows or UNIX online help.

1. The variable `lang` stands for the language-specific reference.

Implementing regional support

The installation utility requires you to install at least one language.

During the installation you must choose a default language. If you want to change the default language after installing OpenEdge, see [OpenEdge Development: Internationalizing Applications](#) for detailed instructions.

The **International Settings** dialog box of the installation program creates an OpenEdge Startup (`startup.pf`) file to accommodate international conventions such as Date format, Number format, Character set, Collation, and Case.

Once you select the default language, the Installation Program copies the contents of the `DLC\prolang` directory to `OpenEdge-install-dir`. This affects your `empty.db`, `promsgs`, and `progress.ini` files.

See [OpenEdge Development: Internationalizing Applications](#) for more information about the following files:

- `empty.db` in multiple block sizes
- `startup.pf`
- `promsgs`
- `progress.ini`

International databases

As part of the installation media, OpenEdge supplies empty databases that support the language and collation standards of over thirty languages. The databases are located in the `OpenEdge-install-dir\prolang` subdirectories. See [Table 127](#) for the subdirectory name for your language. For example, the empty database that you might use to build a Russian application is

`OpenEdge-install-dir\prolang\rus\empty.db.`

These empty databases provide a database labelled with the appropriate code page and collation table for a language. However, if you are developing applications for a language or region that is not represented in `OpenEdge-install-dir\prolang`, the OpenEdge utility `PROUTIL` can be used to set up a unique database. See [OpenEdge Development: Internationalizing Applications](#) for detailed instructions.

Progress messages

The text used in Progress messages is contained in the `PROMSGS` file. OpenEdge provides various language editions of the `PROMSGS` file in the `OpenEdge-install-dir\prolang` subdirectories that you select during installation.

Note: Throughout the OpenEdge documentation set, Progress messages are also referred to as OpenEdge messages.

Each file has an extension that identifies its language.

To run OpenEdge with a certain language of `PROMSGS`, set the `PROMSGS` environment variable to the appropriate file. For example:

```
PROMSGS=C:\Progress\OpenEdge\prolang\ger\promsgs.ger
```

After you set the `PROMSGS` variable in the `progress.ini` file, you must run `ini2reg` to update the registry.

By default, Progress displays messages from `OpenEdge-install-dir\promsgs`.

File protection

OpenEdge incorporates specific file-protection measures to accommodate files associated with OpenEdge add-on products, which are OpenEdge products released independently of a point or major OpenEdge product release. Add-on products provide functionality that enhances the OpenEdge software product set and ensures that you have the most recent `PROMSGS` files. All OpenEdge products use one centralized method to display Progress messages contained in the `PROMSGS` file. With each OpenEdge add-on product you install, an updated `PROMSGS` file is installed into the destination directory. Add-on installation processes ensure that if the add-on product contains a newer `PROMSGS` file than the associated release, the following activities occur:

- The add-on product's `PROMSGS` file is compared with the product's `PROMSGS` file to determine which of the two files is newer
- The newer file is copied to the OpenEdge directory

Details about the installation and update of `PROMSGS` files

During the OpenEdge installation process, you select the languages that can be used during the product's execution. It is possible to have several translated `PROMSGS` files installed into the OpenEdge destination `path\prolang` subdirectory due to this selection process. During the installation process, the `PROMSGS` files for the language identified as the default language are copied from the OpenEdge destination `path\prolang` subdirectory to the `OpenEdge-install-dir` directory.

The `PROMSGS` files contain the most up-to-date messages at the time the OpenEdge product is released. However, the `PROMSGS` files are constantly being updated. Consequently, add-on products and OpenEdge install service packs that are released after the product release date can contain even more recently updated `PROMSGS` files. As each OpenEdge add-on product is installed, the installation program checks to ensure that the newest copy of the `PROMSGS` file is being used by all products; all products use the centrally located copy of the `PROMSGS` file stored in the `OpenEdge-install-dir` directory.

Procedures to protect `PROMSGS` files from being overwritten

OpenEdge protects `PROMSGS` files and any associated files, and ensures that you always have the most recent `PROMSGS` files. For example:

- A file protection mechanism is part of the installation program and prohibits overwriting any `PROMSGS` file that already exists. If a `PROMSGS` file exists in the local directory, and it is the latest version; then, there is no need to perform any file changes.
- The OpenEdge Installation program supports a versioning scheme that adds date information to the header of the `PROMSGS` file. The install program uses this date information to help determine the latest version of a `PROMSGS` file.

Procedures to ensure `PROMSGS` files are synchronized

In OpenEdge, `PROMSGS` files are considered to be either in synchronization or out of synchronization. These terms reflect the status of the date stamp associated with a `PROMSGS` file when the date in the header of the `PROMSGS` files located in the add-on directory is compared with the date in the header of the `PROMSGS` files currently installed in the `OpenEdge-install-dir` directory.

In OpenEdge, the installation processes are designed to compare and evaluate the date stamp information. A `PROMSGS` file is considered synchronized if, at the conclusion of any product installation process, the OpenEdge installation contains the `PROMSGS` file with the most current, or latest, date stamp. A `PROMSGS` file is considered out of synchronization, and therefore invalid, when the date stamp associated with the `PROMSGS` file does not display the most current date.

[Table 130](#) identifies the general installation sequence that can occur at a customer site when OpenEdge products and add-on products are installed. It illustrates how the `PROMSGS` files are compared, evaluated, and updated to ensure that the `PROMSGS` files are always synchronized.

Table 130: PROMSGS file synchronization process

Install sequence	When ...	Then ...
1.	OpenEdge products are initially installed	The <code>PROMSGS</code> files associated with the languages selected by the user during the install process are installed to the <code>OpenEdge-install-dir</code> directory.
2.	An OpenEdge add-on product is installed	<p>Date stamp information in the header of the existing <code>PROMSGS</code> file in the <code>OpenEdge-install-dir</code> directory is compared with the date stamp information in the header of the add-on product's <code>PROMSGS</code> file.</p> <p>If the <code>PROMSGS</code> file's date is later than the add-on product's <code>PROMSGS</code> file's date, the file is already synchronized and no changes occur.</p> <p>If the <code>PROMSGS</code> file's date is earlier than the add-on product's <code>PROMSGS</code> file's date, the add-on <code>PROMSGS</code> file replaces the existing <code>PROMSGS</code> file.</p>
3.	OpenEdge products are re-installed to add a new product and a new <code>PROMSGS</code> file	<p>These two comparisons and their associated activities occur:</p> <ul style="list-style-type: none"> • If the re-installation process finds that a <code>PROMSGS</code> file exists, the existing <code>PROMSGS</code> file is not overwritten. • If, during the re-installation process, a new language is added, the <code>PROMSGS</code> file associated with that new language is installed into the <code>OpenEdge-install-dir</code> directory.
4.	Another OpenEdge add-on product is installed	<p>The date stamp information in the header of the existing <code>PROMSGS</code> file in the <code>OpenEdge-install-dir</code> directory is compared with the date stamp information in the header of the add-on product's <code>PROMSGS</code> file.</p> <p>If the <code>PROMSGS</code> file's date is later than the add-on product's <code>PROMSGS</code> file's date, the file is already synchronized and no changes occur.</p> <p>If the <code>PROMSGS</code> file's date is earlier than the add-on product's <code>PROMSGS</code> file's date, the add-on <code>PROMSGS</code> file replaces the existing <code>PROMSGS</code> file.</p>

Table 131 illustrates another example of how this process works, using more detailed data for you to review.

The first column of **Table 131** elaborates on the installation sequence outlined earlier in this section. In Step 1, the user initially installs OpenEdge Studio with a `PROMSGS` file for American English. The file header date of this newly installed `PROMSGS` file is 04/14/2007. In Step 2, when the user installs an add-on product, the add-on product installation compares the header date of its American English `PROMSGS` file, 04/15/2007, with the header date of the existing American English `PROMSGS` file, 04/14/2007. Since the header date of the `PROMSGS` file associated with the add-on product is later than the existing `PROMSGS` file, the `PROMSGS` file is updated or synchronized.

This example helps to illustrate the criterion for updating `PROMSGS` files. Only `PROMSGS` files associated with languages that are currently installed in the OpenEdge will be updated by the add-on installation process.

In Step 3, when the user installs another OpenEdge product, such as the OpenEdge AppServer, and identifies the Spanish `PROMSGS` file, the `PROMSGS` file with the date of 04/14/2008 is installed. This latter part of the example illustrates how the `PROMSGS` files can become out of sync per the date information in the respective headers.

Table 131: Example of `PROMSGS` files being out of sync

Installation step order	Install...	And the <code>PROMSGS</code> file is...	Which contains this header date...
1.	A product such as OpenEdge Studio	Installed for American English	04/14/2008
2.	An add-on product	Updated for American English	04/15/2008
3.	A product such as Application Server	Installed for Spanish <code>PROMSGS</code>	04/14/2008

As **Table 131** indicates, the installation of previously non-existing Spanish `PROMSGS` file dated 04/14/2008 into the OpenEdge installation is now out of synchronization with the updated American English `PROMSGS` file dated 04/15/2008, which updated the original American English `PROMSGS` file.

When an additional OpenEdge installation is performed and the OpenEdge Installation program detects that a `PROMSGS` language has been installed that did not previously exist as illustrated by Step 3 in **Table 131**, the OpenEdge installation program displays a message. This message indicates the following:

- The add-on product name that contains the latest `PROMSGS`
- The destination path of the add-on product

The OpenEdge installation message only displays this message when it detects that add-on products have been installed and it reads a new file called `addons`. The `addons` file is a text file defined as a Windows initialization (`.ini`) file. This file is created and/or updated in the OpenEdge destination directory by the add-on installation program. To resynchronize your `PROMSGS` file, you must reinstall your add-on product.

Environment variables of the SQL client

OpenEdge contains the environment variables `SQL_CLIENT_CHARSET` and `SQL_CLIENT_CHARSET_PROMSGS` for SQL clients. You can use these variables to internationalize your applications. These environment variables determine the code page the client uses to display the following:

- Database data from the server
- `PROMSGS` from the server

Notes: You should set `SQL_CLIENT_CHARSET` only if you want clients to use a code page that is different from the code page the client operating system uses. You should set `SQL_CLIENT_CHARSET_PROMSGS` only if you want run-time messages to use a code page that is different from either the code page the client operating system uses, or the code page set by `SQL_CLIENT_CHARSET`.

If you do not set either of these environment variables, then the SQL client code page will correspond to the language of the client operating system.

Code page client uses to display data

To display database data from the server, the client uses the code page set by `SQL_CLIENT_CHARSET`, if you have set this environment variable on the client machine. Otherwise, the client uses the code page of the client's operating system.

If you want to specify an SQL client code page that is different from the client operating system, you can set the `SQL_CLIENT_CHARSET` environment variable to the name of a Progress code page. When you set this variable to a code page, the SQL server converts text data that is sent **from the server to the client** to the code page set by `SQL_CLIENT_CHARSET`. The server also uses this code page when it converts text data that is sent **from the client to the server** to the server code page.

Code page client uses to display PROMSGS from the server

To display `PROMSGS` from the server, the client uses the code page set by `SQL_CLIENT_CHARSET_PROMSGS`, if you have set this environment variable. Or, the client uses the code page set by `SQL_CLIENT_CHARSET`, if you have set this environment variable. If you have set neither of these environment variables, then the client uses the code page of the client's operating system.

If you want run-time messages at the SQL client to use a different code page from either the client operating system or the code page set by `SQL_CLIENT_CHARSET`, you can set the `SQL_CLIENT_CHARSET_PROMSGS` environment variable. When you set this variable to a code page, the SQL server converts run-time messages that are sent **from the server to the client** to the code page set by `SQL_CLIENT_CHARSET_PROMSGS`.

Note: The `SQL_CLIENT_CHARSET_PROMSGS` environment variable applies to `SQLDUMP` and `SQLLOAD`, which are actually SQL applications.

Regional parameter files

A useful technique for controlling an OpenEdge client session or server is to use a parameter file (.pf) with a startup or connection command. OpenEdge provides parameter files that set up OpenEdge sessions appropriately for a wide range of countries. You can use .pf files to specify the correct code-page settings for international applications. The setup of the *install-path\startup.pf* file is based on the installation options that you select.

The international parameter files are located in the *OpenEdge-install-dir\prolang* subdirectories. Parameter files are region- or country-specific rather than language-specific because parameter files set options that can vary from country to country. The *OpenEdge-install-dir\prolang\ger* directory has parameter files for Austria, Germany, and Switzerland to account for the differences among these German-speaking countries.

You should use the parameter file to make sure that the application and database are using the appropriate international settings. Typically, a parameter file for an internationalized application sets the parameters listed in [Table 132](#).

Table 132: Startup parameters for a deployed application (1 of 2)

Parameter	Description
Internal Code Page (-cpinternal)	The code page that OpenEdge uses in memory.
Stream Code Page (-cpstream)	The code page for stream I/O.
Case Code Page (-cpcase)	A case table in the <i>convmap.cp</i> file to use for uppercase/lowercase rules. Case rules are used by the CAPS and LC functions and by the ! formatting character.
Collation Code Page (-cpcoll)	A table in the <i>convmap.cp</i> file to use for collation rules.
Date Format (-d)	The format in which an application displays dates. Specify the format as a three-character string, comprised of the letters d, m, y, in the order that you display the date.
Language (-lng)	The initial value for the CURRENT-LANGUAGE function, which determines from which r-code segment OpenEdge reads character-string constants. Specify the language as a character string in quotes.
European Numeric Format (-E)	OpenEdge interprets and displays a comma as a decimal separator and a period as a thousands separator for numeric values.

Table 132: Startup parameters for a deployed application (2 of 2)

Parameter	Description
Fractional Separator (-numdec)	Specifies the numeric value of the character that represents, in formatted text, a number's decimal point. The default decimal point is a period (.).
Thousands Separator (-numsep)	Specifies the numeric value of the character that represents, in formatted text, the thousands separator in numbers. The default thousands separator is a comma (,).

Note: You can also use parameter files with OpenEdge utilities, for example, `PROSHUT` and `PROUTIL`.

Progress.ini file and the Windows registry

The `progress.ini` file sets up the user interface environment for Progress applications running in Windows and is an important part of deploying a localized application. It controls parts of the environment that vary across locales, and it allows you to specify colors and fonts.

OpenEdge supports the use of the Windows registry, and searches the registry first for system configuration information. However, you can still use an initialization file to ensure that deployed applications are configured correctly and consistently at customer sites. The information from the `.ini` file can be added to the registry upon installation.

Be sure to create and edit the `progress.ini` file on a system configured like the target system on which you intend to run it. For example, Japanese font names probably use Japanese characters. You should edit a `progress.ini` file for use in Japan on a system supporting Japanese.

If you edit the `progress.ini` file, run `ini2reg` to update the registry.

The sections of the `progress.ini` file that can typically affect a localized application are the `[Startup]`, `[WinChar Startup]`, and `[fonts]` sections.

[Startup] and [WinChar Startup]

The `[Startup]` and the `[WinChar Startup]` sections contain OpenEdge environment variable settings. The `[Startup]` section includes the variables for GUI clients, and the `[WinChar Startup]` section includes the variables for character clients, WebSpeed Agents, and the AppServer.

[Table 133](#) lists the environment variables that a typical localized application might need.

Table 133: Environment variables

Environment variable	progress.ini file section	Description
DefaultFont	[Startup]	The default display font.
DefaultFixedFont	[Startup]	The default-fixed display font.
PrinterFont PrinterFont1 PrinterFont2 PrinterFont3	—	The font that the printer uses for the OpenEdge OUTPUT TO PRINTER statement.
PROMSGS	[Startup] [WinChar Startup]	The <code>PROMSGS</code> file that an application should use. For example, for an OpenEdge application to access Swedish translations of Progress error messages, set <code>PROMSGS</code> to <code>c:\Progress\OpenEdge\prolang\swe\promsgs.swe</code> .

[fonts]

The [fonts] section of the `progress.ini` sets the fonts that an OpenEdge application running on that system uses. The default `progress.ini` file that OpenEdge supplies in the United States sets the following fonts:

```
font0=Courier New, size=8
font1=MS Sans Serif, size=8
font2=Courier New, size=8
font3=Courier New, size=8
font4=MS Sans Serif, size=8
font5=MS Sans Serif, size=10
font6=MS Sans Serif, size=8, bold
font7=MS Sans Serif, size=8
```

These font settings might not apply to all the locales where your application will run. Some of the `OpenEdge-install-dir\prolang` directories contain `progress.ini` files with font settings appropriate for that country.

NameServer and NameServer Load Balancing Details

This appendix presents detailed information about the NameServer and NameServer load balancing feature, as outlined in the following sections:

- [NameServer overview](#)
- [Understanding load balancing](#)
- [Understanding server-level and connection-level fault tolerance](#)
- [Configuring OpenEdge NameServer instances](#)

NameServer overview

A *NameServer* is a single process that mediates client connections for a set of Unified Brokers that have registered with it. Any number and type of Unified Broker instance can register with a single NameServer, and each Unified Broker instance can register with exactly one NameServer. The NameServer that a broker instance registers with is the broker's *controlling NameServer*.

Note: Keep in mind that the NameServer is not required. The use of this element will depend on your implementation.

When a Unified Broker instance starts up, it can register with its controlling NameServer by sending its location and other configuration information. The NameServer uses this information to help resolve client connection requests. Part of this registration information is the Application Service that the Unified Broker supports. An *Application Service* is a designation for the particular business function that a Unified Broker provides. For more information on Unified Brokers, see the [“Working with Unified Brokers”](#) section on page 276 and the [“Unified Broker and Name Server relationship”](#) section on page 489.

A NameServer can provide the following services for a Unified Broker product:

- **Location Transparency** — A requesting client does not need to know the network location of a Unified Broker instance. When a client attempts to create a connection to a Unified Broker instance, it first requests the connection from a NameServer to a broker that provides a specified Application Service. The NameServer then locates and assigns a broker to complete the connection that provides the specified Application Service.
- **Server-level fault tolerance and load balancing** — If you have installed the load-balancing option, you can provide server-level fault tolerance, where the NameServer can select from several Unified Broker instances to satisfy a client request. This option also allows you to balance connection load among multiple Unified Broker instances that provide the same Application Service. The NameServer then assigns connections among several Unified Broker instances based on a weight factor that you configure for each instance. Note that any NSMAN command that specifies a username typically also prompts for a password. For complete information about the syntax and options of the NSMAN command-line utility, see the [“NSMAN”](#) section on page 456.
- **Connection-level fault tolerance** — You can also make multiple NameServer instances available to help ensure that at least one NameServer is available even if another fails. In this type of configuration, one of several possible NameServers resolves the connection request. Thus, you can provide connection-level fault tolerance for requesting clients.

For more information on how your Unified Broker product uses NameServers, see your specific product documentation.

Unified Broker and Name Server relationship

This section highlights the role of the Name Server as it specifically affects a Unified Broker.

Application Services

The Application Service that a Unified Broker provides is identified by a list of one or more names that you can optionally specify during broker configuration. Each Application Service name you specify is an arbitrary designation for the business function that the Unified Broker instance provides.

The NameServer maintains a separate Application Service name space for each Unified Broker type. So, an AppServer, OpenEdge Adapter for SonicMQ, WebSpeed Transaction Server, and DataServer instance can each register the same Application Service name with the same controlling NameServer without conflict. However with the load-balancing option, if you have multiple Unified Broker instances of the same type register the same Application Service name with the same controlling NameServer, you must guarantee that each Unified Broker instance provides exactly the same functionality. For AppServers and WebSpeed Transaction Servers, this means providing the same application procedures and database resources for all instances. For DataServers, this means accessing the same database for all instances.

If, for example, you use the same Application Service name to identify functionality on several AppServers, each of which supports different remote procedures and database connections, multiple requests from the same client application are likely to provide inconsistent results.

The default service

You do not have to designate an explicit Application Service for a Unified Broker. Instead, you can specify that the broker supports the *default service*. The default service is a special Application Service designation that supports default client connection requests. Thus, any Unified Broker that supports the default service and is of the appropriate type (AppServer, OpenEdge Adapter for SonicMQ, WebSpeed, or DataServer) can satisfy a connection request from a client that does not specify an Application Service name as part of its connection request.

Configuring NameServer communications

Both clients and Unified Brokers use User Datagram Protocol (UDP) to communicate with NameServers. UDP is an Internet standard, combined network layer, transport layer, and session layer protocol that provides a mechanism for connectionless communications. The connectionless nature of this protocol affords built-in benefits, such as the ability to implement fault-tolerant NameServers for client connections.

To establish a Unified Broker connection, a Unified Broker client must specify the location of the NameServer that provides the connection. To register with a NameServer, a Unified Broker must specify the location of the NameServer where it needs to register. To specify the NameServer location, both components must know the UDP port number (or service name) on which the NameServer is listening and the host address of the machine where it resides.

Specifying NameServer ports and hosts

ABL clients provide this information when they specify the `CONNECT()` method, and AppServer Open Clients provide it using an equivalent Open Client method. You must specify this information for NameServers and other Unified Broker components in OpenEdge Management or OpenEdge Explorer or directly in the Unified Broker properties file (`ubroker.properties`). For more information, see [Chapter 10, "Configuration."](#)

If the component uses a UDP service name rather than a port number, you must also ensure that the `services` file on the component host (or Network Information Services (NIS), if used) properly defines the UDP service name.

Editing the services file

The `services` file stores the service name, port number, and protocol for various services on the network. For each NameServer that the component accesses, the `services` file on the component host must specify a service name associated with the NameServer UDP port number (default, 5162). Thus, if the component connects using the service name `namesv`, you might enter the service name definition for the `services` file extract shown in [Figure 8](#).

<i>Server name</i>	<i>Port number</i>	<i>Protocol</i>
namesv	5162	udp
db1sv	2501	tcp
db2sv	2502	tcp
db3sv	2503	tcp

Figure 8: Sample Unified Broker client services file

Understanding load balancing

Load balancing is a feature that allows client connection requests to be distributed among multiple Unified Broker instances that support the same Application Service. Load balancing is a NameServer option that comes installed with some products (for example, the WebSpeed Enterprise Transaction Server) or that you must install as an option with others (for example, the AppServer). If you have load balancing with your product, the NameServer assigns client connections to the appropriate Unified Broker instances based on weight factors that you specify.

If the weight factor that you specify for each Unified Broker instance is appropriate in relation to the others, the effect is to assign more connections to broker instances with greater resources, and thus to balance the connection load among all the instances. You can set the load-balancing weight factor for each Unified Broker instance in OpenEdge Management or OpenEdge Explorer or by editing the `priorityWeight` property in the `ubroker.properties` file.

Percentage weight factors

Properly specified, weight factors give some sense of the amount of work that an individual Unified Broker instance can handle. For example, [Table 134](#) shows the effect of weight factors specified for three Unified Broker instances registered to support the same Application Service.

Table 134: Weight factors based on percentage

Unified Broker name	Weight factor	Percent of time selected
AS1	20	20
AS2	20	20
AS3	60	60

The selection algorithm used by the NameServer guarantees that AS1 and AS2 are each selected 20% of the time and AS3 is selected 60% of the time. Thus, if the sum of weight factors for all Unified Broker instances that support the same application adds up to 100, each weight factor specifies the exact percentage of time that the NameServer selects the given Unified Broker instance over time.

Arbitrary sum weight factors

You can specify arbitrary weight factors as any sum of values (not necessarily 100), but the weight of each is always proportional to the sum, as shown in [Table 135](#).

Table 135: Weight factors based on arbitrary sums

Unified Broker name	Weight factor	Percent of time selected
AS1	2	2/7
AS2	2	2/7
AS3	3	3/7

Fail-over weight factor

You can also specify a fail-over weight factor of zero (0) for a Unified Broker instance that you want to accept connection requests when the NameServer finds no other Unified Broker instance available in the pool.

Understanding server-level and connection-level fault tolerance

By default, a Unified Broker instance relies on a single controlling NameServer to resolve client connection requests and a single Unified Broker instance to provide services to the client. You can configure the controlling NameServer so that multiple NameServer instances are available to resolve any client connection request, thus providing connection-level fault tolerance. If your product supports load balancing, you can also configure a single NameServer to resolve each connection request using multiple Unified Broker instances that support the same Application Service, thus providing server-level fault tolerance. Figure 9 shows the relationship between these configuration options.

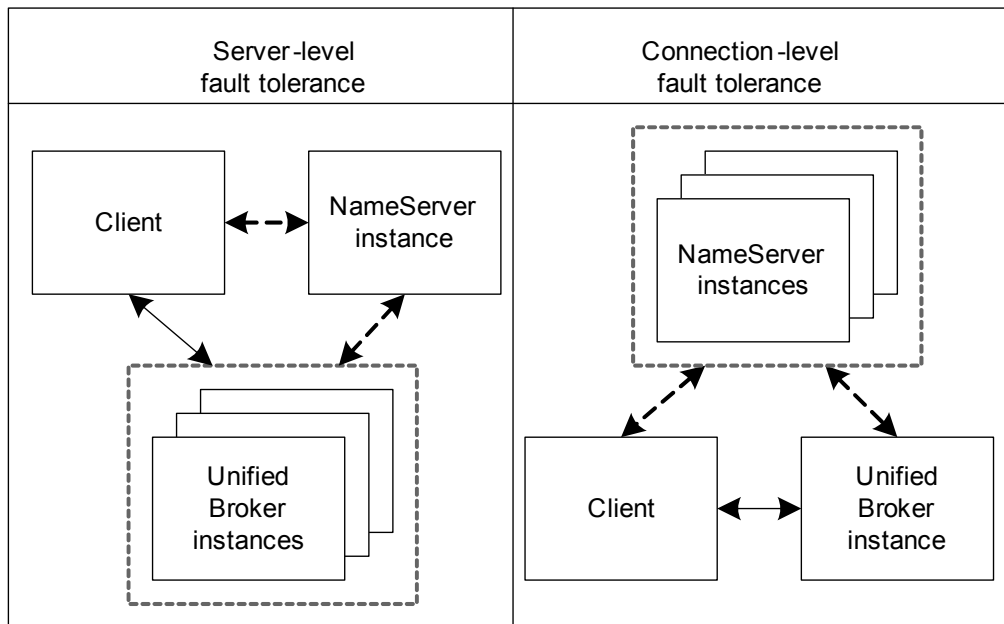


Figure 9: Server-level and connection-level fault tolerance

These two levels of fault tolerance operate as follows:

- Server-level fault tolerance** — Allows multiple Unified Broker instances to register with a NameServer for the same Application Service. A client requesting a connection is connected to one of several registered Unified Broker instances that the NameServer determines are available to provide the specified Application Service. If appropriate weight factors are specified, the NameServer also balances connection load among the several broker instances. For more information on load balancing, see the [“Understanding load balancing”](#) section on page 491.
- Connection-level fault tolerance** — Allows you to configure a collection of NameServers that work together to resolve a client connection request. You can use two different techniques, individually or together, to implement a fault-tolerant NameServer collection. This section describes these techniques and assumes that you are familiar with the documentation on using the NameServer with your Unified Broker product.

You can apply server-level and connection-level fault tolerance individually or together to achieve the level of fault tolerance that your application requires.

Connection-level fault tolerance

Connection-level fault tolerance enables a client (AppServer client, SonicMQ Adapter, WebSpeed Messenger, or DataServer client) to have its connection request satisfied by any NameServer from a set of related NameServers. You can configure NameServers for fault-tolerant operation using two different techniques, and you can use these techniques independently or together:

- **NameServer replication** — Where you configure multiple NameServer instances within a single subnet on different machines to listen on the same UDP port. Clients send connection requests and Unified Brokers send registration requests to all NameServer instances using UDP broadcasting. Using UDP broadcasting, the registration information from brokers is replicated on each NameServer that is listening (hence **NameServer replication**). Similarly, each client connection request is sent to each of the replicated NameServers.
- **NameServer neighbors** — Where you configure multiple NameServers on machines located in one or more subnets so that an initial NameServer instance receives the client connection request. If this initial NameServer cannot resolve the request, it passes the request on to a specified list of NameServer neighbors. These NameServer neighbors then attempt to resolve the connection request. Each NameServer neighbor represents the controlling NameServer for a separate Unified Broker instance.

Using either or both techniques, a client can receive multiple responses from multiple NameServers. The client uses the first response that indicates that the requested Application Service was found. A client only receives a connection error if all NameServers that respond indicate that the Application Service cannot be found.

In general, you can combine NameServer replication with NameServer neighbors to provide connection-level fault tolerance across an entire network. The sections that follow describe how to implement connection-level fault tolerance using these techniques.

Using UDP broadcasting

As described earlier, UDP is a connectionless protocol. This feature allows you to configure the following two types of communications with a NameServer:

- **Host request** — The client or Unified Broker sends a message directly to a NameServer residing on a specific host and listening on a specific port. The IP address represents the actual network location of a specific host. Only the NameServer on the specified host and listening on the specified port receives the message.
- **Broadcast request** — The client or Unified Broker sends a message specifying the UDP broadcast address of the NameServer host and the UDP port number on which the NameServer is listening. The UDP broadcast address represents the entire subnet where a host is located, and you can determine this address using the appropriate operating system commands from any host on the subnet. When a client or Unified Broker sends a UDP broadcast request, every NameServer on any host in the subnet that is listening on the specified port receives the message.

UDP broadcasting insulates the client and Unified Broker from having to know the exact host location of the NameServer. If there is some reason that you need to move the NameServer to a different machine in the same subnet, you can safely do it without having to change your client application or your Unified Broker configuration.

Figure 10 shows a client and Unified Broker using UDP broadcasting to communicate with the NameServer. Using the UDP broadcast address, 172.20.255.255, this client and Unified Broker can communicate with a NameServer running on any host in the 172.20 subnet.

Thus, you can use UDP broadcasting to support location transparency for a single NameServer. However, as Figure 10 implies, you can also use UDP broadcasting as the basis to support fault-tolerant NameServers using NameServer replication.

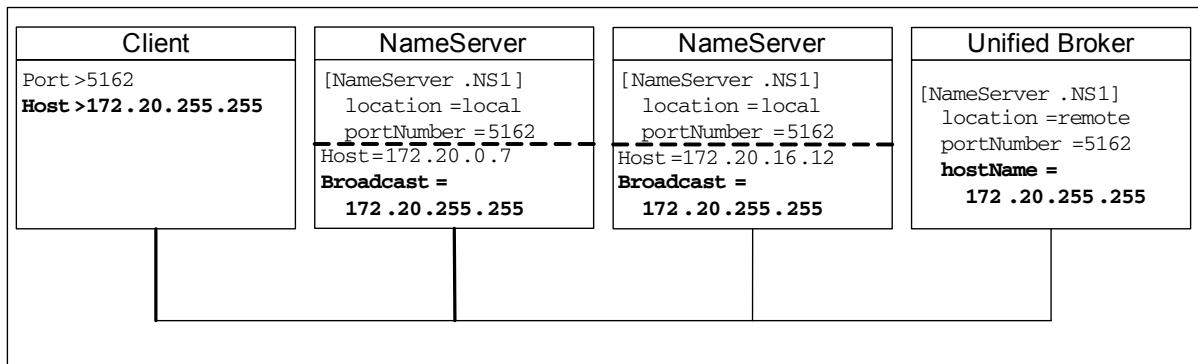


Figure 10: NameServer replication

Using NameServer replication

UDP broadcasting supports NameServer replication by allowing a client or Unified Broker request to be received by multiple NameServers listening on the same UDP port and configured on different machines within the same subnet. Because every host on a subnet receives every broadcast request, one or more of these hosts can support a NameServer that receives and handles the same messages. This provides fault tolerance for both a client connection request and a Unified Broker registration request.



To configure and use replicated NameServers:

1. Run each NameServer instance on a separate host located within the same subnet.
2. Configure each NameServer instance to listen on the same UDP port.
3. Configure each client application to send its connection request and each Unified Broker to send its registration request using the subnet UDP broadcast address instead of the NameServer host address.

There is one broadcast address for each subnet. Using this address and the specified UDP port number, a client or Unified Broker sends a single request that is recognized by every NameServer listening on that port in the subnet.

Figure 10 shows a client, a Unified Broker, and two replicated NameServers. The NameServer configurations shown for NameServer NS1 (above the dotted line) appear as they might in the `ubroker.properties` file for each host.

In Figure 10, one NameServer is located on a machine with the IP address 172.20.0.7 and another is located on a machine with the IP address 172.20.16.12. Both NameServers listen on UDP port 5162. The UDP broadcast address for these NameServers is 172.20.255.255. The Unified Broker is configured to register with a controlling NameServer remote from the Unified Broker machine using the UDP broadcast address 172.20.255.255 as the `hostName`. When the Unified Broker registers with its controlling NameServer using the UDP broadcast, it registers with both replicated NameServers. Similarly, when the client broadcasts its connection request using 172.20.255.255 as the NameServer host name, both replicated NameServers receive the request. The client uses the Unified Broker connection returned by the first NameServer that responds.

Note that if the NameServer at IP address 172.20.0.7 moves to a different host on the subnet, for example, with IP address 172.20.16.5, neither the client application nor the Unified Broker configuration has to change.



To configure and use NameServer replication:

1. Install the NameServer on each host within a single subnet where you want to replicate a NameServer configuration.
2. Configure each replicated NameServer to listen on the same UDP port number.
3. Determine the UDP broadcast address for the subnet where the NameServer hosts reside. For more information, see the [“Determining the broadcast address”](#) section on page 497.
4. Configure each Unified Broker instance (AppServer, SonicMQ Adapter Broker, WebSpeed Transaction Server, or DataServer) to use a controlling NameServer as follows:
 - **Location** — Remote
 - **Host name** — The UDP broadcast address that you determined from Step 3
 - **Port number** — The UDP port number that you specified in Step 2
5. Provide connection parameters to the client (AppServer, DataServer, or WebSpeed) that specify the required Application Service name, the broadcast address from Step 3, and the UDP port number that you specified in Step 2.

Determining the broadcast address

You can determine the broadcast address of a UNIX machine by using the `netstat` and `ifconfig` commands, as in the following example:

```
$ netstat -i
Name  Mtu  Net/Dest  Address  Ipkts  Ierrs Opkts  Oerrs Collis Queue
lo0   8232 loopback  localhost  771334  0      771334  0      0      0
le0   1500 bali      bali      15069970 286170 10019158 1      302211 0
$ ifconfig le0
le0: flags=863<UP,BROADCAST,NOTRAILERS,RUNNING,MULTICAST> mtu 1500
      inet 172.20.0.7 netmask ffff0000 broadcast 172.20.255.255
```

This example shows that the IP address for bali is 172.20.0.7, and its broadcast address is 172.20.255.255.



To determine the broadcast address in Windows:

1. Enter the `ipconfig` command in the console, as shown in the following example:

```
C:\>ipconfig

Windows IP Configuration

Ethernet adapter CE2XPS1:

    IP Address. . . . . : 172.18.103.44
    Subnet Mask . . . . . : 255.255.0.0
    Default Gateway . . . . . : 172.18.0.19
```

2. For each bit in the Subnet Mask that has a value of 0, convert the corresponding bit in the IP Address to 1.

Note that the IP Address and Subnet Mask are composed of four dot-separated decimal numbers and each decimal number represents an 8-bit binary number. Also note that the decimal number 255 is 11111111 in binary.

In this example, the last two decimal digits of the Subnet Mask are zeros. Since the corresponding bits in the IP Address must be converted to 1, the last two decimal numbers of the IP Address should be 255. Therefore the broadcast address is 172.18.255.255. (For more information on determining broadcast addresses, consult with your network administrator.)

Using NameServer neighbors

In a typical environment where UDP broadcasting is used, there is at least one NameServer in each subnet where a Unified Broker exists. A client application that wants to make use of Unified Brokers in each subnet can make a separate connection request to the appropriate controlling NameServer for each Unified Broker. However, NameServer neighbors allow the client to make all of its connection requests using a single NameServer address.

NameServer neighbors are alternate NameServers that you specify as part of a NameServer configuration. When a NameServer receives a connection request from a client that it cannot resolve, it automatically passes the request to the specified NameServer neighbors to attempt the resolution.

NameServer neighbors support client applications in a similar way to replicated NameServers in that the client uses the first response returned by a NameServer, indicating that the requested Application Service was found. However, unlike replicated NameServers, each NameServer neighbor is typically the controlling NameServer for a separate and distinct Unified Broker configuration that might not support the same Application Services as the others.

You can configure each NameServer neighbor as a NameServer with its own set of NameServer neighbors. Thus, you can link NameServer neighbors to other NameServer neighbors for an arbitrary level of depth. You can also replicate both initial NameServers and NameServer neighbors for maximum fault tolerance.



To configure NameServer neighbors:

1. Define the NameServer neighbors as local NameServer instances on your network by using OpenEdge Management or OpenEdge Explorer or by editing the `ubroker.properties` file. These neighbor NameServers can be defined in the same or different subnets in your network.
2. Specify their names as NameServer neighbors when you define the NameServer in OpenEdge Management or OpenEdge Explorer, or assign the names in a comma-separated list as the value of the `neighborNameServers` property for the definition in the `ubroker.properties` file.

Figure 11 shows how to configure NameServer neighbors and combines NameServer replication with NameServer neighbors.

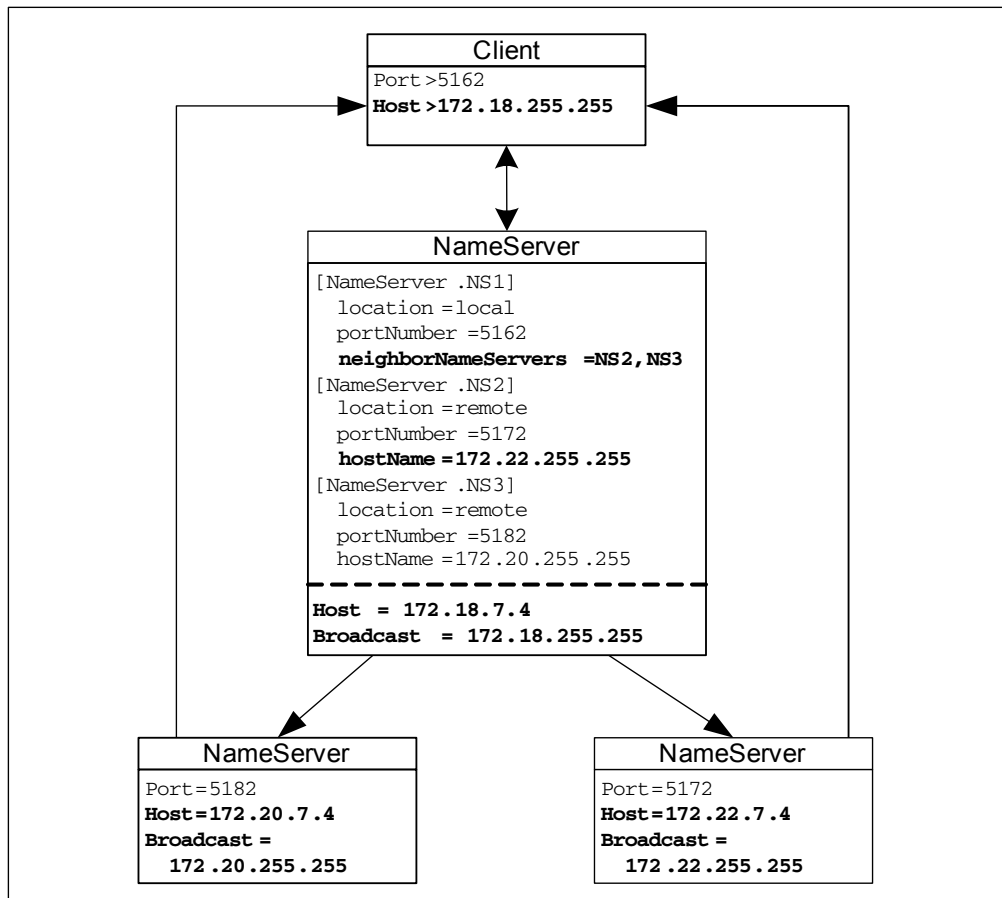


Figure 11: NameServer neighbors

In Figure 11, if a client requests a connection to a Unified Broker that supports the Inventory Application Service using the 172.18.255.255 broadcast address, the NameServer at host 172.18.7.4 receives the request. If this NameServer has a Unified Broker that registered for the Inventory Application Service, it returns the location of that Unified Broker back to the client. If it does not have a Unified Broker that registered for the Inventory Application Service, this NameServer forwards the request to its neighbors, specified using the broadcast addresses 172.20.255.255 and 172.22.255.255. In this instance, the NameServers at hosts 172.20.7.4 and 172.22.7.4 receive the request.

Note: If you replicate these NameServers, all of the replicated NameServers in each subnet receive the request.

Neither of these NameServers has neighbors of their own, so both of them send a response back to the client. It does not matter if one of the NameServers does not know about the requested Application Service. The client uses the first positive acknowledgement from a NameServer and disregards the rest. The client application only receives an indication that the Application Service was not found if all responding NameServers indicate that the Application Service was not found.

Note that while NameServer neighbors provide the most benefit when using UDP broadcast, there is no requirement to do so. The `hostName` properties for NS2 and NS3 in [Figure 11](#) can explicitly specify 172.20.7.4 and 172.22.7.4. You might want to use NameServer neighbors without broadcasting when you must tie together preconfigured NameServers, but where the performance implications of broadcasting outweigh the benefits. For more information, see the [“Performance implications of broadcasting”](#) section on page 500.

Performance implications of broadcasting

When you use UDP to send a specific host request, only the specified host examines the message to determine what port it was sent to and whether an application (like the NameServer) is running and listening to that port on the host. However, when you use UDP broadcasting, either for NameServer replication or to provide location transparency for a single NameServer, every host in the specified subnet examines the message for this same information.

Thus, using UDP broadcast might have a significant impact on the performance of your network if you have a large number of client applications that frequently connect to Unified Brokers. In deciding whether to use UDP broadcasting, you must weigh the benefits of location transparency for a single NameServer or replication of multiple NameServers against the impact on your Unified Broker and network performance.

Configuring OpenEdge NameServer instances

You can use OpenEdge Management or OpenEdge Explorer to configure NameServer and Unified Broker instances, locally or remotely for both Windows and UNIX hosts. If you plan to configure instances directly on a UNIX host, you must edit the properties file (`ubroker.properties`) for each NameServer and Unified Broker instance directly on the host.

Note: The properties file that comes installed with your Unified Broker product includes one sample NameServer and Unified Broker instance for each type of Unified Broker that you can use as a guide. This section addresses using the NameServer and the Unified Broker instance. However, keep in mind that using the optional NameServer will depend on your company's implementation.

Downloading NameServer executables

If you need a NameServer executable, for example, for a different deployment platform, you can download it from the Progress Download Center available at <http://www.progress.com/esd>. Follow the instructions for the NameServer executable to download the OpenEdge NameServer executable for your OpenEdge release and platform.

Note: The Progress Download Center is located at <http://www.progress.com/esd>. You must have a valid user name and password to download products from this site. Contact a Progress Customer Service Representative to set up your Download Center account.

Order of configuration

To configure a NameServer and Unified Broker instance, you generally configure components in the following order:

1. Controlling NameServer and any replicated or neighbor NameServers
2. Unified Broker product components

In OpenEdge Management or OpenEdge Explorer, you must have an initial configuration for the controlling NameServer instance to identify it when you configure your Unified Broker product instance. Editing the properties file, you can configure these components in any order. Whatever order you configure these components, you must have the controlling NameServer configured and running before clients can access your Unified Broker instance.

Configuring and using NameServer instances

You can configure two types of NameServer instances, determined by their functions, as the controlling NameServer for a Unified Broker instance:

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

When you configure a local NameServer instance, you can set all properties for the NameServer. When you configure a remote NameServer instance, you can only set its location (host and port) properties to identify the local NameServer instance that it references. When you want to start, stop, or obtain status on a running NameServer, you must always perform these actions on a local instance. You cannot start, stop, or obtain status on a remote NameServer instance.

How Unified Brokers use NameServer instances

To use a local NameServer instance as its controlling NameServer, a Unified Broker instance must run on the same machine where the local NameServer instance runs. Remote NameServer instances provide a way of having multiple Unified Broker instances use a controlling NameServer that runs on a different machine from the Unified Broker instances.

Whether local or remote, the NameServer instance that you define as the controlling NameServer must be defined on the same machine as the Unified Broker instance it controls. If the controlling NameServer instance is local, it runs on the same machine as the Unified Broker. If the controlling NameServer instance is remote, it references a NameServer running locally on a machine that is remote from the Unified Broker.

Thus, any remote NameServer instance you define must have a corresponding local NameServer instance defined on the machine where it runs. You must define one such remote NameServer instance on each remote machine where a Unified Broker instance references this same corresponding local NameServer instance as its controlling NameServer.

NameServer instances and client connections

Unified Broker clients do not use local and remote NameServer instances. Clients must direct all connection requests to a NameServer on the machine where it runs, that is, to a NameServer where it is defined as a local instance.

Configuring the NameServer in OpenEdge Management or OpenEdge Explorer

You can use OpenEdge Management or OpenEdge Explorer to define and configure a NameServer instance. (See the OpenEdge Management or OpenEdge Explorer online help for detailed information.) When you configure a NameServer instance, you can do the following things in each of the property categories:

- **Location** — Sets port numbers and connection types.
- **General** — Sets the working directory, even when the NameServer starts automatically, and when the NameServer unregisters brokers.

- **Logging setting** — Sets how the NameServer logs events.
- **Advanced features** — Specifies one or more NameServers from the already-configured NameServers to serve as Neighboring NameServers. These are the NameServers that provide connection-level fault tolerance for a Unified Broker application. For more information, see [Appendix E, “NameServer and NameServer Load Balancing Details.”](#)
- **Environment variables** — Sets environment variables for NameServer execution. Windows users should refer to [Chapter 7, “Working in the OpenEdge Environment in Windows,”](#) and UNIX users should refer to [Chapter 8, “Working in the OpenEdge Environment on UNIX”](#) for more information.

Starting and managing a NameServer using OpenEdge Management or OpenEdge Explorer

When you start a NameServer instance with OpenEdge Management or OpenEdge Explorer, a brief status displays for the selected NameServer in the right pane showing that the NameServer is running.

Using OpenEdge Management or OpenEdge Explorer you can also invoke the following management functions for the running NameServer instance:

- Stop the NameServer
- Check the operational status of the NameServer
- View the log file for the NameServer
- Delete the NameServer instance

Note: Before you can delete a NameServer instance, you must stop the NameServer and make sure no running Unified Broker instance still references the NameServer as its controlling NameServer.

For more information on invoking NameServer management functions in OpenEdge Management or OpenEdge Management or OpenEdge Explorer, see the OpenEdge Management or OpenEdge Explorer online help.

Configuration Models

This appendix provides information about different configuration models you can reference and the details about running OpenEdge installations in a network environment, as described in the following sections:

- [Shared-memory configurations](#)
- [Client/server configurations](#)
- [Client/server and OpenEdge AppServer in the network environment](#)
- [Preparing to run OpenEdge on a TCP/IP network](#)

Shared-memory configurations

Shared memory is an area in system memory that multiple users can access concurrently. OpenEdge keeps resources shared by all database users in shared memory and lets multiple servers access those resources efficiently. Optionally, additional asynchronous I/O processes can off load I/O operations from each server, further improving resource utilization and performance.

Local clients running multi-user OpenEdge can access database resources directly, rather than through a database server. This eliminates client/server message exchange and task-switching overhead. Database requests do not have to be queued until a server can process them. Local direct-access clients are known as *self-service clients*.

To run OpenEdge over a network, you need information regarding network-related system files, network configuration, and the startup parameters required to start remote clients. For more information about the network files and configuration, see the [“Client/server and OpenEdge AppServer in the network environment”](#) section on page 512 and the [“Preparing to run OpenEdge on a TCP/IP network”](#) section on page 518. For information about starting remote clients, see [Chapter 11, “Starting and Running OpenEdge.”](#)

Shared-memory architecture

Figure 12 shows the shared-memory OpenEdge architecture.

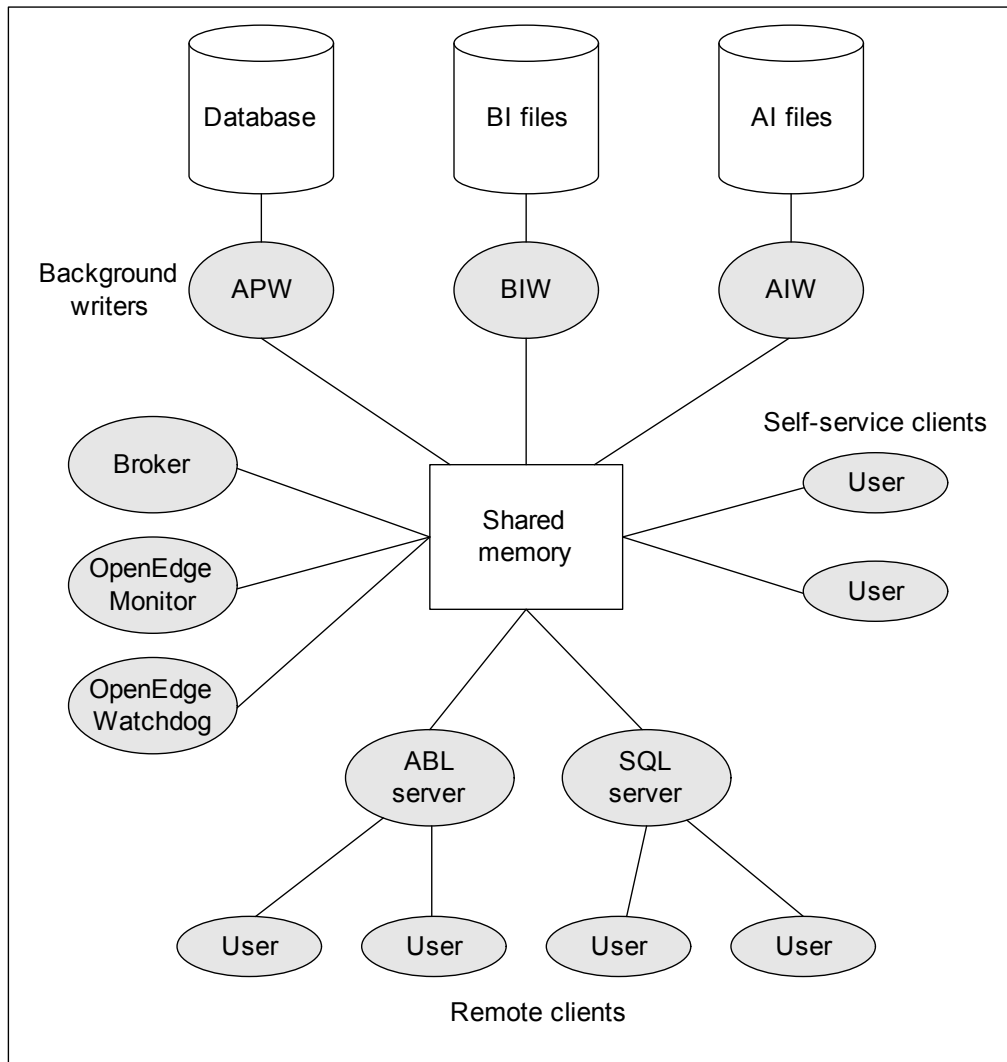


Figure 12: Shared-memory OpenEdge architecture

The sections that follow explain the components of the architecture.

Broker

The initial database server process is identified as the *broker* (`_mprosrv`). The broker process manages shared resources and starts servers for remote users, as needed.

OpenEdge Database Monitor utility

The OpenEdge Database Monitor utility (`_dbagent`) displays performance and usage information about database status and activity.

For more information about the Database Monitor utility, see the description of the `PROMON` utility in [OpenEdge Data Management: Database Administration](#).

OpenEdge Watchdog utility

If a process terminates improperly, it can maintain a lock on a record or shared-memory structure. This can impact database concurrency. The OpenEdge Watchdog utility detects processes that have terminated improperly and cleans up after them.

At regular intervals, the Watchdog utility checks for processes that have terminated unexpectedly. If it finds one, it releases any locks or shared-memory structures that the process might hold.

The Watchdog utility checks for inactive processes approximately once every 10 seconds. It also checks for self-service clients that are no longer active, releases all the appropriate record locks, backs out of any live transactions, and releases any shared-memory locks. If a server process terminates unexpectedly, the Watchdog utility disconnects and cleans up the server's remote clients.

For more information about the Watchdog utility, see the description and other details about the `PROWDOG` utility in [OpenEdge Data Management: Database Administration](#).

Background writers

The OpenEdge Enterprise RDBMS offers three background writer processes that improve performance. These processes continually perform certain housekeeping functions in the background. Because these functions are performed regularly by the dedicated background writer processes, client and server processes rarely have to wait for these functions to be performed.

The three types of background writers, asynchronous page writers, before-image writers, and after-image writers, are described in the [“Processes on Windows and UNIX platforms”](#) section on page 195 and in the [“Processes on UNIX platforms”](#) section on page 200. The AdminService starts the background writers if the AdminService has been configured to do this by OpenEdge Management or OpenEdge Explorer. For more information about background writers, see [OpenEdge Data Management: Database Administration](#).

Client/server configurations

Wherever it runs, multi-user OpenEdge functions in a client/server architecture. On a single machine, OpenEdge provides multi-user access to a database by using a separate client process for each user. In a client/server configuration, one or more clients access the database through a server. The server provides a connection to the database through the shared memory. While separate and distinct, the OpenEdge client and server processes compete for the same machine resources.

In client/server configurations, the client application and the database server are separate processes. Client processes can be local or remote.

The OpenEdge user interface and OpenEdge applications execute in the client session, sending requests to the OpenEdge server. The OpenEdge server accesses the database on behalf of each client session.

Terminology

This section introduces the terminology used to describe client/server configurations.

Application workstation

An *application workstation* is any node that runs one or more OpenEdge clients. Depending on its configuration, an application workstation might run local clients and servers as well.

Database server machine

A *database server machine* is any node that runs one or more OpenEdge servers for local or remote OpenEdge clients.

Network file server

A *network file server* is any node that provides shared services such as file, printing, and security services to other nodes, including application workstations and database server machines. A network file server usually provides these services by allowing other nodes to access its local files and printers as if they were local. For example, OpenEdge clients can run application procedures and OpenEdge servers can access database files stored on a remote network file server.

A network operating system (NOS) is a network environment that includes one or more network file servers that provide a common set of resource sharing and security services to other nodes. A network file server usually runs the *kernel* of an NOS, the program that controls access to shared network resources. Depending on its operating system, a network file server might also run one or more OpenEdge database clients and servers.

Although Progress Software Corporation recommends that you store the database on a disk locally attached to the database server machine, you can store the database on a network file server. Clients can access shared application code and communicate with the database server. However, depending on your application and network environment, you might lose database integrity.

Note that OpenEdge often runs in local area networks (LANs) that have no network file servers. On these LANs, application workstations can only access locally stored procedures, and database server machines can only access locally stored databases. However, the application workstations and database server machines can communicate with each other as remote processes.

Single-process database server machine

A *single-process database server machine* is a node that runs only one server process for each database, providing access to that database for self-service clients only.

Multi-process database server machine

A *multi-process database server machine* is a node that runs multiple server processes for each database, providing multiple data paths to the database. Each server queues and runs requests for one or more clients. A separate broker process starts a new server for each additional client (or set of clients, in specified increments) that access the database. For more information on server machine configurations, see the [“Shared-memory configurations”](#) section on page 506.

You can dedicate all the resources of a database server machine to run database servers. However, depending on your application and operating system, you can also run local clients and remote clients for other database server machines.

Simple client/server configurations

[Figure 13](#) shows a simple client/server configuration, where the client and server components both run on a single system.

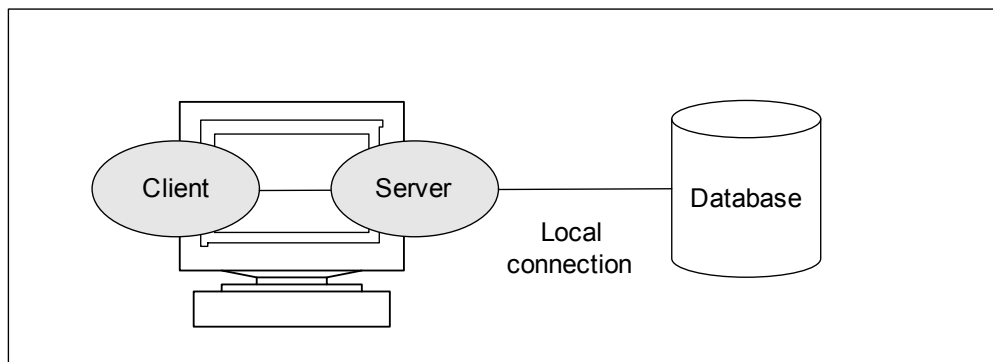


Figure 13: Simple client/server configuration

Figure 14 shows a multiple system client/server configuration. In this configuration, the server runs on the system where the database resides. The clients run on remote systems, accessing the database through the server system.

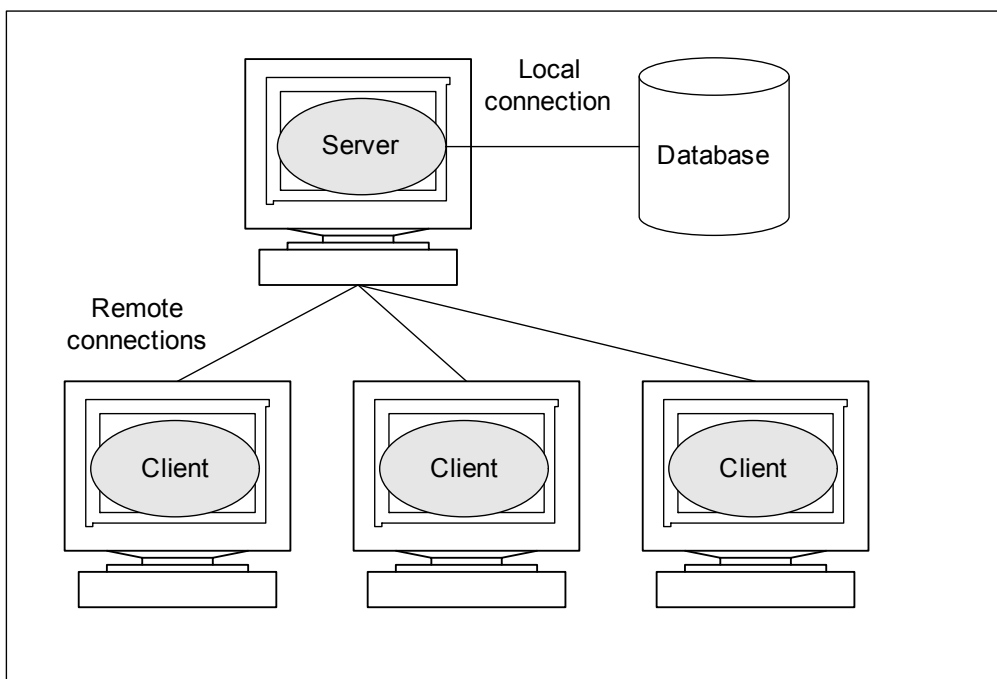


Figure 14: Multiple system client/server configuration

Client/server and OpenEdge AppServer in the network environment

The OpenEdge client/server architecture fits naturally into a network environment, allowing clients and servers to run together in many different (heterogeneous) hardware and operating system environments. On a network, the OpenEdge client and server processes are distributed to separate nodes where they communicate through a common network protocol. Some nodes run client processes, while others run server processes. One advantage of this is that adding users or databases has minimal impact on the machine resources used by others. Each has its own resources devoted only to its client or server tasks. Another advantage is that a single OpenEdge application can take advantage of the strengths of a multi-machine, multi-operating system environment, without regard to differences in file resources on the separate machines. Remote OpenEdge clients and servers interact transparently, regardless of the type of machine environment in which they run. The result is a cooperative application environment with many more possibilities for expansion.

OpenEdge TCP network support

OpenEdge allows client and server operation among Windows systems that communicate using TCP/IP. In an OpenEdge AppServer configuration, the client connection to the application server is always TCP. The OpenEdge AppServer supports all of the client/server network types for the connection of the application server to the database server.

Figure 15 shows the simplest OpenEdge network configuration—a database server machine and an application workstation. Although the figure shows only one database server machine and workstation, there can be more than one of each.

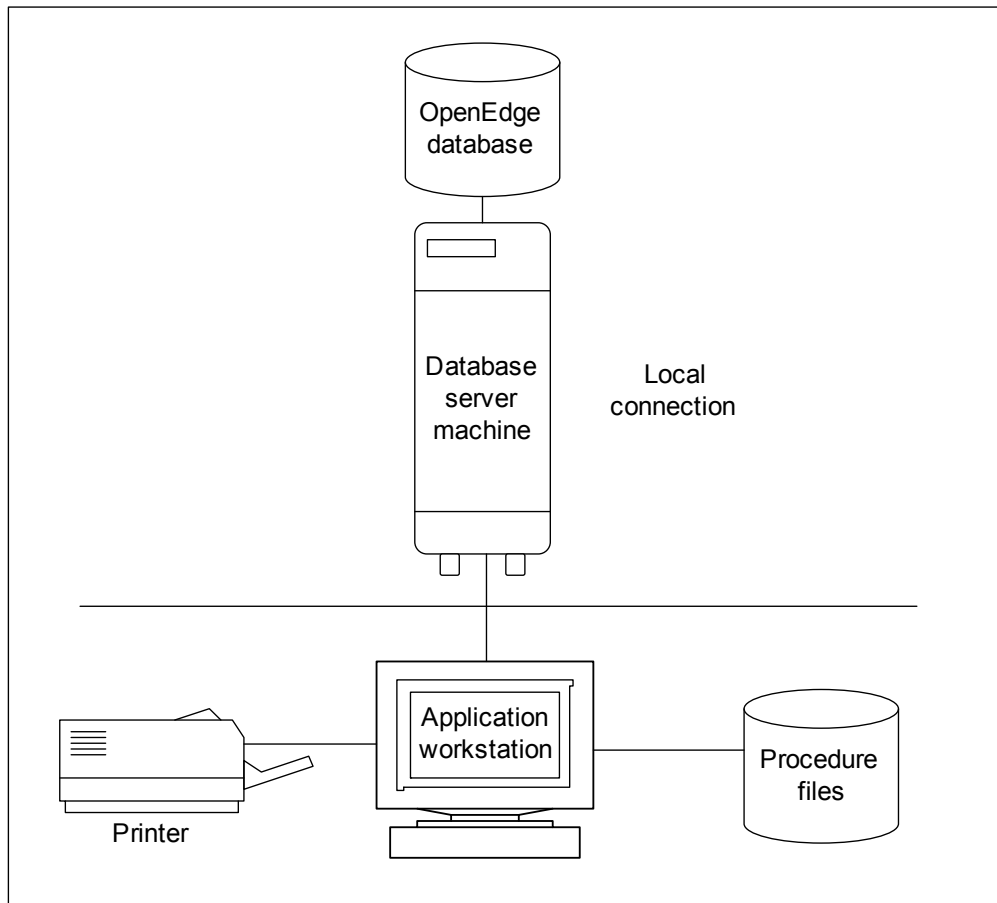


Figure 15: Simple OpenEdge network configuration

This configuration is typical of TCP/IP networks without file servers. There are no shared resources except the OpenEdge database. The application workstation and database server machine each have a hard disk. A printer is also attached to the application workstation. OpenEdge is installed on each node.

A workstation in this configuration often supports multiple users and clients (for example, a system with multiple terminals) who share the local printer and OpenEdge application. The database server machine is usually a high-performance back-end processor that can also support local self-service clients. This network configuration, with the OpenEdge database local to the database server machine, ensures full database integrity. With all files stored local to each node, it generally (but not always) provides the highest performance on a LAN.

Figure 16 shows a dedicated network file server, dedicated OpenEdge database server machine, and application workstations. Although the figure shows a limited number of workstations, file servers, and database server machines, there can be more of each.

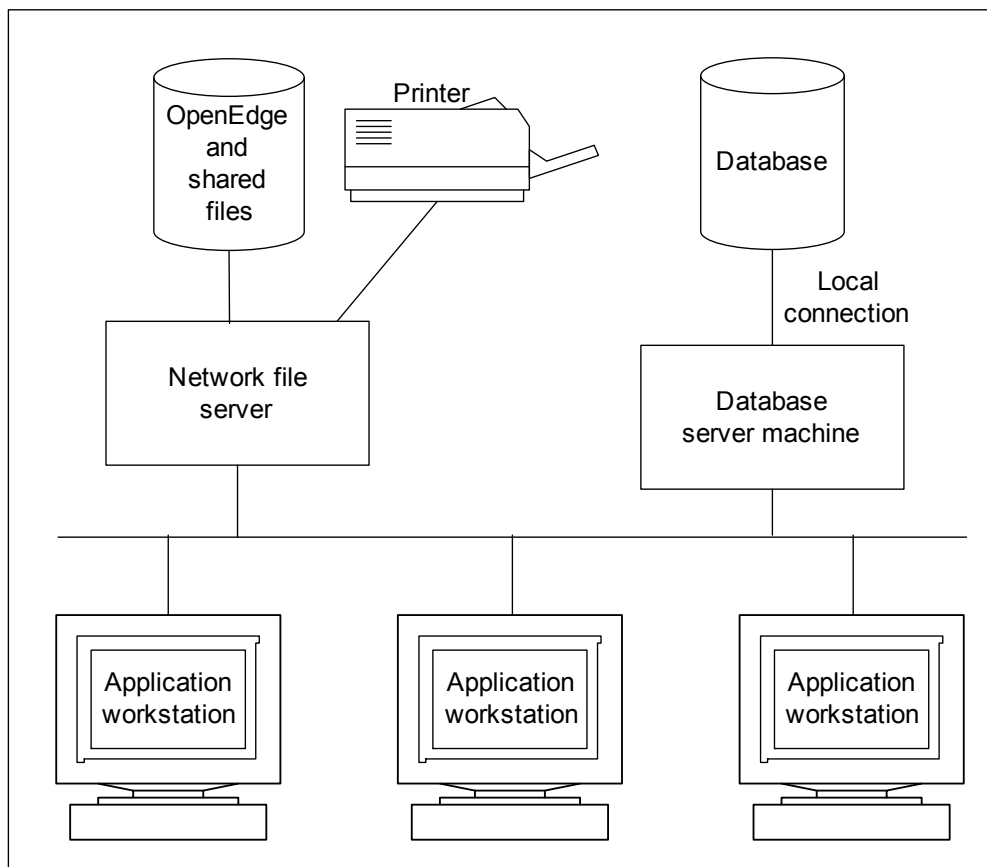


Figure 16: Network file server for application files

This is a configuration typical of PC LANs with file servers and network operating systems. A hard disk and a printer are attached to the network file server, and an additional hard disk is attached to the OpenEdge database server machine. The OpenEdge database is on the disk drive that is locally attached to the OpenEdge database server machine. OpenEdge and all application procedures are installed on the file server and shared by all other nodes.

This network configuration ensures full database integrity and high performance, limited only by network and application performance capabilities.

Figure 17 shows a network file server doubling as an OpenEdge database server machine and disk-optional application workstations. Although the figure shows a limited number of workstations, file servers, and database server machines, there can be more of each.

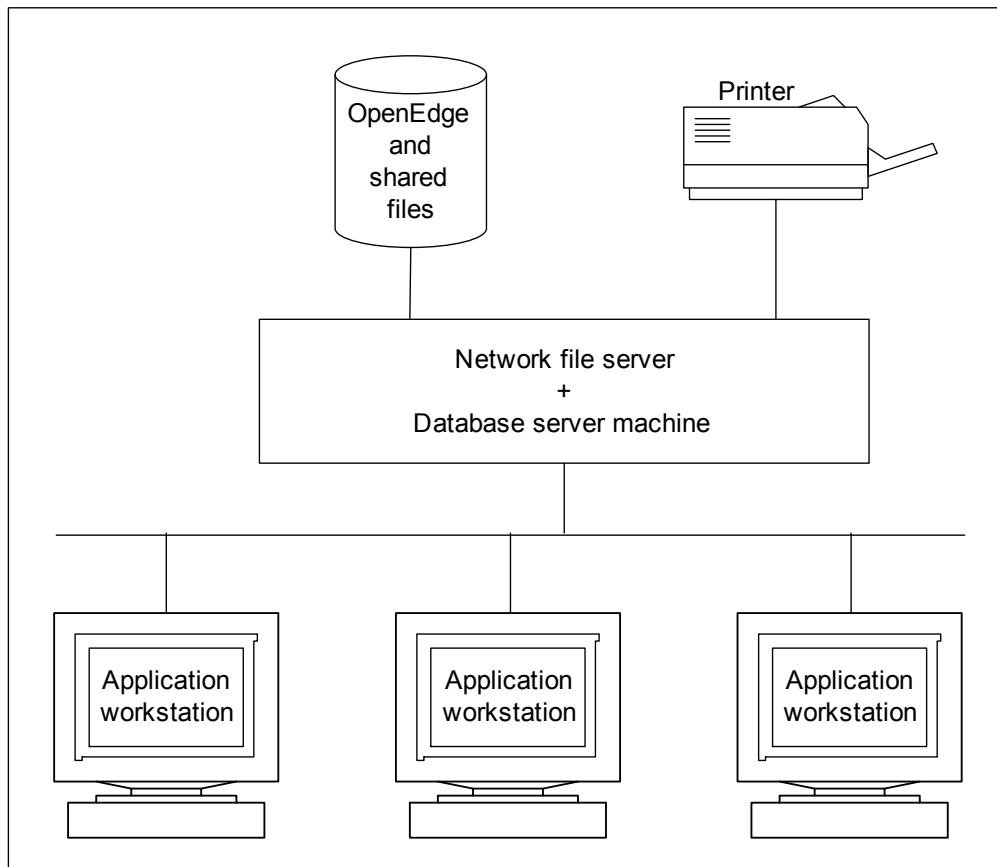


Figure 17: Network file server as a database server

This is a configuration you might find on a PC LAN with a powerful file server running a multi-tasking operating system. OpenEdge, application procedures, and the OpenEdge database are all installed on the file server and are shared by the other nodes.

This network configuration provides full database integrity and acceptable performance on a file server with high-speed CPU and I/O resources.

Note: Avoid doubling a network file server as a database server machine on low-capacity nodes or on nodes where the database server machine can run only in an emulated environment.

Figure 18 shows database files residing on a network file server.

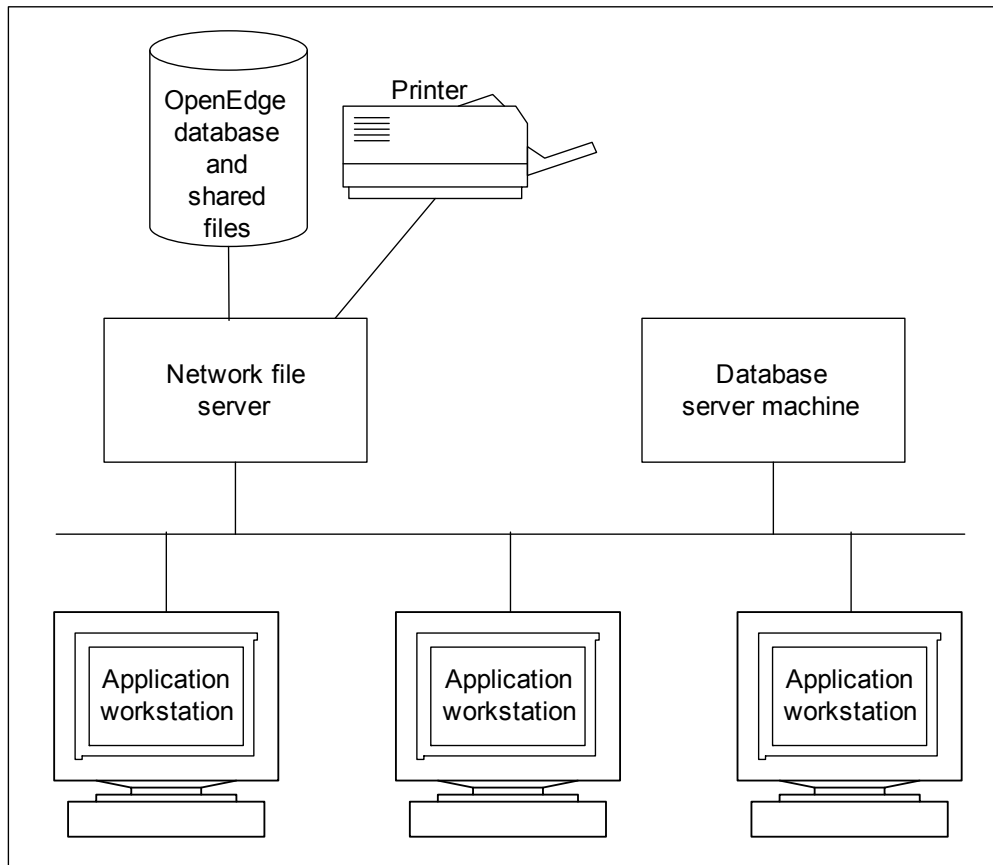


Figure 18: Network file server for application and database files

This network configuration runs the risk of compromising database integrity if the network file server or database server machine crashes, because the before-image (BI) file is on the network file server, making synchronous writes to it impossible. Performance also depends on whether network file server I/O efficiency compensates for traffic across the network.

An application server running on the application server machine connects through shared memory to an OpenEdge database and has access to a set of procedure files. An ABL application runs on the application workstation, connects to the application server running on the OpenEdge AppServer machine, and sends the requests to the application server to run remote procedures. The procedure execution and database access occur in a remote OpenEdge session context.

Figure 19 shows the simplest LAN configuration for OpenEdge on a network.

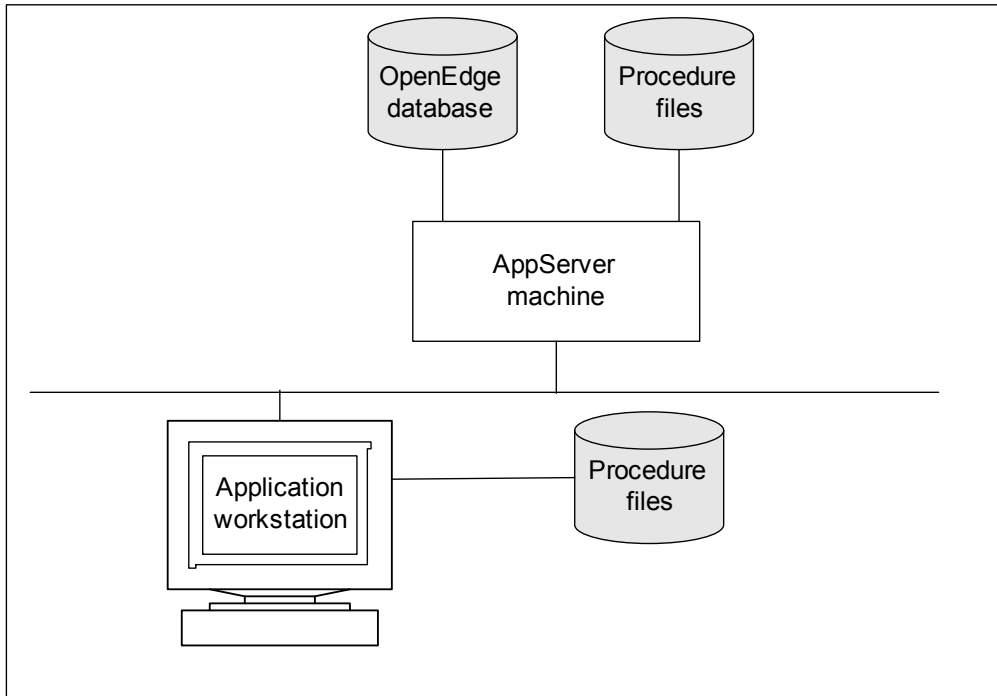


Figure 19: LAN configuration with the OpenEdge AppServer

In more complex implementations of the OpenEdge AppServer, an application server can connect to another application server in order to connect with a database. For more information about the OpenEdge AppServer, see [OpenEdge Getting Started: Application and Integration Services](#).

Preparing to run OpenEdge on a TCP/IP network

You can make OpenEdge operational in a network environment by following these guidelines:

- Identify and configure the nodes on your network for use as application workstations, database server machines, application server machines, and network file servers.
- Install OpenEdge on each node, or if your network has a network file server, install OpenEdge on the file server. For more information, see the [“Sharing an OpenEdge installation on a network overview”](#) section on page 145.
- If any application workstations and database server machines have incompatible processors or operating systems, you must install the appropriate OpenEdge product on each node.
- Set up network system files on each node.
- If you are using a network file server, make its resources, including printers and directories, available to all other nodes that require them.
- If you installed OpenEdge on a network file server, you might want to distribute the appropriate OpenEdge system files to the compatible application workstations and database server machines that use them. This takes advantage of networks where the local file and data access is faster than using the network.
- Set up your OpenEdge databases on each file server, database server, and application server machine.

Installing OpenEdge on your TCP/IP network

When installing OpenEdge on your network, note the following basic considerations:

- Where to place your database
- Where to place your OpenEdge executables and r-code files

Locating your database

Place your database on the hard disk of the machine that runs the OpenEdge server. If you place the database on a remote file server, synchronous writes are lost along with your database’s integrity, in the event of a system crash.

Synchronous writes ensure database integrity by flushing system buffers directly to disk. This is especially important for maintaining the before-image (BI) file. Therefore, if you must keep your database separate from the database server machine, use the before-image filename startup parameter to keep the before-image file local to the database server.

Note: Remote OpenEdge clients do not have to be concerned about synchronous writes because they do not write to the database.

Typical TCP/IP configuration with a hard disk on each machine

Figure 20 shows the configuration for a typical network when there is a hard disk on each machine and no file server is used.

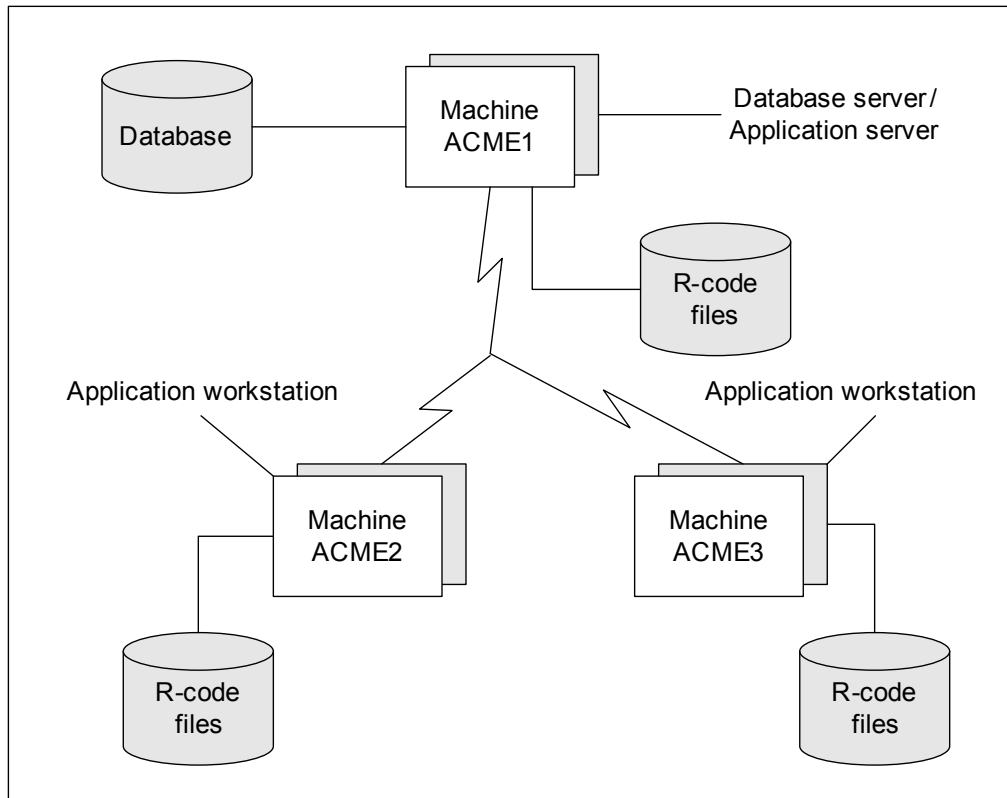


Figure 20: Typical TCP/IP configuration (file server not used)

When you use this configuration, you must install OpenEdge on each machine in the network. In Figure 20, the client machines do not have to be running the same operating system.

Setting up network files to run OpenEdge

There are several files you must check, and modify if necessary, before you can run OpenEdge on your network. The filenames and locations might differ for different operating systems and TCP/IP implementations, but the functional contents are identical. Table 136 lists these files.

Table 136: TCP/IP network files

File	Purpose
hosts	Lists machine names/network addresses
services	Lists OpenEdge server/port number
protocols	Defines system protocols

Configuring OpenEdge on a network operating system

This section describes preparations that you can make to promote efficient and reliable OpenEdge operation in a network operating system (NOS) environment, that is, a network environment that includes one or more network file servers that provide a common set of resource sharing and security services to other nodes. This section describes some of the more general considerations.

Making network resources available

Once you have installed OpenEdge, you must make sure that each application workstation and application server machine has access to OpenEdge system files, application files, and any other necessary network resources (such as printers). Each NOS provides a set of commands or utilities to make these resources available across the network. In general, you set up pointers to remote resources so that each workstation can access them as though they were local to the workstation. These pointers can be in the form of logical drives in Windows nodes, or mounted directory paths on UNIX nodes.

For more information on making network resources available, see the specific documentation for your network and operating system.

Setting network resource attributes

After you have made network resources available, you must make sure that they possess the necessary attributes to allow all application workstations to access them simultaneously. Each NOS provides a different means of setting the attributes to make network resources shareable.

For example, suppose you want to set the attributes of the OpenEdge installation directory on a network file server so that the OpenEdge files can be accessed by all workstations. OpenEdge is already loaded onto the network file server and is available to the network. The commands used to set resource attributes vary from network to network.

For further information on how to use these or equivalent commands for your network, see the documentation for your specific network and operating system.

Granting user access rights

After making OpenEdge network resources available and setting resource attributes, you might have to grant access rights to client users and Application Server machines in the network. Depending on your network, these access rights can include attributes such as read, execute, or open permissions that you must set for each user. See the network documentation for details about how to grant user access rights.

Note: User rights in an Application Server configuration are assigned to the machine where the application server resides, not to the user's client machine.

Remember that an OpenEdge database server can be a user on your network. Like application workstations, it might need user access rights granted to it. If you locate any database files on your network file server, be sure to grant the OpenEdge database server the necessary rights to access the network directory that contains the database.

AdminServer Authorization and Authentication

This appendix addresses additional AdminServer-related activities you can perform in Windows, as described in the following sections:

- [AdminServer logging details](#)
- [Determine the data logged in the AdminServer log](#)
- [Setting authentication option to start servers administered by the AdminServer](#)

Note: The procedures to establish AdminServer authorization options are located in the Windows online help system under these topic titles: “Establishing AdminServer Authorization Options during the Installation” and “Selecting the Authorization Feature when Starting the AdminServer.”

AdminServer logging details

There are logging entries that are specifically related to user authentication and authorization. This section identifies the log format and describes the information that it can contain.

Log format

The log lists both successful and failed operations in the following format:

```
[date] [level] ["security"] UserName:UserSuppliedPwd:GroupInfo:Text
```

Log contents

The following describes the fields in the security entry:

- **Date** — The existing Logging tool automatically inserts the current date using the existing AdminServer log format.
- **Level** — The possible levels are 1 through 5, in compliance with the existing AdminServer log conventions. The security entry will use only the following levels:
 - **0** indicates an internal error
 - **2** indicates an error condition and explains why the client was not authenticated or authorized
 - **3** indicates success and is used for tracking purposes
- **“security”** — This is a text constant that Progress specifies in order to simply log file scanning tools, so that an automated parser can easily identify security events.
- **UserName** — This field contains the user account being authenticated to the AdminServer. This field might indicate “no-user” if the authentication and/or authorization operation failed before the authentication portion could take place. In Windows systems only, the *UserName* might be in the form *[domain\]UserName* where domain is the result of an account lookup operation when the user has not specified a fully qualified user account.
- **UserSuppliedPwd** — This field indicates whether the password being validated for the user account is one of the three following possible conditions:
 - **Y** indicates that the password is supplied by the user
 - **N** indicates that the password supplied is by the single sign-on password generator
 - **X** indicates that the password has not yet been validated
- **GroupInfo** — This field contains group authorization information. When the AdminServer initializes, it validates that a minimum of one group is accessible before allowing startup. In this instance, the field will contain the list of available groups and unavailable groups. Unavailable groups are identified within enclosing braces.

The following example shows the format of **GroupInfo**:

```
group, group...;{unavailablegroup,unavailablegroup...}
```

In Windows only, the list of available groups might have the Windows domain prefixed in square brackets to indicate where the group name lookup operation found the entry.

When a security entry is made for an authentication or authorization operation, it can contain:

- **No Group Checking** — Indicates that the AdminServer started without the `-admingroup` option and no group authorization took place
- **GroupName** — Indicates that a single group name was successfully authorized for the user with a success message logged
- **GroupNames** — Indicates the group names that the user failed to authorize when the failure message was logged
- **Text** — This field contains one of the messages that further explains the success or failure. The possible text messages follow:
 - User is not authenticated
 - User is authenticated and authorized
 - User is not authorized
 - Failed to find the admingroup(s)
 - Failed to find the admingroup, not a valid group list
 - Failed to find the admingroup, please provide a valid group list
 - User password is not valid
 - System generated password has expired
 - Error, system generated password is not valid, user and host are valid
 - Valid group list

The default behavior for logging is that both success and failure events will be logged.

Determine the data logged in the AdminServer log

There is an AdminServer command-line option for JVMARGS that is called `DLogLevelSecurity`, that, when set, determines the type of logging that the AdminServer log file captures. The syntax for JVMARGS is as follows:

Syntax

```
JVMARGS="$JVMARGS -DLogLevelSecurity={2|3}"
```

`DLogLevelSecurity=2`

Stops successful logins from being logged.

`DLogLevelSecurity=3`

Logs failures and successes.

Setting authentication option to start servers administered by the AdminServer

You can require that when users are starting servers of the AdminServer (AppServer, Adapter for SonicMQ, and WebSpeed) the `ubroker.properties` file must provide a valid username and password. This authentication for starting the AppServer, Adapter for SonicMQ, and WebSpeed uses the `ubroker.properties` file hierarchy to find usernames and passwords. The OpenEdge Management or OpenEdge Explorer password field can be set to supply the username's password.

The command-line option that tells the AppServer, Adapter for SonicMQ, and WebSpeed to require a username and password from the `ubroker.properties` file is **Require Username** (`-requireusername`). You can run `OpenEdge-install-dir\bin\genpassword`. This gives the user an obfuscated password that the user can enter into OpenEdge Management or OpenEdge Explorer.

The Require Username syntax is as follows:

Syntax

<code>-requireusername</code>

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

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

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/>.

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/>)."

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 -

Copyright 1999-2004 (C) Intalio Inc., and others. 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 "ExoLab" must not be used to endorse or promote products derived from this Software without prior written permission of Intalio Inc. For written permission, please contact info@exolab.org. 4. Products derived from this Software may not be called "Castor" nor may "Castor" appear in their names without prior written permission of Intalio Inc. Exolab, Castor and Intalio are trademarks of Intalio Inc. 5. Due credit should be given to the ExoLab Project (<http://www.exolab.org/>). THIS SOFTWARE IS PROVIDED BY INTALIO 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 INTALIO 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 Decimal Conversion Code (dtoa.c; g_fmt.c; rnd_prod.s; decstrod.c; decstrtof.c; dmisc.c; gdtoa.c; gdtoa.h; gdtoaimp.h; gethex.c; gmisc.c; hd_init.c; misc.c; smisc.c; strtodg.c; strtord.c; sum.c; ulp.c). Such technologies are subject to the following terms and conditions: dtoa.c License: The author of this software is David M. Gay. Copyright (c) 1991, 2000, 2001 by Lucent Technologies. Permission to use, copy, modify, and distribute this software for any purpose without fee is hereby granted, provided that this entire notice is included in all copies of any software which is or includes a copy or modification of this software and in all copies of the supporting documentation for such software. THIS SOFTWARE IS BEING PROVIDED "AS IS", WITHOUT ANY EXPRESS OR IMPLIED WARRANTY. IN PARTICULAR, NEITHER THE AUTHOR NOR LUCENT MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND CONCERNING THE MERCHANTABILITY OF THIS SOFTWARE OR ITS FITNESS FOR ANY PARTICULAR PURPOSE. g_fmt.c License: The author of this software is David M. Gay. Copyright (c) 1991, 1996 by Lucent Technologies. Permission to use, copy, modify, and distribute this software for any purpose without fee is hereby granted, provided that this entire notice is included in all copies of any software which is or includes a copy or modification of this software and in all copies of the supporting documentation for such software. THIS SOFTWARE IS BEING PROVIDED "AS IS", WITHOUT ANY EXPRESS OR IMPLIED WARRANTY. IN PARTICULAR, NEITHER THE AUTHOR NOR LUCENT MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND CONCERNING THE MERCHANTABILITY OF THIS SOFTWARE OR ITS FITNESS FOR ANY PARTICULAR PURPOSE.

rnd_prod.s License: The author of this software is David M. Gay. Copyright (c) 1991 by Lucent Technologies. Permission to use, copy, modify, and distribute this software for any purpose without fee is hereby granted, provided that this entire notice is included in all copies of any software which is or includes a copy or modification of this software and in all copies of the supporting documentation for such software. THIS SOFTWARE IS BEING PROVIDED "AS IS", WITHOUT ANY EXPRESS OR IMPLIED WARRANTY. IN PARTICULAR, NEITHER THE AUTHOR NOR LUCENT MAKES ANY

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.

gdtoc.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.

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 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.

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 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.

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. strtford.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.

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 reStructuredText 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 libtiff 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 "cjpeg" and "djpeg", 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:

```
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,

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.

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. Dobbs'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 `gzseek` (Frank Faubert)
- fix `miniunzip` when the last entry in a zip file is itself a zip file (J Lillge)
- 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@bfsec.bt.co.uk> is in the CPAN (Comprehensive Perl Archive Network) sites, such as:

ftp://ftp.cis.ufl.edu/pub/perl/CPAN/modules/by-module/Compress/Compress-Zlib*

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 (formerly 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:

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

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. Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

Progress OpenEdge v11.3 may incorporate International Components for Unicode v4.8.0. Such technology is subject to the following terms and conditions: ICU License - ICU 1.8.1 and later COPYRIGHT AND PERMISSION NOTICE Copyright (c) 1995-2011 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 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.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

Progress OpenEdge v11.3 may incorporate java.net args4j v2.0.12 from Kohsuke Kawaguchi. Such technology is subject to the following terms and conditions: Copyright (c) 2003, Kohsuke Kawaguchi 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. 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 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 Lentzsch. All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: o 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 (ey@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com). Original SSLeay License. Copyright (C) 1995-1998 Eric Young (ey@cryptsoft.com) All rights reserved. This package is an SSL implementation written by Eric Young (ey@cryptsoft.com). The implementation was written so as to conform with Netscape's SSL. This library is free for commercial and non-commercial use as long as the following conditions are adhered to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, 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.

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 Univeristy" and "Indiana Univeristy 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 Univeristy" name nor may "Indiana Univeristy" 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 Univeristy" and "Indiana Univeristy 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 Univeristy" name nor may "Indiana Univeristy" 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. 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

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" 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

Numbers

4GL Development System
 components during a Complete
 installation 337
 components installed during
 Complete installation 397

A

Access rights
 networks 520

Accessing a server behind a firewall 329

Additional resources for most current
 system requirements
 Product Availability Guide 32
 Release Notes 32

Addressing
 OpenEdge brokers/servers 326
 OpenEdge hosts 326

AdminServer
 additional considerations 286
 authorization options 227
 changing the default port 284
 changing the default startup setting
 285
 considerations 226
 defined 271, 274
 hosts 452
 logging details related to
 authentication and authorization
 522 to 524
 ports 453
 querying 286

 setting authentication option to start
 servers 525
 starting 283
 stopping 283
 understanding and using 282 to 287

AdminServer plugins.properties file
 defined 273

After-image writers (AIW) 508

alias.pem files 259

alias.pk1 files 259

alias.pk10 files 259

Application Server
 Components of Complete installation
 345, 347

Application services 489

Application workstations 509

Applications
 starting on networks 327

AppServer Internet Adapter
 components installed during
 Complete installation 400
 components of Complete installation
 340

AppServer Maximum Maintained
 Prestart Counter (-Mms) parameter
 326

AppServer Maximum Prestart Counter
 (-Ms) parameter 326

Arbitrary sum weight factors 492
Asynchronous page writers (APWs) 508

B

Background mode
 multi-user OpenEdge 331, 332
 single-user OpenEdge 318

Batch mode
 defined (UNIX) 165
 defined (Windows) 123
 multi-user OpenEdge 331, 332
 oesetup.ini (Windows) 123, 165
 single-user OpenEdge 323

Batch startup commands
 single-user 316, 322

Before-image Filename (-g) startup
 parameter 518

Before-image writers 508

BPRO utility 318, 323

Broadcast request to NameServer 494

Brokers
 addressing 326
 defined 508
 multi-broker access 328
 starting 327
 servers for remote users 319, 324

C

Certificate store
 OpenEdge
 installing trust CA/root certificates 261
 management 261
 managing 231, 254, 255

certutil
 purpose 262
 syntax and operations 263

Changing the AdminServer default port 284

Client Networking
 components of Complete installation 341

Client Networking, components
 installed during Complete installation 400

Client startup parameters
 networks 325

Client/Server configurations 509
 samples 510
 terminology 509

Client/Server OpenEdge
 addressing 326
 broker/server addressing 326
 execution 325
 starting clients and servers 325

Clients
 defined 509
 OpenEdge
 installing trusted CA/root certificates 261
 SSL certificate store management 261
 starting
 TCP/IP on UNIX 327

Command-line utilities
 defined 271

Complete installation 71

Completing OpenEdge installation in Windows 99

Components
 used to calculate memory needs 196

Components of Complete installation
 OpenEdge Adapter for Sonic ESB 344
 Personal RDBMS 367

Components of Complete installation in Windows 335 to 390
 4GL Development 337
 Application Server - Basic 345
 Application Server Enterprise 347
 AppServer Internet Adapter 340
 Client Networking 341
 Enterprise RDBMS 364
 NameServer Load Balancer 344
 ODBC DataServer 356
 OpenEdge Adapter for Sonic ESB 344
 OpenEdge Application Server Enterprise 347
 OpenEdge Development Server 411
 OpenEdge Replication 369
 OpenEdge Replication Plus 370
 OpenEdge Studio 372
 Oracle DataServer 358
 Personal RDBMS 367
 Progress Developer Studio for OpenEdge 350
 Query/Results 380
 Translation Manager 382, 383
 Ultra Controls 377
 Visual Translator 383
 Web Services Adapter 385

- WebSpeed Messenger 386, 393
- WebSpeed Workshop 387
- Workgroup RDBMS 377
- Components on UNIX
 - installed during a Complete installation
 - 4GL Development System 397
 - AppServer 405
 - AppServer Internet Adapter 400
 - Client Networking 400
 - NameServer 403
 - NameServer Load Balancer 404
 - OpenEdge Replication 402
 - OpenEdge Replication Plus 403
 - Oracle DataServer 409
 - Personal Database 417
 - Secure AppServer 409
 - WebSpeed Messenger 424
 - Workgroup Database 419
- Configuration XML file
 - editing manually 108
- Configurations
 - client/server 509
 - displaying information 192
 - networks 512
 - shared memory 506
- Configuring
 - NameServers
 - using OpenEdge Management or OpenEdge Explorer 502
 - OpenEdge on an NOS 520
- conmgr.properties file
 - defined 273
- Connection parameters
 - icfdb 101
- Connection-level fault tolerance 488
 - two techniques 494
 - understanding 493
- Control number
 - displaying 187
- Custom installation 71
- D**
 - d9855a82.0 file 261
 - Data Administration Tool
 - creating a database 228
 - Data Dictionary
 - creating a database 228
 - Database server machines
 - defined 509
 - multi-process
 - defined 510
 - single-process
 - defined 510
 - Databases
 - monitoring 508
 - startup 327
 - DataServer for ODBC
 - components of Complete installation 356
 - DataServer for Oracle
 - components of Complete installation 358
 - Default services 489
 - defined 489
 - Default SSL identity
 - Progress CA root digital certificate 261
 - DefaultFont environment variable
 - international progress.ini files 485
 - Delta files
 - defined 291
 - DLC
 - and OpenEdge-install-dir 79, 85
 - environment variable, UNIX 242
 - setting the environment variable (Windows) 219
 - Downloading executables for heterogeneous environments 121, 164
 - Dynamics Configuration Utility (DCU)
 - running 101
- E**
 - Enterprise Database, components
 - installed during Complete installation 414
 - Enterprise RDBMS
 - components of Complete installation 364
 - Environment variables
 - Java (Windows) 216
 - PATH, UNIX 242
 - setting (Windows) 216
 - UNIX 241, 242
 - Windows 218
 - Error code parameters 202
 - Error messages

- related to installation 203
- European Numeric Format (-E) 483
- Expiration date
 - displaying 187
- F**
- Fail-over weight factor 492
- File descriptors
 - UNIX requirements for WebSpeed 59
- File types
 - supported by OpenEdge 82
- Files
 - conmgr.properties 452
 - icf.ini 108
 - icf.pf 108
 - icfconfig.xml 108
 - network sharing 520
 - protocols, *See* protocols file
 - services, *See* services file
 - startdb.bat 108
 - stopdb.bat 108
- Formulas, for calculating memory requirements 199

G

- genpassword tool
 - example 260
 - purpose 260

H

- H startup parameter
 - client connection 325
 - server startup 325
 - specifying 326
- Host addressing 326
- Host Name (-H) parameter
 - client connection 325
 - server startup 325
 - specifying 326
- Host request to NameServer 494
- HP Tru64
 - preinstallation tasks for 84
- HP-UX
 - modifying JDKHOME value for 246

I

- icf.ini file
 - editing manually 108
- icf.pf file
 - editing manually 108
- ICFDB database 101
 - creating 101
- Icons
 - used by OpenEdge 82
- Installation
 - directory structure in Windows 79
 - directory structure on UNIX 85
 - displaying configuration information 192
 - displaying date 187
- Installation activities
 - additional tasks in Windows 116
- Installation considerations
 - UNIX-specific 84
 - Windows-specific 75
- Installation media
 - loading 98
 - types of 98
- Installation method
 - defined 70
 - online, interactive 70
 - silent (or batch mode) 70
- Installation type
 - Complete option 71
 - Custom option 71
 - defined 71
- Installing and managing keys and digital certificates
 - certutil 464
 - genpassword 466
 - mkhashfile 467
 - pkiutil 468
- Installing OpenEdge 97 to 121
 - and performing postinstallation tasks (on UNIX) 175
 - choosing an installation type when 396
 - finishing 99
 - in a heterogeneous environment 121
 - in a heterogeneous environment (on UNIX) 164
 - troubleshooting when 203 to 205
 - viewing registry information 120
 - for ODBC driver 120
- Installing OpenEdge in Windows
 - post-installation tasks 137

Installing OpenEdge products to a
current installation 117

Interactive mode
multi-user OpenEdge 330

Interactive mode, multi-user OpenEdge
330

International
databases 477
parameter files 483

J

Java
system requirements (in Windows) 32

Java Development Kit (JDK)
environment variables (Windows)
216

Java requirements on UNIX 54
JDK 54
JRE 57

Java requirements specific to each
UNIX platform 58

Java Runtime Environment (JRE)
environment variables (Windows)
216

JavaTools.properties file
defined 273

JDKHOME
setting (Windows) 217

JDKHOME, setting (UNIX) 246

K

Kernel
configuration 200
reconfiguration parameters 202

Key store
OpenEdge
backup subdirectory 259
content and structure 258
creating and managing entries 256
management 256
managing 231, 254, 255
policy subdirectory 259
requests subdirectory 259
specifying entry password 257

L

LAN OpenEdge

startup parameters 325

Language
support 471

License
information 51, 65

License Addendum 69

License Update utility
changes to accommodate license
updates 184
defined 184
updating a PSDN subscription
renewal 184
updating an evaluation license to a
non-evaluation license 184

Licenses
displaying current information 186
managing 190
user counts 192

Load balancing
arbitrary sums 492
fail-over 492
percentage-based 491
priority weight factors 491

Loading
remote databases 101

Local NameServer instances 502

Location transparency 488

Log (.lg) files
commands to removing entries 207

M

-m2 startup parameter 325

-m3 startup parameter 325

-Ma startup parameter 325

Machine addressing 326

Machine Class
displaying 187

Maintaining two versions of OpenEdge
and Progress (Windows) 230

Manual Server (-m2) startup parameter
325

Maximum Clients per Server (-Ma)
parameter
server startup 325

Maximum Dynamic Server (-Maxport)

- parameter
 - server startup 326
- Maximum Servers (-Mn) parameter
 - server startup 326
- Maximum Servers per Broker (-Mpb)
 - startup parameter
 - server startup 326
- Maximum Servers per Protocol (-Mp)
 - parameter
 - server startup 326
- Maxport startup parameter 326
- MBPRO command 331, 332
- Memory
 - and semaphore parameter settings 201, 202
 - and setting swap space 200
 - calculating requirements for 196, 198
 - formulas to calculate requirements for 199
 - multi-user requirements for 198
 - reducing usage 200
 - shared 200
 - parameter settings 201, 202
 - single-user requirements for 198
- Mergeprop utility
 - actions 295
 - defined 271, 291
 - delta file 294
 - overview 291
 - property files managed by 294
 - syntax 292
 - target file 294
- mergeprop utility
 - Java API 299
- Maximum Maintained Prestart Counter for AppServer (-Mms) parameter 326
- Message Buffer Size (-Mm) parameter
 - client connection 325
 - server startup 326
- Mi startup parameter
 - server startup 326
- Minimum Clients per Server (-Ma)
 - parameter
 - server startup 326
- Minimum Dynamic Server (-Minport)
 - parameter
 - server startup 326
- Minimum operating system level, by platform 37, 60
- Minport startup parameter
 - server startup 326
- mkhashfile tool
 - purpose 262
 - syntax and operations 264
- Mm startup parameter
 - client connection 325
 - server startup 326
- Mms startup parameter 326
- Mn startup parameter
 - server startup 326
- Modifying
 - an OpenEdge installation 138
- Mp startup parameter
 - server startup 326
- Mpb startup parameter
 - server startup 326
- MPRO command 327
- MPRO utility 330
- Ms startup parameter
 - server startup 326
- Multi-threaded servers
 - 509
- Multi-user memory requirements 198
- Multi-user OpenEdge 319, 330, 332
 - commands to start
 - batch mode 331
 - interactive mode 330
 - startup commands 319
 - batch mode 331
 - broker 319
 - interactive mode 330
 - server 319
 - Windows service 319
 - commands to start
 - batch mode 332
 - interactive mode 330
 - starting the server 324
 - startup commands
 - batch mode 332
 - broker 324
 - interactive mode 330
 - server 324
- Multi-user OpenEdge startup
 - commands
 - server 319
- Multi-user OpenEdge
 - starting the server 319

N

-N startup parameter

- client connection 325
- defaults 326
- server startup 326
- specifying 326

NameServer

- accessing its port and host 490
- broadcast address 497
- client connections 502
- components installed during Complete installation 403
- configuring
 - using OpenEdge Management or OpenEdge Explorer 502
- configuring communications 489
- downloading executables 501
- editing the services file 490
- instance configuration 502
- neighbors 494
 - example 499
 - using 498
- replication 494
 - example 496
 - using 495
- starting instances
 - using OpenEdge Management or OpenEdge Explorer 503
- UDP broadcasting 494

NameServer Load Balancer Option

- components of Complete installation 344

NameServer Load Balancer,

- components installed during Complete installation 404

NameServers

- defined 488

Neighbor NameServers 494

NetSetup

- running the Shared Network Installation Utility 147
- shortcuts 147

Network file servers

- defined 509

Network operating system

- access rights 520
- accessing resources 520
- OpenEdge startup 325

Network startup parameters 325

Network Type (-N) parameter

- client connection 325
- defaults 326

protocol support 512

- server startup 326
- specifying 326

Network Version (-Nv) startup

- parameter
 - client connection 325

Networks

- access rights 520
- accessing resources 520
- applications
 - starting 327
- client startup parameters 325
- client/server
 - startup 327
- configurations 512
- database server machines 509
- file server 146, 509
 - applications 514, 516
 - as database server machine 515
- file servers 509
- OpenEdge startup parameters 325
- PC LAN permissions 520
- resource sharing 520
- server startup parameters 325
- types 512
- types defined 512

-Nv startup parameter 325

O

ODBC DataServer

- components of Complete installation 356, 358, 360

Online, interactive installation method

- 70

OpenEdge

- broker/server addressing 326
- client/server startup 325
- configuration file (Progress.cfg) 203
- configuring on an NOS 520
- icons 82
- installing 97 to 121
- installing (UNIX) 155
- integration with Windows Explorer 82 to 83
- messages, *See* PROMSGS
- network addressing 326
- network startup parameters 325
- starting multi-user 332
 - batch mode 331, 332
 - interactive mode 330
- starting single-user
 - batch mode 318, 323
 - interactive mode 318, 323
- supported file types 82
- supported platforms (UNIX) 37, 60

- OpenEdge Adapter for Sonic ESB
 - components installed during Complete installation 404
 - components of Complete installation 344
- OpenEdge and Progress
 - maintaining two versions (Windows) 230
- OpenEdge Application Server
 - components of Complete installation 347
- OpenEdge Application Server Basic
 - components of Complete installation 345
- OpenEdge AppServer, components installed during Complete installation 405
- OpenEdge Business Process Modeler 391
- OpenEdge Business Process Server 391, 444
- OpenEdge client/server
 - network file server
 - application files 514, 516
 - network file server and as database server machine 515
 - running on a LAN 325
 - startup parameters 325
 - with AppServer 517
- OpenEdge configuration file (Progress.cfg), altered or missing 203, 204
- OpenEdge Database Monitor utility (PROMON) 508
- OpenEdge Developer Studio
 - components of Complete installation 350
- OpenEdge Development Server
 - components of Complete installation 360, 411
- OpenEdge directory structure
 - in Windows 79
 - on UNIX 85
- OpenEdge installation
 - TCP/IP networks 518
- OpenEdge installation media
 - requirements 35
- OpenEdge key and certificate stores
 - introduction 231
- OpenEdge Management and OpenEdge Explorer
 - Browser support 91
 - product support 91
- OpenEdge Management or OpenEdge Explorer
 - using 288
- OpenEdge Management or OpenEdge Explorer configuration tool
 - launching 289
 - using 288
- OpenEdge Management or OpenEdge Explorer, *See* OpenEdge Management or OpenEdge Explorer configuration tool
- OpenEdge product
 - details about components and subcomponents 395
- OpenEdge Replication 402
 - components installed during Complete installation 369
- OpenEdge Replication Plus
 - components installed during Complete installation 370, 403
- OpenEdge sample configurations
 - network file servers
 - application files 514, 516
 - as database server machine 515
 - with AppServer 517
- OpenEdge Silent Installation Utility
 - syntax (UNIX) 172
- OpenEdge Silent Installation utility
 - syntax (Windows) 133
- OpenEdge SSL
 - certutil tool 262
 - clients
 - certificate store management 261
 - installing root CA certificates 261
 - key and certificate stores 231, 254, 255
 - key store content 258
 - key store entry password 257
 - mkhashfile tool 262
 - pkiutil tool 256
 - servers
 - establishing identity 256
 - key store management 256
- OpenEdge Watchdog utility (PROWDog) 508
- OpenEdge-install-dir and DLC 79, 85

- Oracle DataServer
 - components of Complete installation 358
- Oracle DataServer, components installed during Complete installation 409
- P**
- Parameter file
 - typical startup parameters 483, 484
- Parameters
 - icfdb startup 101
- Parsing XML files 101
- Password for OpenEdge key store entries 257
- PATH, environment variable, UNIX 242
- Percentage-based weight factors 491
- Permissions
 - networks 520
- Personal Database, components installed during Complete installation 417
- Personal RDBMS
 - components of Complete installation in Windows 367
- pkiutil tool
 - purpose 256
 - syntax and operations 257
 - using 256
- Planning your installation 69
- Platforms, supported (UNIX) 37, 60
- Port Number
 - displaying 187
- Post-installation tasks (UNIX) 175
- Post-installation tasks (Windows) 137
- Preinstallation checklist
 - for UNIX 439
 - for Windows 427
- Preinstallation documentation
 - resources 69
 - checklists 69
 - License Addendum 69
 - OpenEdge Installation online help (Windows) 69
 - product component and subcomponent details 69
 - Release Notes 69
- Preinstallation tasks 68
 - for HP Tru64, AIX, and Linux 84
 - for SQL install 76
 - on a shared network 146, 147, 151
- Preinstallation tasks overview 68
- PrinterFont environment variable
 - international progress.ini files 485
- Priority weight factor
 - load balancing 491
- PRO utility 318, 323
- PROAB section in progress.ini 99
- Procedure Editor startup commands
 - UNIX, Windows 316
- Procedure Editor startup commands, UNIX, Windows 322
- PROCFG, environment variable, UNIX 242
- PROCONV, environment variable, UNIX 243
- PRODB
 - creating a database 228
- PRODB utility 101
- Product Availability Guide 54
 - accessing 32
- Product configurations
 - ubroker.properties file 304
- Product name
 - displaying 187
- Product support for OpenEdge Management and OpenEdge Explorer 91
- Product update utility 185
- Product Update utility, *See also* License Update utility
- PROEXE, environment variable, UNIX 243
- Program item properties
 - setting 223
- Progress Developer Studio
 - components of Complete installation 350

Progress Developer Studio for
OpenEdge 94

Progress Download Center
accessing 69

Progress Dynamics
Completing the DCU wizard 102

Progress Version 9 84

progress.cfg file 204

progress.ini file
editing 485
Windows 218

PROLOAD, environment variable, UNIX
243

PROLOG
removing log entries 207

PROMON utility (OpenEdge Database
Monitor) 508

PROMSGS 478
defined 472
environment variable 485
environment variable, UNIX 243
messages 478
removing prior to a reinstall 78
shell integration 82
translations 472

PROPATH, environment variable, UNIX
244

Properties file
general syntax 301
information added by OpenEdge 83
ubroker.properties 304

PROSERVE
command 327
utility 324, 328

PROSRV, environment variable, UNIX
244

PROSTART, environment variable,
UNIX 244

PROTERMCAP, environment variable,
UNIX 245

Protocols file 489

PROWDOG utility (OpenEdge
Watchdog) 508

pscca.cer file 261

pscpki.cnf file 259

Q

Query/Results
components of Complete installation
380

R

Regional support
installing PROMSGS 476

Registry
editing 485
Windows 218

Release Notes 54
accessing 32

Remote database
loading 101

Remote NameServer instances 502

Removing log file entries
PROLOG 207

Removing OpenEdge
manually (Windows) 140

Replicated NameServers 494

Repository
creating 101

Required third-party applications for
Windows
DataDirect ODBC branded drivers 47
DataDirect SQL drivers 47
Microsoft Management Console 46

Required third-party applications in
Windows 46

Resources
network sharing 520

Root CA digital certificate
Progress CA 261

Running OpenEdge
in Windows 229

Running the Shared Network
Installation Utility (NetSetup) 147

S

-S startup parameter 326
client connection 325
server startup 326

Saving an existing OpenEdge or

- Progress installation 76, 78, 84
- Secondary Login Broker (-m3) startup parameter
 - server startup 325
- Secure AppServer
 - components installed during Complete installation 407
- Self-service clients
 - defined 506
- Semaphore parameter settings 201, 202
- Serial number
 - displaying 187
- Server access
 - behind a firewall 329
- Server compatibility
 - ABL clients and database servers (Windows) 43
- Server compatibility in Windows 43
- Server startup parameters
 - networks 325
- Server-level fault tolerance 488
 - understanding 493
- Servers
 - addressing 326
 - defined 509
 - network 509
 - OpenEdge
 - establishing SSL identity 256
 - SSL key store management 256
 - starting 327
- Service Name (-S) startup parameter 326
 - client connection 325
 - server startup 326
- Services file 490
- Setting
 - environment variables (Windows) 216
 - Java environment variables (Windows) 216
 - JDKHOME (Windows) 217
- Setting environment variables for SQL (UNIX) 247
- Setting up the OpenEdge environment
 - [Startup] section 218, 219, 220, 221
 - [WinChar Startup] section 218, 219, 220, 221
 - DLC variable (Windows) 219
 - environment variables (Windows) 216
 - Java environment variables (Windows) 216
 - program item properties (Windows) 223
- Setting up the OpenEdge environment (Windows) 215
- Shared memory 200, 506
 - architecture 507
 - configurations 506
 - parameter settings 201, 202
- Shared network
 - preinstallation tasks when installing on a 146, 147, 151
- Shared Network Installation Utility
 - defined (Windows) 145
- Shared Network Installation utility 74, 145 to 151
- Sharing an OpenEdge installation on a network 145 to 151
 - primary tasks 145
 - running the Silent installation option 152
 - setting up the shared network 147
- Shortcut menu
 - using with OpenEdge 83
- Show Configuration (SHOWCFG) utility 184, 189, 192
- Silent install, *See also* Batch mode
- Silent Installation
 - defined 70
- Silent installation in Windows
 - data input options 124
 - log file details 134
 - optional activities
 - creating data input 136
 - manually modifying data input 136
 - overview 123
 - response.ini 124
 - running
 - commands 133
- Silent installation on UNIX
 - data input options 166
 - log file details 173
 - optional activities
 - creating data input 174
 - manually modifying data input 174
 - overview 165
 - response.ini 166
 - running

- commands 172
- Single-threaded servers 509
- Single-user memory requirements 198
- Single-user OpenEdge
 - startup commands
 - background mode 318
 - batch mode 318
 - interactive mode 318
- Single-user OpenEdge startup commands
 - batch mode 323
- Single-user OpenEdge, startup commands
 - interactive mode 323
- Size increments for increasing startup parameters 198
- SQL Server
 - components of Complete installation 354
- startdbs.bat file
 - editing manually 108
- Starting
 - clients and servers 325
 - network applications 327
- Starting OpenEdge
 - multi-user 332
 - batch mode 330
 - batch mode in Windows 331
 - interactive mode in windows 330
 - interactive mode on UNIX 330
 - Windows service 319
 - with a server 319, 324
 - multi-user on UNIX
 - single-user
 - batch mode 318, 323
 - interactive mode 318, 323
- Startup commands
 - summary 314
- Startup commands, summary 321
- Startup parameters
 - client 325
 - deployed applications 483
 - icfdb 101
 - server 325
 - size increments for increasing 198
- Startup scripts, tailoring 204, 205
- stopdbs.bat file 108
- Supported, platforms on UNIX 37, 60

- Swap space 200
- System requirements
 - for running OpenEdge applications (UNIX) 59
 - Java
 - HP-UX Itanium 58
 - Solaris SPARC 58
 - Java (on UNIX) 54
 - Java (Windows) 32
 - Windows 2003 32
- System requirements for Windows
 - required third-party applications 46
- System requirements on UNIX and Linux 54

T

- Taget files
 - defined 294
- Tailoring startup scripts 204
- Target files
 - defined 291
- TCP/IP clients
 - MPRO command 327, 328
 - starting in Windows 327
 - starting on UNIX 328
- TCP/IP networks
 - installing OpenEdge 518
 - network files 519
 - preparing for OpenEdge 519
 - without file server 519
- TERM, environment variable, UNIX 243, 244, 245
- Terminal identifiers 251, 252
- Translation Manager
 - components of Complete installation 382, 383
- Troubleshooting an installation 203 to 205

U

- ubroker.properties file 304
 - defined 272
 - products supported 307
 - structure 306
- UDP broadcasting 494
 - performance implications 500
- ULIMIT

- WebSpeed Agents requirement (on UNIX) 59
 - Ultra Controls
 - Components of Complete installation 377
 - Unified Broker 268, 269, 271
 - additional characteristics 278
 - and Name Server relationship 489
 - application services 489
 - clients 305
 - common elements of 276
 - default sample broker for each Unified Broker product 277
 - installation prerequisites 305
 - NameServer instances 502
 - order of configuration 501
 - properties file
 - guidelines for editing 308
 - using default sample brokers 277
 - Unified Broker Framework
 - Unified Broker
 - common elements 276
 - default sample brokers 277
 - Unified Broker framework
 - additional considerations 274
 - element and description overview 269
 - Unified Broker
 - configuring and starting 278
 - details 276
 - Unified Broker products
 - administering and configuring 448
 - ASBMAN 448
 - DBMAN 452
 - mergeprop 454
 - NSMAN 456
 - PROADSV 458
 - that support broker functionality 268
 - ubroker.properties file 268
 - WTBMAN 461
 - Uninstalling OpenEdge or Progress 138
 - Uninstall or Add/Remove Programs Utility 138 to 140
 - UNIX
 - default network type 326
 - environment variables 241, 242 to 244
 - starting network clients 327
 - Upgrading
 - an OpenEdge installation 138
 - User licenses, *See* Licenses
 - Utilities
 - BPRO 318, 323
 - MBPRO 331, 332
 - mergeprop 291 to 303
 - MPRO 330
 - PRO 318, 323
 - ProControl 332
 - product update 185
 - PROMON 508
 - PROSERVE 328
 - PROWDOG 508
- ## V
- Version number
 - displaying 187
 - Viewing registry information 120
 - Visual Translator
 - components of Complete installation 383
- ## W
- Watchdog utility 508
 - Web browser
 - requirements (Windows) 36
 - Web development
 - set up 115
 - Web server
 - requirements (Windows) 36
 - Web Services Adapter
 - components of Complete installation 385
 - WebSpeed
 - Configuration choices (Windows) 94
 - WebSpeed Messenger
 - components of Complete installation 386
 - WebSpeed Messenger, components installed during Complete installation 424
 - WebSpeed Workshop
 - components of Complete installation 387
 - Windows Explorer
 - integrating with OpenEdge 82 to 83
 - Workgroup Database, components installed during Complete installation 419
 - Workgroup RDBMS
 - components of Complete installation 377

Working directory
defined 75

Working Path
DCU 101

X

XML files
parsing 101