

Progress DataDirect® OpenAccess™ SDK 7.0

Embedding the OpenAccess™ SDK Server

June 2011

The logo consists of an orange rounded rectangle containing the text "PROGRESS" in a light orange color and "DATADIRECT" in white, stacked vertically.

PROGRESS
DATADIRECT

OVERVIEW	3
EMBEDDING SERVER IN C/C++ APPLICATIONS.....	3
C API.....	3
<i>server_start</i>	3
<i>server_stop</i>	5
SAMPLE FOR WINDOWS.....	5
<i>Files for embedding in C/C++ Applications</i>	5
<i>Compiling C/C++ Sample</i>	5
<i>Running C/C++ Sample</i>	6
SAMPLE FOR UNIX.....	6
<i>Files for embedding in C/C++ Applications</i>	6
<i>Building C/C++ Sample</i>	6
<i>Running C/C++ Sample</i>	6
<i>Remarks</i>	6
EMBEDDING SERVER IN JAVA APPLICATIONS.....	7
JAVA API.....	7
<i>oasload.OaServerControl</i>	7
SAMPLE FOR WINDOWS.....	9
<i>Files for embedding in Java Applications</i>	9
<i>Compiling the Java Sample</i>	9
<i>Running the Java Sample</i>	9
SAMPLE FOR UNIX.....	10
<i>Files for Embedding in Java Applications</i>	10
<i>Compiling the Java Sample</i>	10
<i>Running Java Sample in install_dir/shlibsrv</i>	10
REMARKS.....	10
DEPLOYING IN A JBOSS APPLICATION SERVER	11
INSTALLING THE JBOSS SAMPLE ON WINDOWS	11
RUNNING THE APPLICATION.....	11

Overview

This document describes how the OpenAccess SDK Server can be embedded in a native (C/C++) or a Java application.

The OpenAccess SDK Server can be run as a standalone process or it can be embedded into a C or Java process. The embeddable version of the OpenAccess SDK Server is supplied as a shared library (oasoa.dll or liboasoa.so) with an API that a C or Java program can use to start and stop the service. Code samples are provided for both C and Java environments. We also have samples for JBoss and other applications servers. If your environment is not described in this document, then please contact Progress customer support to find out if we have additional information in a form of a knowledgebase article.

Embedding Server in C/C++ Applications

C API

The embedded server exposes the functions `server_start` and `server_stop` listed below. They are defined in the header file `shlibsrv.h`. These functions can be accessed by dynamically loading the `oasoa.dll` (or `liboasoa.so`) shared library or by statically linking with the associated `lib` (or `.a`) file.

`server_start`

This function should be called from a thread in the application to start the specified OpenAccess SDK service that has already been configured. The required parameters are passed in as strings in an array of strings. This function will not return control back to the calling thread until either the service is stopped using the `server_stop` call or an issue occurs in starting the service. Once the service is started, it behaves just as if it was started as a standalone server.

```
int server_start(int argc, char * argv[])
```

Parameter	Type	Description
IN		
<code>argc</code>	<code>int</code>	Number of arguments passed into the <code>argv</code> string array
<code>argv[]</code>	<code>char *</code>	Array of strings that represent the arguments for the <code>server_start</code> command. <code>argv[0]</code> – a string indicating the name of the application <code>argv[1]</code> - <code>-datamodel</code> – indicates that the next argument specifies the path of the configuration file. <code>argv[2]</code> - <code><datamodel></code> where <code><datamodel></code> is replaced with the file name, including path, of the OpenAccess SDK Server configuration file <code>oadm.ini</code> . <code>argv[3]</code> - <code>-servicename</code> – indicates that the next argument specifies the OpenAccess SDK service name. <code>argv[4]</code> - <code><servicename></code> where <code><servicename></code> is replaced with the OpenAccess SDK service name that is configured in the configuration file specified by the <code>-datamodel</code> argument. For example, <code>OpenAccessSDK700_C</code> .

RETURN

- int
- 0 – Service has been started
 - All other return values – see the following table

Error Code	Reason
0	Success
10	oaso.dll is started without the required command arguments
11	The <code>-servicename</code> or <code>-n</code> option is missing servicename argument
12	The <code>-connectmodel</code> or <code>-m</code> option is missing an argument.
13	The <code>-sessionid</code> option is missing an argument.
14	The <code>-connectinfo</code> option is missing an argument.
15	The <code>-datamodel</code> or <code>-d</code> option is missing an argument.
16	The <code>-msgfile</code> or <code>-g</code> option is missing an argument.
17	The required servicename is missing.
18	Invalid value for connection model passed
21	Cannot open the message file
22	Cannot open the configuration file
23	Configuration file has incorrect version
24	Servicename name not found in configuration file
25	ServiceMessageFile attribute not set in configuration file
30	Failed to Build MGSS mechanism map from DM and create MGSS factory
31	Failed to initialize the EventQueue Cache
32	Failed to create the Core Service component
33	Failed to create the Scheduler component
34	Failed to create the Event Processor component
35	Failed to initialize the Service component
91	Internal error, memory allocation failed.
95	Internal error, fatal server error detected.

For both Windows and Linux/UNIX, when `server_start` returns with exit code 35, a related error message is written to a Service debug log file in the logging subdirectory.

Embedding the OpenAccess SDK Server

For all other exit codes:

On Windows, a related error message is sent to the Windows Application log. Use the Event Viewer to read the Windows Application log.

On Linux/UNIX, a related error message is sent to syslog marked at as level 'err'(LOG_ERR) and facility 'user'(LOGUSER). Verify the contents of /etc/syslog.conf where messages marked as level 'err' and facility 'user' are written.

Some typical locations are shown in the following table.

Operating System	Location of the syslog
Solaris	/var/adm/messages
Linux	/var/log/messages
HP/UX	/var/adm/syslog/syslog.log
AIX	as set in /etc/syslog.conf

server_stop

Call this function to stop the service. Calling this function causes the control from the server_start to be returned back to the thread that called it. The server_stop function places a stop command in the command queue of the service and returns a status code of 0.

```
int server_stop(void)
```

Parameter	Type	Description
-----------	------	-------------

RETURN

int	0 – Service stop command has been placed in the queue
-----	---

Sample for Windows

Files for embedding in C/C++ Applications

Note: The place holder *install_dir* should be replaced with the directory where you have installed the OpenAccess SDK Server and IP SDK.

install_dir\bin\oasoa.dll - OpenAccess SDK Server shared library

install_dir\shlibsrv\oac\shlibsrv.h - C/C++ Interface to OpenAccess Server Shared library

install_dir\shlibsrv\oac\oasrvw.c - Sample for Windows

Compiling C/C++ Sample

```
cd install_dir\shlibsrv
```

```
CALL ... VSvars32.BAT
```

```
CL oac\oasrvw.c
```

```
C:\Program Files\DataDirect\oaserver700\shlibsrv>CL oac\oasrvw.c
Microsoft (R) 32-bit C/C++ Optimizing Compiler Version 13.10.3077 for
80x86
```

```
Copyright (C) Microsoft Corporation 1984-2002. All rights reserved.
```

```
oasrvw.c
Microsoft (R) Incremental Linker Version 7.10.3077
Copyright (C) Microsoft Corporation. All rights reserved.

/out:oasrvw.exe
oasrvw.obj
```

Running C/C++ Sample

```
SET OAINSTALLDIR=D:\Program Files\DataDirect\oaserver700
cd %OAINSTALLDIR%\shlibsrv

PATH=%OAINSTALLDIR%\bin;%PATH%
oasrvw.exe -datamodel "%OAINSTALLDIR%\cfg\oadm.ini" -servicename
"OpenAccessSDK700_C"
```

Sample for UNIX

Files for embedding in C/C++ Applications

Note: Replace the place holder *install_dir* with the directory in which you have installed the OpenAccess SDK Server and IP SDK.

```
install_dir/bin/liboasoa.so : OpenAccess SDK Server shared library
install_dir/shlibsrv/oac/shlibsrv.h : C/C++ Interface to OpenAccess Server Shared library
install_dir/shlibsrv/oac/oasrvu.c : OpenAccess Server Shared library Sample for UNIX
install_dir/shlibsrv /makefile : make file
install_dir/shlibsrv /runsvr_c.sh : korn shell script starting the C/C++ sample
```

Building C/C++ Sample

- a. Type `cd install_dir`
- b. If you are using a C shell, type:

```
source ip/cfg/setenv.sh
```

If you are using a Korn shell:

```
. ./ip/cfg/setenv.sh
```

- c. Type `cd install_dir/shlibsrv`
- d. Type `make csample`

Running C/C++ Sample

```
cd install_dir/shlibsrv
runsvr_c.sh
```

By default, the `runsvr_c.sh` script starts the OpenAccess SDK C Service. To start the OpenAccess SDK Java Service, edit this script and set the `ServiceName` service attribute to `${SERVICENAME_JAVA}`:

Remarks

On HP Integrity, the thread stack size must be at least 416 KB to run Java. (see the *HP-UX Java SDK, RTE, and Plug-In Release Notes*). For this reason, the `oasrvu.c` sample calls the function `pthread_default_stacksize_np()` to increase the thread stack size on HP Integrity.

Embedding Server in Java Applications

Java API

The embedded server shared library exposes a native function `OaServerControl` that can be called from a Java program. The wrapper class `oasload` (found in `install_dir\shlibsrv\oajava\server\oasload.java`) defines the interface that can be used to access this function. The `OaServerControl` method is invoked to start and to stop a service. Example classes that demonstrate the embedding of the OpenAccess SDK server into a Java program are located in `install_dir\shlibsrv\oajava\server` directory. A Java application can embed an OpenAccess SDK Server to run C or Java services.

oasload.OaServerControl

This function should be called from a thread in the application to start the specified OpenAccess SDK service that has already been configured and to stop a service. The required parameters are passed in as strings in an array of strings. When called to start a service, this method will not return control back to the calling thread until the service is stopped from a different thread. Once the service is started it behaves just as if it was started as a standalone server.

```
int OaServerControl(int argc, String argv[], int command)
```

Parameter	Type	Description
IN		
<code>argc</code>	<code>int</code>	Number or arguments passed into the <code>argv</code> string array
<code>argv[]</code>	<code>char *</code>	<p>Array of strings that represent the arguments for the <code>server_start</code> command.</p> <p><code>argv[0]</code> – a string indicating the name of the application</p> <p><code>argv[1]</code> - <i>-datamodel</i> – indicates that the next argument specifies the path of the configuration file.</p> <p><code>argv[2]</code> - <code><datamodel></code> where <code><datamodel></code> is replaced with the file name, including path, of the OpenAccess SDK Server configuration file <code>oadm.ini</code>.</p> <p><code>argv[3]</code> - <i>-servicename</i> – indicates that the next argument specifies the OpenAccess service name.</p> <p><code>argv[4]</code> - <code><servicename></code> where <code><servicename></code> is replaced with the OpenAccess SDK service name that is configured in the configuration file specified by the <code>-datamodel</code> argument. For example, <code>OpenAccessSDK700_C</code>.</p>

command int Command:

- OASERVER_START (2) – start the service using the arguments passed in the `argv` string array.
- OASERVER_STOP (3) – stop the service
- -OASERVER_START_WITH_OBJECT (4) – start the service in Extended Mode . This command is valid only for Java Service.

RETURN

int

- 0 – For OASERVER_START, service has been started. For OASERVER_STOP, stop command has been placed in the service’s queue.
- All other return values for OASERVER_START and OASERVER_START_WITH_OBJECT – see the following table.

Error Code	Reason
0	Success
10	liboasoa.so is started without the required command arguments
11	the –servicename or -n option is missing servicename argument
12	the –connectmodel or -m option is missing an argument
13	the –sessionid option is missing an argument
14	the -connectinfo option is missing an argument
15	the –datamodel or –d option is missing an argument
16	the –msgfile or –g option missing an argument
17	missing the required servicename
18	invalid value for connection model passed
21	cannot open the message file
22	cannot open the configuration file
23	configuration file has incorrect version
24	Servicename name not found in configuration file
25	ServiceMessageFile attrbute not set in configuration file
30	Failed to Build MGSS mechanism map from DM and create MGSS factory
31	Failed to initialize the EventQueue Cache
32	Failed to create the Core Service component
33	Failed to create the Scheduler component
34	Failed to create the Event Processor component
35	Failed to initialize the Service component
91	Internal error, memory allocation failed.
95	Internal error, fatal server error detected.

For both Windows and Linux/UNIX, when OaServerControl returns with exit code 35, a related error message is written to a Service debug log file in the logging subdirectory

For all other exit codes:

On Windows, a related error message is sent to the Windows Application log. Use the Event Viewer to read the Windows Application log.

Embedding the OpenAccess SDK Server

On Linux/UNIX, a related error message is sent to syslog marked at as level 'err'(LOG_ERR) and facility 'user'(LOGUSER). Verify the contents of /etc/syslog.conf where messages marked at as level 'err' and facility 'user' are written.

Some typical locations are shown in the following table.

Operating System	Location of the syslog
Solaris	/var/adm/messages
Linux	/var/log/messages
HP/UX	/var/adm/syslog/syslog.log
AIX	as set in /etc/syslog.conf

Sample for Windows

Files for embedding in Java Applications

NOTE: Replace the place holder *install_dir* with the directory in which you have installed the OpenAccess SDK Server and IP SDK.

install_dir\bin\oasoa.dll : OpenAccess SDK Server shared library
install_dir\shlibsrv\oajava\server\oasload.java : OpenAccess SDK embedded Server load class
install_dir\shlibsrv\oajava\server\oaserver.java : OpenAccess SDK embedded Server Class
install_dir\shlibsrv\oajava\server\oaservertest.java : OpenAccess SDK embedded Server sample

Compiling the Java Sample

```
set OAINSTALLDIR=C:\Program Files\DataDirect\oaserver700
cd %OAINSTALLDIR%\shlibsrv
set JVM=D:\Tools\j2sdk1.4.2_07
```

```
%JVM%\bin\javac -d . oajava\server\*.java
```

Running the Java Sample

```
set OAINSTALLDIR=C:\Program Files\DataDirect\oaserver700
cd %OAINSTALLDIR%\shlibsrv
set JVM=D:\Tools\j2sdk1.4.2_07
PATH=%OAINSTALLDIR%\bin;%PATH%
```

```
%JVM%\bin\java -server -classpath . oaservertest -datamodel
"%OAINSTALLDIR%\cfg\oadm.ini" -servicename "OpenAccessSDK700_C"
```

When running a service for a Java IP, add the class path from ServiceJVMClassPath to the CLASSPATH variable.

```
SET CLASSPATH=.;%OAINSTALLDIR%\ip\oajava\oasql.jar;%OAINSTALLDIR%\ip
%JVM%\bin\java -server oaservertest -datamodel
"%OAINSTALLDIR%\cfg\oadm.ini" -servicename "OpenAccessSDK700_Java"
```

In addition, use the same JVM that was set in the ServiceJVMLocation attribute.

Sample for UNIX

Files for Embedding in Java Applications

NOTE: The place holder *install_dir* should be replaced with the directory where you have installed the OpenAccess SDK Server and IP SDK.

install_dir/bin/liboasoa.so : OpenAccess SDK Server shared library
install_dir/shlibsrv/oajava/oaserver/oasload.java : OpenAccess SDK embedded Server load class
install_dir/shlibsrv/oajava/oaserver/oaserver.java : OpenAccess SDK embedded Server Class
install_dir/shlibsrv/oajava/oaserver/oaservertest.java : OpenAccess SDK embedded Server sample
install_dir/shlibsrv /makefile : make file
install_dir/shlibsrv /runsvr_java.sh : korn shell script starting the Java sample

Compiling the Java Sample

1. Type `cd install_dir`
2. If you are using a C shell, type:

```
source ip/cfg/setenv.sh
```

If you are using a Korn shell, type:

```
. ./ip/cfg/setenv.sh
```

3. Type `cd install_dir/shlibsrv`
4. Type `make javasample`

Running Java Sample in install_dir/shlibsrv

```
cd install_dir/shlibsrv
runsvr_java.sh
```

By default, the `runsvr_java.sh` script will start the OpenAccess SDK C Service. To start the OpenAccess SDK Java Service, edit this script and set the `ServiceName` service attribute to `$(SERVICENAME_JAVA)`:

Remarks

1. When embedding the server in Java, the values of the service attributes `ServiceJVMClassPath` and `ServiceJVMOptions` have no effect. These values must be passed directly to the Java program.
2. Use the JVM that was set in `ServiceJVMLocation`.

Deploying in a JBoss Application Server

Installing the JBoss Sample on Windows

Install Embedded Server components

1. Extract oaserver32_embedded_win.zip file at the location where the OpenAccess SDK Server and IP SDK is installed – for example, c:\Program Files\DataDirect\oaserver700.

Configure JBoss

1. Deploy the OpenAccess WAR application by extracting attached oa_jboss_deploy.zip under the folder where JBoss is installed – for example, C:\Program Files\jboss-4.2.2.GA.
2. Edit C:\Program Files\jboss-4.2.2.GA\bin\openaccess.bat to set the following:
 - OpenAccess SDK install folder
 - Java Home
 - JBOSS_CLASSPATH – should include the classes required by OpenAccess SDK and the classes required by your IP code (for Java IPs)
3. PATH – update it to include the location of the oasoa.dll file – install_dir\bin
Run JBoss using openaccess.bat.

Running the Application

1. Access <http://localhost:8080/openaccess/oaservice.html>.
2. Select the Service to Start, Select the Start option and then click **GO**.
3. A status screen is displayed.
4. You can verify from the OpenAccess SDK Management Console that the Service is started. Currently, the JBoss application always says the service is Started.
5. To stop the Service, you must reload oaservice.html (step 1) and select the Stop option and click **GO**. It will show the status correctly indicating whether the service was stopped. You can confirm this by running the OpenAccess SDK Management console. If you are already running the Management Console, select the OpenAccess SDK service you are working with and press **F5** to refresh its status.

NOTE: You can stop only the last service that was started.

Copyright © 2011. Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved. Actional, Apama, Artix, Business Empowerment, DataDirect (and design), DataDirect Connect, DataDirect Connect64, DataDirect Technologies, DataDirect XML Converters, DataDirect XQuery, DataXtend, Dynamic Routing Architecture, EdgeXtend, Empowerment Center, Fathom, Fuse Mediation Router, Fuse Message Broker, Fuse Services Framework, IntelliStream, IONA, Making Software Work Together, Mindreef, ObjectStore, OpenEdge, Orbix, PeerDirect, POSSENET, Powered by Progress, PowerTier, Progress, Progress DataXtend, Progress Dynamics, Progress Business Empowerment, Progress Empowerment Center, Progress Empowerment Program, Progress OpenEdge, Progress Profiles, Progress Results, Progress Software Developers Network, Progress Sonic, ProVision, PS Select, Savvion, SequeLink, Shadow, SOAPscope, SOAPStation, Sonic, Sonic ESB, SonicMQ, Sonic Orchestration Server, 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, ASPen, ASP-in-a-Box, BusinessEdge, Business Making Progress, Cache-Forward, DataDirect Spy, DataDirect SupportLink, Fuse, FuseSource, Future Proof, GVAC, High Performance Integration, ObjectStore Inspector, ObjectStore Performance Expert, OpenAccess, Orbacus, Pantero, POSSE, ProDataSet, Progress Control Tower, Progress ESP Event Manager, Progress ESP Event Modeler, Progress Event Engine, Progress RFID, Progress RPM, Progress Software Business Making Progress, PSE Pro, SectorAlliance, SeeThinkAct, Shadow z/Services, Shadow z/Direct, Shadow z/Events, Shadow z/Presentation, Shadow Studio, SmartBrowser, SmartComponent, SmartDataBrowser, SmartDataObjects, SmartDataView, SmartDialog, SmartFolder, SmartFrame, SmartObjects, SmartPanel, SmartQuery, SmartViewer, SmartWindow, Sonic Business Integration Suite, Sonic Process Manager, Sonic Collaboration Server, Sonic Continuous Availability Architecture, Sonic Database Service, Sonic Workbench, Sonic XML Server, The Brains Behind BAM, 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 and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. Any other trademarks contained herein are the property of their respective owners.

Third Party Acknowledgments:

Progress OpenAccess v7.0 incorporates OpenLDAP (v.2.2.6) library. Such technology is subject to the following terms and conditions: The OpenLDAP Public License Version 2.2, 1 March 2000 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 "OpenLDAP" must not be used to endorse or promote products derived from this Software without prior written permission of the OpenLDAP Foundation. 4. Products derived from this Software may not be called "OpenLDAP" nor may "OpenLDAP" appear in their names without prior written permission of the OpenLDAP Foundation. 5. Due credit should be given to the OpenLDAP Project (<http://www.openldap.org/>). 6. The OpenLDAP Foundation may revise this license from time to time. Each revision is distinguished by a version number. You may use the Software under terms of this license revision or under the terms of any subsequent the license. THIS SOFTWARE IS PROVIDED BY THE OPENLDAP FOUNDATION 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 THE OPENLDAP 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. OpenLDAP is a trademark of the OpenLDAP Foundation. Copyright 1999-2000, The OpenLDAP Foundation, Redwood City, California, USA. All Rights Reserved. Permission to copy and distributed verbatim copies of this document is granted.

Progress OpenAccess v7.0 incorporates OpenSSL toolkit version 1.0.0. Such technology is subject to the following terms and conditions: LICENSE ISSUES ===== The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org. OpenSSL License ----- Copyright (c) 1998-2011 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

Embedding the OpenAccess SDK Server

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. ===== This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com). - Original SSLeay License ----- Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com) All rights reserved. This package is an SSL implementation written by Eric Young (eay@cryptsoft.com). The implementation was written so as to conform with Netscapes SSL. This library is free for commercial and non-commercial use as long as the following conditions are aheared to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, 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 rouines 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 OpenAccess v7.0 incorporates ICU version 4.2.1. 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.