



# Orbix®

---

## Actional Integration Guide

Version 6.3, October 2009

© 2009 Progress Software Corporation and/or its affiliates or subsidiaries. All rights reserved.

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

---

Actional, Apama, Apama (and Design), 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, IntelliStream, IONA, IONA (and design), 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, SequeLink, Shadow, SOAPscope, SOAPStation, Sonic, Sonic ESB, SonicMQ, Sonic Orchestration Server, SonicSynergy, 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, Fuse Mediation Router, Fuse Message Broker, Fuse Services Framework, Future Proof, GVAC, High Performance Integration, ObjectStore Inspector, ObjectStore Performance Expert, OpenAccess, Orbacus, Pantero, POSSE, ProDataSet, Progress ESP Event Manager, Progress ESP Event Modeler, Progress Event Engine, Progress RFID, 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 marks, StormGlass, The Brains Behind BAM, WebClient, Who Makes Progress, and Your World. Your SOA. are trademarks or service marks of Progress Software Corporation or one of its affiliates or subsidiaries 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.

Updated: November 6, 2009

### Third Party Acknowledgments:

(1) The IDL compiler feature of Orbix 6.3.4 incorporates IDL Compiler Front End 1 from Sun Microsystems, Inc. Copyright 1992, 1993, 1994 Sun Microsystems, Inc. Printed in the United States of America. All Rights Reserved. Such technology is subject to the following terms and conditions:

This product is protected by copyright and distributed under the following license restricting its use. The Interface Definition Language Compiler Front End (CFE) is made available for your use provided that you include this license and copyright notice on all media and documentation and the software program in which this product is incorporated in whole or part. You may copy and extend functionality (but may not remove functionality) of the Interface Definition Language CFE without charge, but you are not authorized to license or distribute it to anyone else except as part of a product or program developed by you or with the express written consent of Sun Microsystems, Inc. ("Sun"). The names of Sun Microsystems, Inc. and any of its subsidiaries or affiliates may not be used in advertising or publicity pertaining to distribution of Interface Definition Language CFE as permitted herein. This license is effective until terminated by Sun for failure to comply with this license. Upon termination, you shall destroy or return all code and documentation for the Interface Definition Language CFE.

INTERFACE DEFINITION LANGUAGE CFE IS PROVIDED AS IS WITH NO WARRANTIES OF ANY KIND INCLUDING THE WARRANTIES OF DESIGN, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR ARISING FROM A COURSE OF DEALING, USAGE OR TRADE PRACTICE. INTERFACE DEFINITION LANGUAGE CFE IS PROVIDED WITH NO SUPPORT AND WITHOUT ANY OBLIGATION ON THE PART OF Sun OR ANY OF ITS SUBSIDIARIES OR AFFILIATES TO ASSIST IN ITS USE, CORRECTION, MODIFICATION OR ENHANCEMENT. Sun OR ANY OF ITS SUBSIDIARIES OR AFFILIATES SHALL HAVE NO LIABILITY WITH RESPECT TO THE INFRINGEMENT OF COPYRIGHTS, TRADE SECRETS OR ANY PATENTS BY INTERFACE DEFINITION LANGUAGE CFE OR ANY PART THEREOF. IN NO EVENT WILL Sun OR ANY OF ITS SUBSIDIARIES OR AFFILIATES BE LIABLE Sun ANY LOST REVENUE OR PROFITS OR OTHER SPECIAL, INDIRECT AND CONSEQUENTIAL DAMAGES, EVEN IF Sun HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 and FAR 52.227-19. Sun, Sun Microsystems and the Sun logo are trademarks or registered trademarks of Sun Microsystems, Inc. SunSoft, Inc. 2550 Garcia Avenue, Mountain View, California 94043 NOTE: SunOS, SunSoft, Sun, Solaris, Sun Microsystems or the Sun logo are trademarks or registered trademarks of Sun Microsystems, Inc.

(2) The administrator web console feature of Orbix 6.3.4 incorporates Jakarata-struts 1.0.2 from the Apache Software Foundation (<http://www.apache.org>). Such Apache Technology is subject to the following terms and conditions:

The Apache Software License, Version 1.1 Copyright (c) 1999-2001 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 "The Jakarta Project", "Struts", 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](mailto: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/>.

(3) The Java Class file analyser feature of Orbix 6.3.4 incorporates Jakarta-bcel 5.0 from the Apache Software Foundation (<http://www.apache.org>). Such Apache Technology is subject to the following terms and conditions:

The Apache Software License, Version 1.1 Copyright (c) 2001 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" and "Apache BCEL" 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](mailto:apache@apache.org).

5. Products derived from this software may not be called "Apache", "Apache BCEL", 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/>.

(4) The Java Class file analyser feature of Orbix 6.3.4 incorporates Jakarat-regexp 1.2 from the Apache Software Foundation (<http://www.apache.org>). Such Apache 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 "The Jakarta Project", "Jakarta-Regexp", and "Apache Software Foundation" and "Apache BCEL" 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](mailto: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/>.

(5) The logging feature of Orbix 6.3.4 incorporates the Jakarta-log4j 1.2.6 from the Apache Software Foundation (<http://www.apache.org>). Such Apache 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 "log4j" and "Apache Software Foundation" and "Apache BCEL" 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](mailto: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>.

(6) The demo feature of Orbix 6.3.4 incorporates Ant 1.5 from the Apache Software Foundation (<http://www.apache.org>). Such technology is subject to the following terms and conditions:

The Apache Software License, Version 1.1 Copyright (c) 2000-2002 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" and "Apache BCEL" 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](mailto: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/>.

(7) The xsume feature of Orbix 6.3.4 incorporates Xalan-j 2.3.1 from the Apache Software Foundation (<http://www.apache.org>). Such Apache 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 "Xalan" and "Apache Software Foundation" and "Apache BCEL" 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](mailto: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/>.

(8) The security service feature of Orbix 6.3.4 incorporates the Xerces-c++ 2.4 from the Apache Software Foundation (<http://www.apache.org>). Such Apache Technology is subject to the following terms and conditions:

The Apache Software License, Version 1.1. Copyright (c) 1999-2001 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" and "Apache BCEL" 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](mailto: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/>.

(9) The xsume feature of Orbix 6.3.4 incorporates xerces-j 2.5 from the Apache Software Foundation (<http://www.apache.org/>). Such Apache Technology is subject to the following terms and conditions:

The Apache Software License, Version 1.1. Copyright (c) 1999-2002 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](mailto: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/>.



(10) The xsume feature of Orbix 6.3.4 incorporates the Tomcat 4.0.4 from the Apache Software Foundation (<http://www.apache.org>). Such Apache Technology is subject to the following terms and conditions: The Apache Software License, Version 1.1. Copyright (c) 1999, 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 "The Jakarta Project", "Tomcat" 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](mailto: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/>.

(11) The xsume feature of Orbix 6.3.4 incorporates the MinML 1.7 from John Wilson. Such Technology is subject to the following terms and conditions:

Copyright (c) 1999, John Wilson ([tug@wilson.co.uk](mailto:tug@wilson.co.uk)). 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. All advertising materials mentioning features or use of this software must display the following acknowledgement:

This product includes software developed by John Wilson. The name of John Wilson may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY JOHN WILSON "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 JOHN WILSON 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.

(12) The codeset negotiation feature of Orbix 6.3.4 incorporates ICU library 2.6 from IBM. Such Technology is subject to the following terms and conditions:

Copyright (c) 1995-2009 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.

(13) The configuration deployer feature of Orbix 6.3.4 incorporates JDOM vbeta9 from JDOM. Such Technology is subject to the following terms and conditions:

LICENSE.txt, v 1.10 2003/04/10 08:36:05 jhunter Exp \$ Copyright (C) 2000-2003 Jason Hunter & Brett McLaughlin. 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 disclaimer that follows these conditions in the documentation and/or other materials provided with the distribution.
3. The name "JDOM" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact [license@jdom.org](mailto:license@jdom.org).

4. Products derived from this software may not be called "JDOM", nor may "JDOM" appear in their name, without prior written permission from the JDOM Project Management (pm@jdom.org). In addition, we request (but do not require) that you include in the end-user documentation provided with the redistribution and/or in the software itself an acknowledgement equivalent to the following: "This product includes software developed by the JDOM Project (<http://www.jdom.org/>)." Alternatively, the acknowledgment may be graphical using the logos available at <http://www.jdom.org/images/logos>.

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 JDOM AUTHORS OR THE PROJECT 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 JDOM Project and was originally created by Jason Hunter (jhunter@jdom.org) and Brett McLaughlin (brett@jdom.org). For more information on the JDOM Project, please see <http://www.jdom.org/>.

(14) The Secure Socket Layer (SSL) feature of Orbix 6.3.4 incorporates OpenSSL 0.9.8i Copyright (c) 1998-2008 The OpenSSL Project Copyright (c) 1995-1998 Eric A. Young, Tim J. Hudson All rights reserved. Such Technology is subject to the following terms and conditions:

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) 1998-2008 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.

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 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 publicly 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).

(15) The C preprocessor feature of Orbix 6.3.4 incorporates MCPP 2.6.4 from the MCPP Project. Such technology is subject to the following terms and conditions: Copyright (c) 1998, 2002-2007 Kiyoshi Matsui (kmatsui@t3.rim.or.jp). All rights reserved. This software including the files in this directory is provided under the following license. 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.



# Contents

<b>List of Figures</b>	<b>17</b>
<b>Preface</b>	<b>19</b>
<b>Chapter 1 Orbix–Actional Integration</b>	<b>23</b>
Introduction	24
Orbix–Actional Integration Architecture	29
<b>Chapter 2 Configuring Orbix for Actional Integration</b>	<b>35</b>
Configuring an Orbix Domain	36
Configuring Orbix Java Applications	40
Configuring Orbix C++ applications	44
Running the enable_actional Script	47
Troubleshooting Orbix	49
<b>Chapter 3 Configuring Actional for Orbix Integration</b>	<b>51</b>
Prerequisites	52
Configuring Actional	53
Troubleshooting Actional	57
<b>Chapter 4 Managing Orbix Applications in Actional</b>	<b>61</b>
Monitoring Orbix Applications	62
Monitoring Orbix Domain Services	67
Auditing Orbix Applications	70
<b>Glossary</b>	<b>75</b>
<b>Index</b>	<b>83</b>

## CONTENTS



# List of Figures

Figure 1: High-Level Actional Overview	25
Figure 2: Actional Management Server Administration Console	27
Figure 3: Basic Actional Architecture	29
Figure 4: Actional Interceptors	30
Figure 5: Orbix–Actional Integration Architecture	33
Figure 6: Creating a Domain in Expert Mode	36
Figure 7: Selecting Services	37
Figure 8: Specifying Actional Monitoring	38
Figure 9: Actional Agent Administration Console	41
Figure 10: Actional Server Configuration Settings	54
Figure 11: Actional Server Provisioned Node	56
Figure 12: Actional Agent Options	58
Figure 13: Actional Agent Event Logs	59
Figure 14: Actional Server Network View	63
Figure 15: Traffic Between Packages	63
Figure 16: Actional Server Path Explorer	64
Figure 17: Actional Server Statistics Details	65
Figure 18: Actional Server Statistics Chart	66
Figure 19: Traffic Between Domain Services Packages	68
Figure 20: Domain Services in Path Explorer	68
Figure 21: Node Daemon Operation	69
Figure 22: Actional Policy Groups	70
Figure 23: Actional Demo Audit Logs	71
Figure 24: Actional Demo Audit Log Record	72
Figure 25: Domain Services Audit Logs	73
Figure 26: Node Daemon Log Record	74

## LIST OF FIGURES

# Preface

## What is covered in this book

Orbix supports integration with the Actional SOA management system. This guide explains how to enable Orbix applications and services to be monitored by Actional SOA management tools. This guide applies to Orbix applications and services written in both Java and C++.

## Who should read this book

This guide is aimed at Orbix system administrators using Actional to monitor SOA environments, Orbix system architects, and Orbix application developers. System administrators do not require detailed knowledge of the technology used to create distributed enterprise applications.

## Organization of this book

This book contains the following chapter:

- [Chapter 1](#) describes the architecture of the Orbix integration with Actional.
- [Chapter 2](#) explains how to configure integration between Orbix applications and services, and Actional.
- [Chapter 3](#) provides some basic Actional configuration guidelines.
- [Chapter 4](#) shows examples of managing Orbix applications and services in Actional SOA management tools.

## Related documentation

The Orbix documentation also includes the following related guides:

- *Orbix Administrator's Guide*
- *Orbix Configuration Reference*
- *Orbix Deployment Guide*
- *Orbix Management User's Guide*
- *Orbix Management Programmer's Guide*

## Additional Resources

- The most up-to-date versions of Orbix technical documentation are available from:  
<http://web.progress.com/orbix/support/6.3.4/>
- The Orbix Knowledge Base is a database of articles that contain practical advice on specific development issues, contributed by developers, support specialists, and customers. This is available from:  
<http://www.progress.com/orbix/orbix-support.html>
- Contact Orbix technical support at:  
<http://www.progress.com/orbix/orbix-support.html>

## Document Conventions

This guide uses the following typographical conventions:

Constant width	<p>Constant width font in normal text represents commands, portions of code and literal names of items (such as classes, functions, and variables). For example, constant width text might refer to the <code>itadmin orbname create</code> command.</p> <p>Constant width paragraphs represent information displayed on the screen or code examples. For example the following paragraph displays output from the <code>itadmin orbname list</code> command:</p> <pre>ifr naming production.test.testmgr production.server</pre>
<i>Italic</i>	<p>Italic words in normal text represent emphasis and new terms (for example, <i>location domains</i>).</p>

<i>Code italic</i>	Italic words or characters in code and commands represent variable values you must supply; for example, process names in your <i>particular</i> system:  <code>itadmin process create <i>process-name</i></code>
<b>Code bold</b>	Code bold font is used to represent values that you must enter at the command line. This is often used in conjunction with constant width font to distinguish between command line input and output. For example:  <code><b>itadmin process list</b></code> <code>ifr</code> <code>naming</code> <code>my_app</code>

The following keying conventions are observed:

No prompt	When a command's format is the same for multiple platforms, a prompt is not used.
%	A percent sign represents the UNIX command shell prompt for a command that does not require root privileges.
#	A number sign represents the UNIX command shell prompt for a command that requires root privileges.
>	The notation > represents the DOS or Windows command prompt.
...	Horizontal ellipses in format and syntax descriptions indicate that material has been eliminated to simplify a discussion.
[ ]	Italicized brackets enclose optional items in format and syntax descriptions.
{ }	Braces enclose a list from which you must choose an item in format and syntax descriptions.
	A vertical bar separates items in a list of choices. Individual items can be enclosed in {} (braces) in format and syntax descriptions.



# Orbix–Actional Integration

*Orbix provides support for integration with Actional SOA management products. This chapter explains the main components and concepts used in this integration.*

---

## **In this chapter**

This chapter includes the following section:

<a href="#">Introduction</a>	<a href="#">page 24</a>
<a href="#">Orbix–Actional Integration Architecture</a>	<a href="#">page 29</a>

# Introduction

---

## Overview

Actional is a SOA management product that provides operational and business visibility, policy-based security, and control of services and business processes in a heterogeneous runtime environment. This section explains the main concepts and components used in the Orbix–Actional integration.

---

## Orbix and Actional

Integration between Orbix and Actional enables Orbix applications to be monitored by Actional SOA management tools. For example, you can use Actional to perform discovery, monitoring, auditing, and reporting on Orbix applications. You can also correlate and track all messages through your SOA network to perform dependency mapping and root cause analysis.

The Orbix–Actional integration is deployed on Orbix systems to enable reporting of management data back to the Actional server. The data reported back to Actional includes system administration metrics such as response time, fault location, auditing, and alerts based on policies and rules. The Orbix–Actional integration can be used with Orbix applications written in both Java and C++.

---

## Actional SOA management

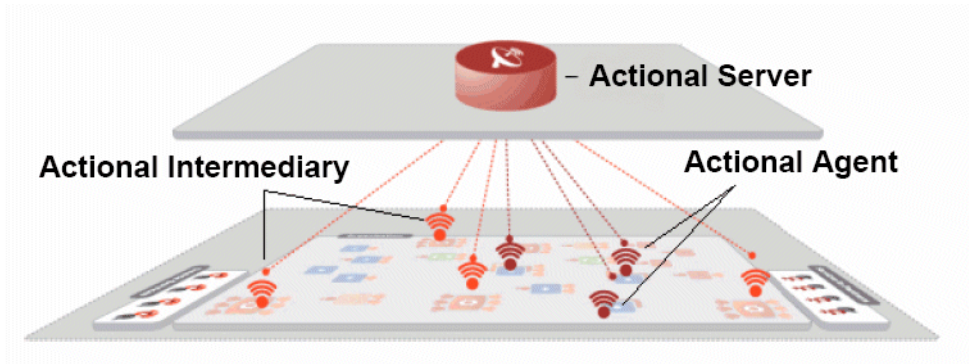
The main components in the Actional SOA management system are the Actional server, Actional agents, and Actional intermediaries.

The Actional server is the central engine that correlates data received from Actional agents and distributes policies. The Actional agent collects data about service traffic from an application server and applies policies. The Actional intermediary acts as a proxy that brokers interaction between Web service applications and systems built on them.

All Actional components are Java applications. The Actional server uses the Jetty application server by default, while its web console uses JSP and Adobe Flash.



Figure 5 shows a high-level overview of the main Actional components.



**Figure 1:** *High-Level Actional Overview*

---

#### Managed nodes

A node is defined as a system on the current network. A node with an Actional agent installed is referred to as an *instrumented node* or a *managed node*.

The managed node uses Actional's interceptor API to send monitoring data to the Actional agent. On any managed node, one Actional agent and one or more interceptors must be running.

---

#### Actional server

The Actional server is a central management server that manages nodes containing an Actional agent. The Actional server correlates the data it receives from each of its agents, and distributes policies to those agents. It enables an administrator to analyze service network data and create system-wide policies.

The Actional server hosts a database and pings Actional agents to obtain management data at configured time intervals. It analyzes the management data and displays it in a console—for example, the **Actional Management Server Administration Console**. This is a Web application deployed on Apache Tomcat, which provides runtime management and agent configuration. In addition, any alerts triggered at the Actional agent are sent immediately to the Actional server.

The default Actional server database is Apache Derby. Other supported databases include:

- PostgreSQL
- OpenEdge
- MSDE
- SQL Server
- Oracle
- DB2

By default, the Actional server uses port 4040 (for example, `http://HostName:4040/lgserver/`).

---

### Actional agent

An Actional agent is run on each Orbix host that you wish to manage, and is used to provide instrumentation data back to the Actional server. The Actional agent includes two main components: an analyzer, and one or more interceptors. The analyser gathers and evaluates data such as records, statistics, and alerts. The interceptors collect data about service traffic from an application server, and apply policies to that traffic.

Actional agents are provisioned from the Actional server to establish initial contact and send configuration to the Actional agent. There is one Actional agent per managed node. By default, the Actional agent uses port 4041 (for example, `http://HostName:4041/lgagent/`).

---

### Actional intermediary

An Actional intermediary is an in-network service broker that includes an integrated Actional agent. It serves as a proxy for Web service applications, providing features such as security, bridging, and activity tracking. The Actional intermediary supports application servers such as WebLogic, WebSphere, JBoss, and Oracle.

---

### Actional agent interceptor SDK

The Actional Agent Interceptor Software Development Kit (SDK) is an Actional-specific API used to create custom interceptors. These can be used to send management instrumentation data from an application to the Actional agent.

## Actional SOA management tools

In this guide, Actional is the general term used to describe the Actional SOA management system in which all data is stored and viewed. This simplifies the architecture of Actional for the sake of this discussion.

Figure 2 shows an example of the **Actional Management Server Administration Console**. Managed nodes are displayed as blue boxes, and unmanaged nodes are displayed as grey boxes. The green arrows indicate the message flow through various nodes. Clicking on each of the nodes shows more in-depth information regarding the response time, alerts and warnings, and so on.

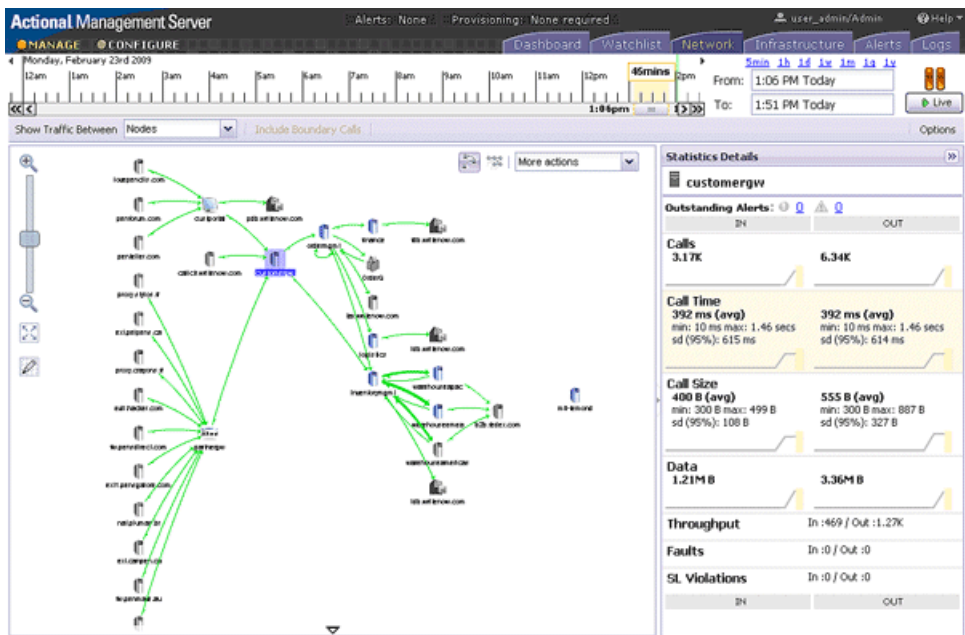


Figure 2: Actional Management Server Administration Console

---

**NGSO mapping**

When you click and drill down in the Actional **Path Explorer** view, the organization of the information displayed is *Node-Group-Service-Operation* (NGSO). In Orbix, this translates to *Host-Module-Interface-Operation*. [Table 1](#) shows the mapping from Actional to Orbix.

**Table 1:** *NGSO Mapping*

Actional	Orbix
Node	Host
Group	Module
Service	Interface
Operation	Operation

---

**Further information**

For detailed information on all Actional features, see the Actional product documentation.

# Orbix-Actional Integration Architecture

## Overview

This section shows a basic Actional architecture, simplified for the purposes of this discussion. It explains how Actional interceptors provide data to the Actional agent, and how the Actional server manifest is used to correlate the origin and business flow of a request.

It then shows the Orbix-Actional integration architecture, and explains how Orbix plug-ins and Orbix interceptors are used to configure integration with Actional.

## Basic Actional architecture

Figure 3 shows a high-level overview of a basic Actional architecture from the perspective of a consumer and service provider.

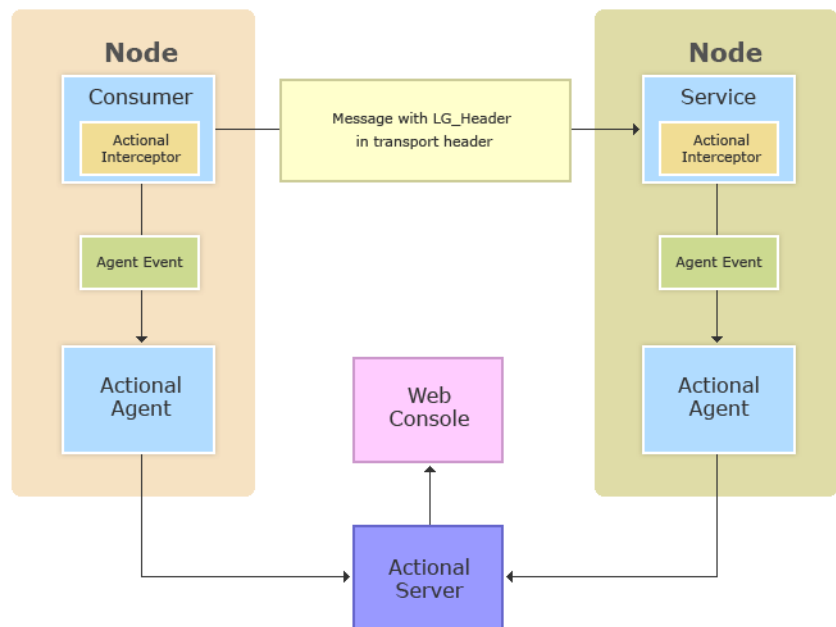


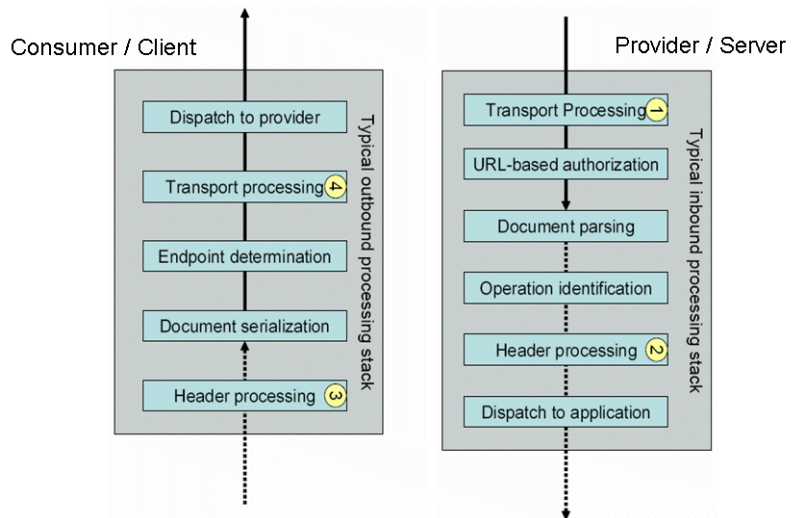
Figure 3: Basic Actional Architecture

In the interaction shown in [Figure 3](#), the Actional interceptors sit in the flow between the application logic and the consumers and providers of other services. They intercept all inbound and outbound calls, and feed information about those calls to the Actional agent as asynchronous events. The Actional agent is responsible for processing the event stream from the interceptors, computing and storing aggregate statistics, executing policies, and communicating with the Actional server.

The Actional server manifest (`LG_Header`) is a token that is sent in the transport header of the message to each participant in a call. This token identifies the origin and business flow of a request. For more details, see [“Actional server manifest” on page 31](#).

### Actional interceptors

Actional interceptors sit in the flow at the edge of an application, intercepting all incoming and outgoing messages. An Actional interceptor is designed as a lightweight component that imposes minimal overhead on the application (typically less than 100 microseconds per call).



**Figure 4:** *Actional Interceptors*

The interceptor must perform the following tasks to gain the full functionality of the Actional server:

1. Extract an Actional server manifest (if any) from the incoming request document.
2. Insert an Actional server manifest into any outgoing request documents.
3. Transfer the interceptor context along the internal business flow, from the incoming interceptor, to any related outgoing interceptors.
4. Send the Actional agent an event for each incoming or outgoing document.

## Actional server manifest

The Actional server sends an Actional server manifest (`LG_Header`) with a request document to provide information about the request's origin and the business flow that the request belongs to.

The Actional server manifest is used by the Actional server to correlate information it receives, from multiple agents, about interactions between different services. For this reason, the server manifest is sometimes referred to as a correlation ID.

The consumer and provider of the service must have an agreed mechanism (transport or protocol) for transferring the manifest. The following is an example `LG_Header`:

```
Interaction=CgJkcB+YlN0ZyBABdysAAA==;
Locus=ApM1eYGBAR4LFJlVvHOdg==;
Flow=CgJkcB+YlN0ZyBABdSsAAA==;
UpstreamOpID=FtfeJXMlnqJ0C995IBmkEQ==;
Path=7Qg2aVWCdwmP8gGebyLWYA==;
name=E_10-2-100-112-e0c7c3-110c80b4df0--7fdd-INITIATED;
CPTime=1171591682345;
FlowFields=MF1:1254;MF2:1589;
```

The main components in the server manifest are the `Interaction`, `Locus`, `Flow`, and `UpstreamOpID`. The other components are optional.

---

**Orbix–Actional integration architecture**

The Orbix–Actional integration is built using the extensible Orbix plug-in architecture. This means that Orbix–Actional integration can be enabled by adding a monitoring plug-in to your Orbix configuration. No code changes are necessary for Orbix client and server applications.

Figure 5 shows an overview of the Orbix–Actional integration architecture from an Orbix client-server perspective. This builds on the architecture shown in Figure 3, with the addition of Orbix monitoring and GIOP plug-ins. In Figure 5, the CORBA GIOP message also includes the `LG_Header` in a GIOP service context. A GIOP service context is a general mechanism for including out-of-band data in a GIOP request or reply message. Service contexts in GIOP are analogous to headers in other protocols such as HTTP.

---

**Orbix interceptors**

In the Orbix–Actional integration, Orbix interceptors for Actional must also be added to your Orbix client and server binding lists. Orbix interceptors are objects that ORB services and transports implement to process operation invocations. Orbix interceptors are arranged in a chain, with each interceptor caching a reference to the next interceptor in the chain.

The Orbix monitoring plug-in is implemented as a *request-level interceptor*. This receives a request in the form of a request object from the preceding interceptor in the chain. This enables high-level request processing to be performed. In CORBA, a *binding* is a set of interceptors used to process requests.

---

**Further information**

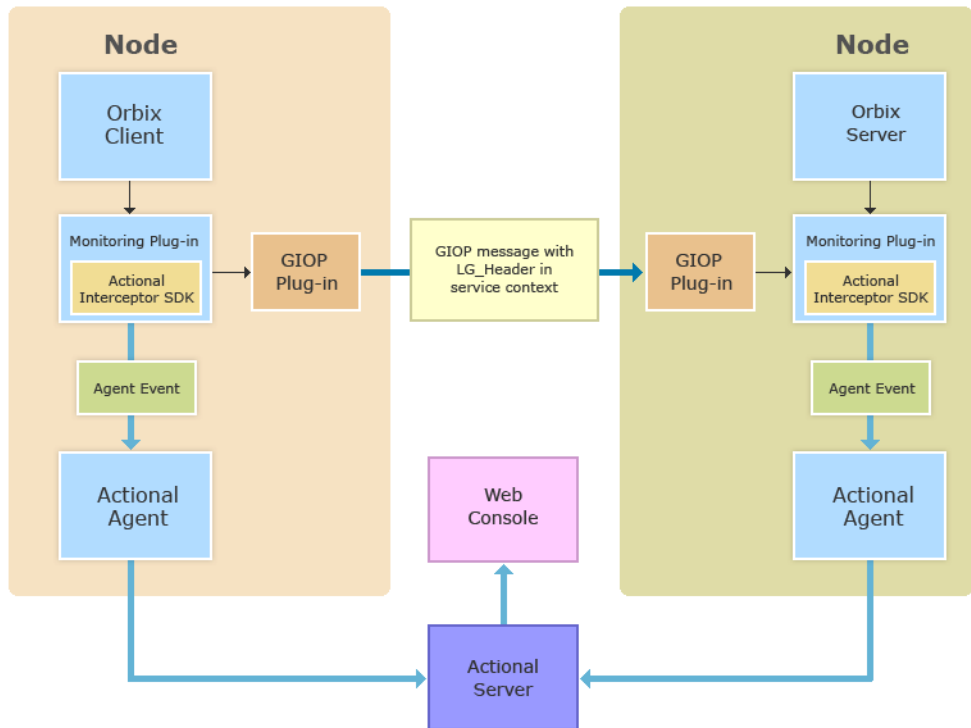
For detailed information on Actional architecture and components, see the Actional product documentation.

For details on how to configure the Orbix plug-in and interceptors for Orbix–Actional integration, see [Chapter 2](#).

For detailed information on Orbix interceptors, see:

- *Orbix Configuration Reference*
- *Orbix C++ Programmer's Guide*
- *Orbix Java Programmer's Guide*





**Figure 5:** *Orbix-Actional Integration Architecture*



# Configuring Orbix for Actional Integration

*This chapter explains the steps required to configure Orbix for integration with Actional SOA management products.*

---

**In this chapter**

This chapter includes the following sections:

<a href="#">Configuring an Orbix Domain</a>	<a href="#">page 36</a>
<a href="#">Configuring Orbix Java Applications</a>	<a href="#">page 40</a>
<a href="#">Configuring Orbix C++ applications</a>	<a href="#">page 44</a>
<a href="#">Running the enable_actional Script</a>	<a href="#">page 47</a>
<a href="#">Troubleshooting Orbix</a>	<a href="#">page 49</a>

# Configuring an Orbix Domain

## Overview

This section explains how to use the **Orbix Configuration** tool to enable an Orbix configuration domain for Actional integration. It shows how to configure and deploy your Orbix domain services with the Orbix configuration settings required for monitoring by Actional. For example, Orbix domain services include the locator daemon, configuration repository, naming service and so on.

## Configuring Orbix services for Actional integration

To configure Orbix domain services for Actional integration, perform the following steps:

1. Start the **Orbix Configuration** tool using the following command:

```
OrbixInstallDir\asp\6.3\bin\itconfigure
```

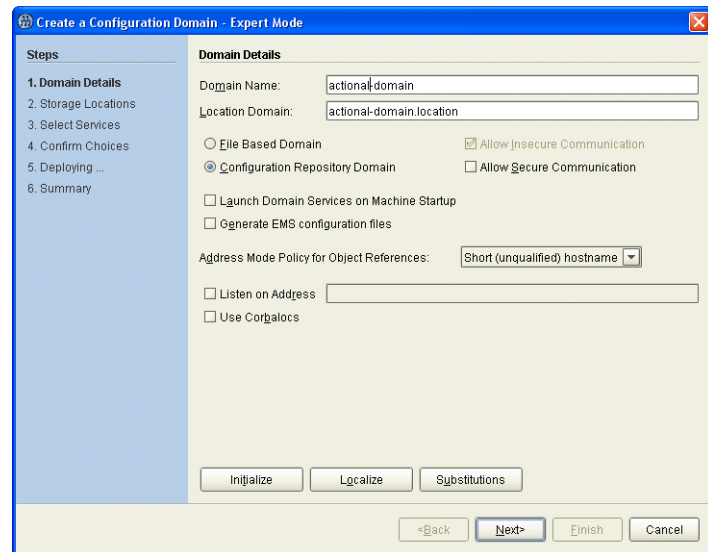
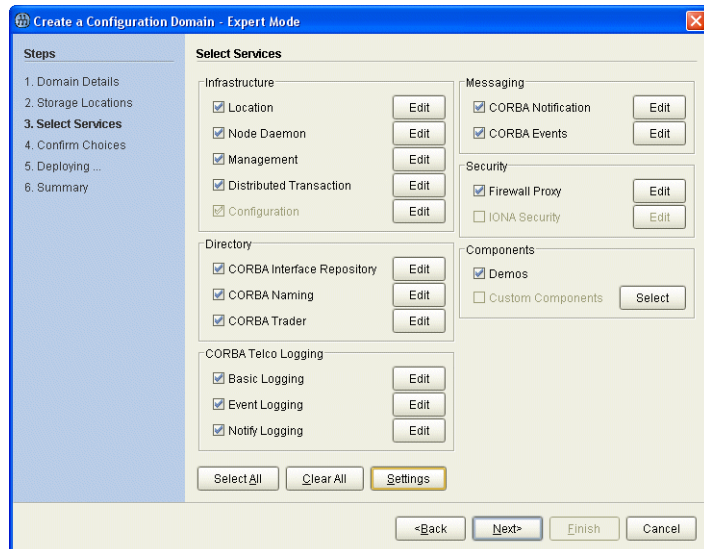


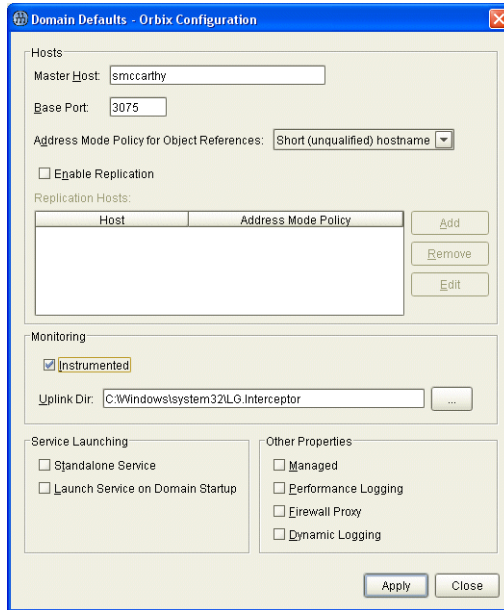
Figure 6: Creating a Domain in Expert Mode

2. Click **Cancel** or press the Esc key to close the **Orbix Configuration Welcome** dialog box.
3. Select **File|New|Expert** to create a domain in **Expert Mode** (shown in [Figure 6](#)).
4. Specify the **Domain Details** (for example, whether it is configuration file-based or configuration repository-based).
5. Click **Next** to specify any custom storage locations.
6. Click **Next** to specify the required Orbix domain services.
7. Select the services you require and click the **Settings** button at the bottom of the screen (shown in [Figure 7](#))



**Figure 7:** *Selecting Services*

8. In the **Domain Defaults** screen, in the **Monitoring** panel, select the **Instrumented** check box (shown in [Figure 8](#)). This will add the required Orbix configuration settings to the Orbix services that you selected.



**Figure 8:** *Specifying Actional Monitoring*

9. If your Actional `Uplink.cfg` configuration file is not located in its default path, specify its directory path in the **Uplink Dir** text box. The path specified must match that specified for your Actional agent. The default values are:

**UNIX**            `/var/opt/actional/LG.Interceptor`

**Windows**        `%systemroot%\system32\LG.Interceptor`

10. Click **Apply**.
11. Click **Close**.
12. Click **Next** to view your selections.
13. Click **Next** to deploy your domain.
14. Click **Finish**.

---

**Using the command line**

You can also use the `enable_actional.tcl` script to automatically add the configuration necessary for Actional integration to the configuration scope of any Orbix service. For more details, see [“Running the `enable\_actional Script`” on page 47](#).

---

**Further information**

For more detailed information on using the **Orbix Configuration** tool, see the *Orbix Deployment Guide*.

---

# Configuring Orbix Java Applications

---

## Overview

This section explains how to configure Orbix Java applications for integration with Actional. It shows some examples from the Orbix Actional integration demo:

```
OrbixInstallDir/asp/6.3/demos/corba/orb/actional_demo
```

---

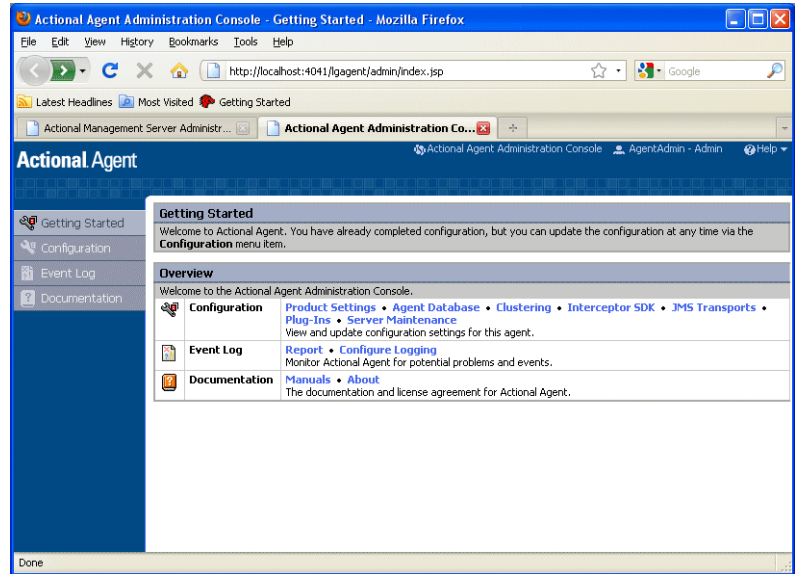
## Update your Actional SDK

You must first update your Actional SDK JAR file as follows:

1. In the **Actional Agent Administration Console**, select **Getting Started | Interceptor SDK** (see [Figure 9](#)), and download the Windows (.zip) or UNIX (.tar) file. This includes the `actional-sdk.jar`, documentation, and samples.
2. Replace the existing `actional-sdk.jar` in the following location with the version that you downloaded:

```
OrbixInstallDir/lib/platform/orbmon/1.3
```





**Figure 9:** *Actional Agent Administration Console*

### Configuring the Orbix monitoring plug-in

You can configure the monitoring plug-in by editing the settings in your application configuration scope in your Orbix configuration file. This includes the following steps:

- Specify the monitoring plug-in
- Add monitoring handlers to the interceptor chain
- Specify the monitoring log filter

**Note:** Alternatively, you can use the `enable_actional.tcl` script to add all the configuration necessary for Actional integration to an Orbix configuration scope (see [“Running the enable\\_actional Script” on page 47](#)).

### Specifying the plug-in name

To set the monitoring plug-in name, add the following settings:

```
# Specify the monitoring class name.
plugins:orbmon:ClassName = "com.iona.corba.plugin.monitoring.MRIPlugIn";

# Load the monitoring plug-in:
orb_plugins = ["local_log_stream", "orbmon", "iiop_profile", "giop", "iiop"];
```

### Adding handlers to the interceptor chain

You must also specify monitoring handlers to the Orbix interceptor binding lists, on both the client side and server side. For example:

```
# Add the client-side handlers to the interceptors chain.
binding:client_binding_list = ["POA_Coloc", "ORBMON+GIOP+IIOP", "GIOP+IIOP"];

# Add the server-side handlers to the interceptors chain.
binding:server_binding_list = ["ORBMON", ""];
```

For more details on configuring Orbix binding lists and interceptors, see the *Orbix Configuration Reference*.

### Specifying the monitoring filter

You can specify the monitoring log filter as follows:

```
event_log:filters = ["IT_MONITORING=*"];
```

For more details, see [“Troubleshooting Orbix” on page 49](#).

**Note:** When you run the **Orbix Configuration** GUI tool (`itconfigure` command), all the configuration necessary for the `actional_demo` is added to your configuration file by default. If you select the **Expert** option, you must select the **Demos** component.

## Running client and server applications

No changes are necessary when running your Orbix Java client and server applications if the Actional `Uplink.cfg` configuration file is located in its default path:

<b>UNIX</b>	<code>/var/opt/actional/LG.Interceptor</code>
<b>Windows</b>	<code>%systemroot%\system32\LG.Interceptor</code>

The `Uplink.cfg` file is responsible for communication between the Actional interceptors and the analyzer in the Actional agent.

If the `Uplink.cfg` is not located in its default path, the `-Dcom.actional.lg.interceptor.config` system property must be added to the Java commands for both the client and the server. For example:

```
java -Dcom.actional.lg.interceptor.config=Path ...
```

### Specifying endorsed directories

If you are using JDK 1.4.x, you must also specify `-Djava.endorsed.dirs` system property on the Java command line as follows:

**Windows** `-Djava.endorsed.dirs="IT_PRODUCT_DIR\lib\art\omg\5"`

**UNIX** `-Djava.endorsed.dirs=IT_PRODUCT_DIR/lib/art/omg/5`

### Sample Orbix configuration

The following sample configuration shows the settings required for Java integration with Actional in an example application configuration scope:

```
my_app
{
  plugins:orbmon:ClassName = "com.iona.corba.plugin.monitoring.MRIPlugIn";
  orb_plugins = ["local_log_stream", "orbmon", "iiop_profile", "giop", "iiop"];

  binding:client_binding_list = ["POA_Coloc", "ORBMON+GIOP+IIOP", "GIOP+IIOP"];
  binding:server_binding_list = ["ORBMON", ""];

  event_log:filters = ["IT_MONITORING=*"];
};
```

### Further information

For more information on Orbix configuration, see the following:

- *Orbix Administrator's Guide*
- *Orbix Configuration Reference*
- *Orbix Deployment Guide*

---

# Configuring Orbix C++ applications

---

## Overview

This section explains how to configure Orbix C++ application for integration with Actional. It shows some examples from the Orbix Actional integration demo:

```
OrbixInstallDir/asp/6.3/demos/corba/orb/actional_demo
```

---

## Setting your environment

No changes are necessary if the Actional `Uplink.cfg` configuration file is located in its default path:

```
UNIX          /var/opt/actional/LG.Interceptor
Windows      %systemroot%\system32\LG.Interceptor
```

The `Uplink.cfg` file is responsible for communication between the Actional interceptors and the analyzer in the Actional agent.

If the `Uplink.cfg` is not located in its default path, you must specify the path to this file as follows:

```
UNIX          export LG_INTERCEPTORCONFIG=PathToFile
Windows      set  LG_INTERCEPTORCONFIG=PathToFile
```

---

## Configuring the Orbix monitoring plug-in

You can configure the monitoring plug-in by editing the settings in your application configuration scope in your Orbix configuration file. This includes the following steps:

- Specify the monitoring plug-in
- Add the monitoring handlers to the interceptor chain
- Specify the monitoring log filter

**Note:** Alternatively, you can use the `enable_actional.tcl` script to add all the configuration necessary for Actional integration to an Orbix configuration scope (see [“Running the enable\\_actional Script” on page 47](#)).

### Specifying the plug-in name

To set the monitoring plug-in name, add the following settings:

```
# Specify the monitoring library.
plugins:orbmon:shlib_name = "it_orb_monitoring";

# Load the monitoring plug-in.
orb_plugins = ["local_log_stream", "orbmon", "iiop_profile", "giop", "iiop"];
```

### Adding handlers to the interceptor chain

You must also specify monitoring handlers to the Orbix interceptor binding lists, on both the client side and server side. For example:

```
# Add the client-side handlers to the interceptors chain.
binding:client_binding_list = ["POA_Coloc", "ORBMON+GIOP+IIOP", "GIOP+IIOP"];

# Add the server-side handlers to the interceptors chain.
binding:server_binding_list = ["ORBMON", ""];
```

For more details on configuring Orbix binding lists and interceptors, see [Orbix Configuration Reference](#).

### Specifying the monitoring filter

You can specify the monitoring log filter as follows:

```
event_log:filters = ["IT_MONITORING=*"];
```

For more details, see [“Troubleshooting Orbix” on page 49](#).

**Note:** When you run the **Orbix Configuration** GUI tool (`itconfigure` command), all the configuration necessary for the `actional_demo` is added to your configuration file by default. If you select the **Expert** option, you must select the **Demos** component.

---

**Sample Orbix configuration**

The following sample configuration shows some example settings in a `my_app` configuration scope:

```
my_app {  
  
  plugins:orbmon:shlib_name = "it_orb_monitoring";  
  orb_plugins = ["local_log_stream", "orbmon", "iiop_profile", "giop", "iiop"];  
  
  binding:client_binding_list = ["POA_Coloc", "ORBMON+GIOP+IIOP", "GIOP+IIOP"];  
  binding:server_binding_list = ["ORBMON", ""];  
  
  event_log:filters = ["IT_MONITORING=*"];  
  
};
```

---

**Further information**

For more information on Orbix configuration, see the following:

- *Orbix Administrator's Guide*
- *Orbix Configuration Reference*
- *Orbix Deployment Guide*

---

# Running the enable\_actional Script

---

## Overview

This section explains how to use the `enable_actional.tcl` script to automatically add the configuration for Actional integration to an Orbix configuration scope. This script can be used to instrument an Orbix C++ or Java application, or an Orbix domain service (for example, locator daemon, naming service, and so on).

## Script usage

The `enable_actional.tcl` script is located in the following directory:

```
OrbixInstallDir\asp\6.3\bin\enable_actional.tcl
```

This script has the following syntax:

```
itadmin enable_actional.tcl ScopeToBeInstrumented
```

You must supply the Orbix configuration scope to be instrumented. This script does not apply to nested configuration scopes.

## Script output

When you run the `enable_actional.tcl` script, it adds the monitoring plug-in (`orbmon`) to the following configuration variables in the specified scope:

- `orb_plugins`
- `binding:server_binding_list`
- `binding:client_binding_list`

It also adds the necessary C++ and Java libraries to the global scope, if not present:

- `plugins:orbmon:shlib_name`
- `plugins:orbmon:ClassName`

## Examples

The following are some example commands

- `itadmin enable_actional.tcl my_c++_app`
- `itadmin enable_actional.tcl my_java_app`
- `itadmin enable_actional.tcl iona_services.locator.MyHost`
- `itadmin enable_actional.tcl iona_services.node_daemon.MyHost`

The following is an example of the configuration settings that are added when the script is run:

```
...

plugins:orbmon:shlib_name = "it_orb_monitoring";
plugins:orbmon:ClassName = "com.iona.corba.plugin.monitoring.MRIPlugIn";

...

my_app {

    orb_plugins = ["orbmon", "local_log_stream", "iiop_profile", "giop", "iiop"];

    binding:server_binding_list = ["ORBMON", "OTS", ""];
    binding:client_binding_list = ["ORBMON+GIOP+IIOP", "POA_Coloc", "GIOP+IIOP"];
} ;
```



---

# Troubleshooting Orbix

---

## Overview

This section provides some tips to help troubleshoot your Orbix integration with Actional.

---

## Ensure the monitoring plug-in is loaded

To verify that the Orbix monitoring plug-in is loaded and participating in the Orbix interceptor chain, you can enable logging by adding `IT_MONITORING` filter to the event log. For example:

```
event_log:filters = ["IT_MONITORING=*"];
```

When logging has been enabled for the monitoring plug-in, logging statements for `IT_MONITORING` should appear in your log files or on screen. This verifies that the monitoring plug-in is correctly loaded, and that and calls are going through the Orbix interceptors.

### Java example

The following are some example logging statements for Orbix Java client and server applications:

```
13:30:43 11/05/2009 [_it_orb_id_1@zajonzd690/10.2.4.13]
(IT_MONITORING:203) I - Client Interaction begin
13:30:43 11/05/2009 [_it_orb_id_1@zajonzd690/10.2.4.13]
(IT_MONITORING:203) I - Server Interaction begin
```

In addition, when the `actional-sdk.jar` is used, it prints the following logging statement to `stderr`:

```
2009-11-05 13:30:43.070+0000 Actional logging to System.err
```

### C++ example

The following are some example logging statements for Orbix C++ client and server applications:

```
Thu, 05 Nov 2009 13:38:32.0000000 [ZAJONZD690:4584]
(IT_MONITORING:4) I - ServerInteraction url:
Simple/SimpleObject opname: call_me self: 10.2.4.13 peer:
10.2.4.13
Thu, 05 Nov 2009 13:38:32.0000000 [ZAJONZD690:5688]
(IT_MONITORING:4) I - ClientInteraction url:
Simple/SimpleObject opname: call_me peer: 10.2.4.13
```

# Configuring Actional for Orbix Integration

*This chapter gives some basic guidelines on setting up Actional to run the Orbix Actional integration demo.*

---

**In this chapter**

This chapter includes the following sections:

<a href="#">Prerequisites</a>	<a href="#">page 52</a>
<a href="#">Configuring Actional</a>	<a href="#">page 53</a>
<a href="#">Troubleshooting Actional</a>	<a href="#">page 57</a>

# Prerequisites

---

## Overview

This section describes prerequisites for integration between Actional SOA management products and Orbix.

---

## Actional products

The following Actional products should be installed:

- Actional Management Server 8.0 (Actional server)
- Actional Flex Point 8.0 (Actional agent/intermediary)

Alternatively, the following Actional products can be installed separately:

- Actional Point of Operational Visibility 8.0 (Actional agent)
  - Actional Client Security Enforcement 8.0 (Actional intermediary)
- 

## Actional agents

You must ensure that Actional agents have been set up on each Orbix host node that you wish to manage. The provisioning of Actional agents is performed using the Actional server. For some basic details, see [“Configuring Actional for Orbix Integration” on page 51](#).

For full details on how to set up Actional agents on managed nodes, see the Actional product documentation.

---

## Further information

For information on installing Actional products, and the full range of platform and database versions supported by Actional, see the Actional product documentation.

This Orbix integration with Actional supports the full range of operating systems and compilers supported by Orbix. For more details, see the *Orbix Installation Guide*.

---

# Configuring Actional

---

## Overview

This section provides some basic configuration guidelines on Actional agent and server configuration. For full details, see the Actional product documentation.

This basic configuration helps to set up the Orbix `actional_demo`. For information on how to run this demo, see the `README` text files in the following directory:

```
OrbixInstallDir/asp/6.3/demos/corba/orb/actional_demo
```

---

## Actional agent configuration

No specific Actional agent configuration settings are required for integration with Orbix. For example, for the purposes of the Orbix–Actional integration demos, the Actional agent can be started with the default configuration settings.

---

## Actional server configuration

The following sample configuration steps describe how to set up the Actional server to run a simple Orbix–Actional demo:

1. Install the Actional server with typical installation options, and select the Apache Derby database.

**Note:** The Apache Derby database is provided for demo purposes only, and is not recommended for a production environment.

2. Specify the following URL in your browser:

```
http://localhost:4040/lserver
```

3. If this is a new installation click **Start**, and follow the new Actional server setup steps.

Otherwise, if the Actional server is already installed, perform the following steps:

- i. In the Actional console Web interface, select the **Configure** radio button in the top left of the screen.
- ii. Select the **Platform** tab. This displays the general configuration settings, as shown in [Figure 10](#).

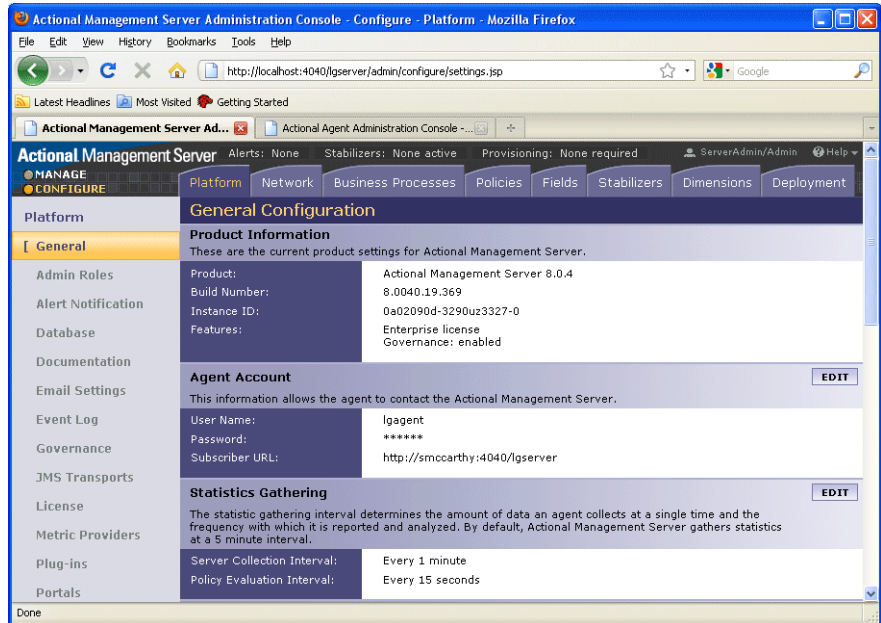


Figure 10: Actional Server Configuration Settings

### Creating a managed node

To create a managed node for a simple Orbix demo, perform the following steps:

1. In the Actional **Configure** view menu bar, open the **Network** tab. This displays the **Network Nodes**.
2. Select **Add**. This displays **Node Creation / Managing Agents**.
3. Click **Managed Node**.

## Configuring a new node

To configure a managed node for the demo, perform the following steps in the wizard:

### Step 1: New Node - Identification

1. Specify the **Name** as `agent1`.
2. Specify the **Display icon** as `Auto Discover`.
3. Click **Next**.

### Step 2: New Node - Management

1. Specify the **Transport** as `HTTP/S`.
2. Supply your Actional agent user name and password.
3. Ensure that **Override Agent Database** is checked.
4. Click **Next**.

### Step 3: New Node - Agents

1. Specify the following URL:  
`http://HostName:4041/lagent`  
You can specify a host name or an IP address in this URL.
2. Click **Add**. The agent URL is added.
3. Click **Next**.

### Step 4: New Node - Endpoints

1. For **Endpoints**, add the hostname, fully qualified hostname, or IP address.
2. Click **Next**.

### Step 5: New Node - Filters

1. Do not specify any filters for the demo.
2. Click **Next**.

### Step 6: New Node - Trust Zone

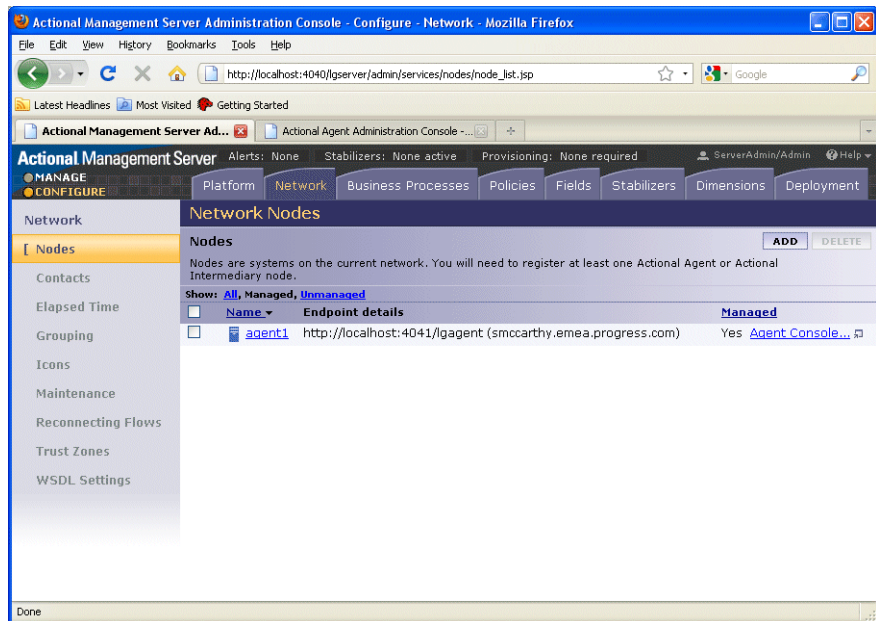
1. Do not specify a trust zone for the demo.
2. Click **Finish**

The newly created managed node now needs to be provisioned.

## Provisioning a new node

To provision the new node to bring it under management, perform the following steps:

1. Select the **Configure** radio button at the top left of the screen.
2. Select the **Deployment** tab from the **Configure** menu bar.
3. The **Provisioning** page is displayed, and `agent1` is listed as not provisioned.
4. Select the `agent1` check box.
5. Click **Provision**. This displays a message when complete:  
Successfully provisioned.
6. Click the **Manage** radio button at the top left of the screen. You should see `agent1` added to the **Network** view as shown in [Figure 11](#).



**Figure 11:** Actional Server Provisioned Node

## Further information

For more details on setting up and running Actional SOA management tools, see the Actional product documentation.



---

# Troubleshooting Actional

---

## Overview

This section provides some tips to help troubleshoot your Actional integration with Orbix.

---

## Setting default polling

For demonstration purposes, to update the display in your Actional server console more frequently, you can set the default polling to a shorter time span as follows:

1. Select the **Configure** radio button at the top left of the screen.
2. Select the **Platform** tab from the **Configure** menu bar.
3. In **Statistics Gathering** on the right, select **EDIT**.
4. Set the **Server Collection Interval** to 1 minute by using the drop down list.
5. Set the **Policy Evaluation Interval** to 15 seconds.

**Note:** These settings are for demonstration purposes only, and may not be suitable for a production environment.

---

## Ensuring events are reported to the Actional Agent

To ensure that Orbix monitoring events are being reported to your Actional agent, perform the following steps:

1. Ensure your Actional agent is running, and added as a managed node in your Actional server.
2. Verify that the agent generated the `Uplink.cfg` file in the directory specified during installation. If this file was not specified during the installation, it should be in the following default path (which should have write permission):

**UNIX**            `/var/opt/actional/LG.Interceptor`

**Windows**        `%systemroot%\system32\LG.Interceptor`

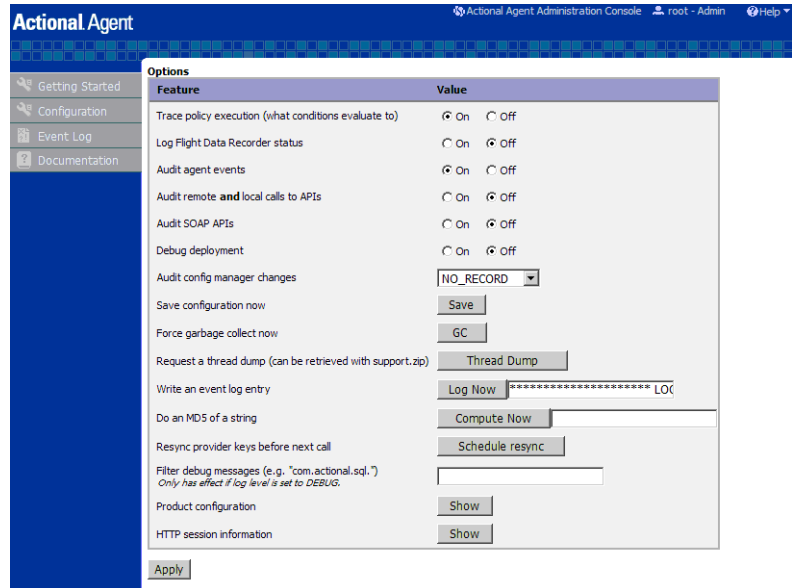
3. Open your Actional agent console and login:

`http://AgentHostName:Port/lgagent/`

- Specify the following URL to display the **Options** page shown in [Figure 12](#):

`http://AgentHostName:Port/lagent/admin/options.js`

- For **Audit agent events**, Click **On**.
- Click **Apply**.

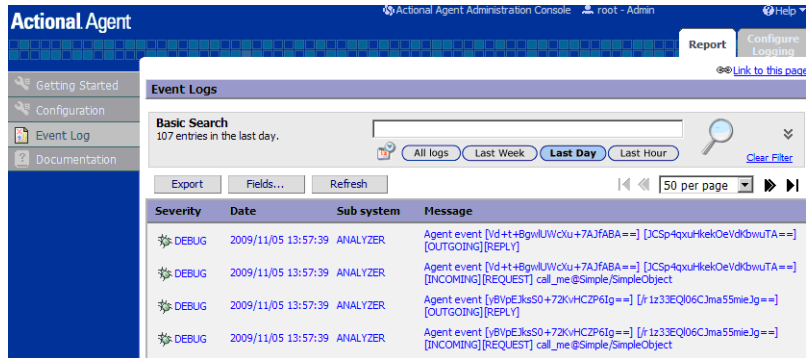


**Figure 12:** Actional Agent Options

**Note:** These settings are not persistent, and are reset when the Actional agent is restarted.

### Viewing agent events

When **Audit agent events** is turned on, all external events coming from the Orbix monitoring plug-in can be reviewed in the Actional agent **Event Logs**, shown in [Figure 13](#).



**Figure 13:** Actional Agent Event Logs

[Figure 13](#) shows INCOMING, OUTGOING, REQUEST, and REPLY events reported from the monitoring plug-in. If these events are not reported, the path for the `uplink.cfg` may be incorrect, and the monitoring plug-in can not find the agent.

### C++ applications

For C++ applications, verify that the `LG_INTERCEPTORCONFIG` environment variable is set correctly, and points to the directory where the agent has written the `uplink.cfg` file.

### Java applications

For Java applications, verify that the `com.actional.lg.interceptor.config` property is passed on to the application correctly, and points to the directory where the agent has written the `uplink.cfg` file. For example:

```
java -Dcom.actional.lg.interceptor.config=%SystemRoot%\system32\LG.Interceptor
-classpath .\java\classes;"%CLASSPATH%" actional_demo.Server -ORName
demos.actional_demo
```

When incoming monitoring events are arriving at the agent, and the agent is configured correctly, you should see the calls displayed in the Actional server console **Network** view, as shown in [Chapter 4](#).

---

**Further information**

For any problems with Actional agent configuration, please refer to the Actional product documentation.

# Managing Orbix Applications in Actional

*This chapter shows examples of managing a simple Orbix application and Orbix domain services in Actional SOA management tools.*

## **In this chapter**

---

This chapter includes the following sections:

<a href="#">Monitoring Orbix Applications</a>	<a href="#">page 62</a>
<a href="#">Monitoring Orbix Domain Services</a>	<a href="#">page 67</a>
<a href="#">Auditing Orbix Applications</a>	<a href="#">page 70</a>

---

# Monitoring Orbix Applications

---

## Overview

When your Orbix applications have been configured for integration with Actional, they can be monitored using the Actional SOA management tools. No code changes are required for monitoring of Orbix applications.

For example, when you run the simple Orbix `actional_demo`, the **Actional Management Server Administration Console** displays the managed node that the demo is running on. Invocations are displayed as arrows flowing to and from managed components.

The Orbix `actional_demo` illustrates the simple use of the ORB monitoring plug-in to report calls made between Orbix clients and servers to Actional. This demo is similar to `demos/corba/orb/simple`, and shows how to configure visibility of your application in Actional. For details on how to run this demo, see the `README` text files in the following directory:

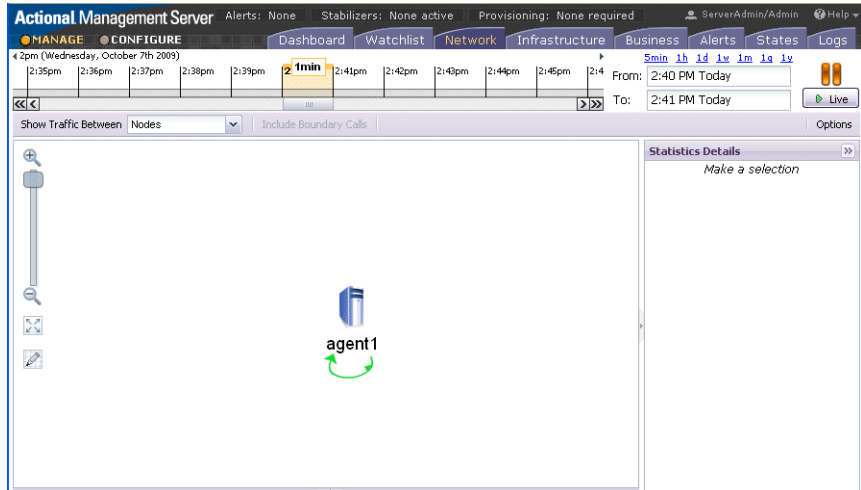
```
OrbixInstallDir/asp/6.3/demos/corba/orb/actional_demo
```

---

## Network view

The Actional network view displays the traffic between various components in your network environment. These include nodes, packages, services and operations.

[Figure 14](#) shows the running Orbix `actional_demo` displayed in the **Network** tab of the **Actional Management Server Administration Console**. In this simple demo, the **Network** tab displays the Actional agent on the Orbix managed node that the demo is running on. This agent reports the monitoring data back to the Actional server. The single invocation is displayed as a green arrow flowing from the node and back to itself. In more complex examples with multiple nodes, the arrows flow between nodes.



**Figure 14:** Actional Server Network View

By default, the **Network** view shows traffic between nodes. There is only one node in this case. You can also select to show traffic between packages in the top left of the screen. [Figure 15](#) shows the traffic between the Orbix client and server packages.

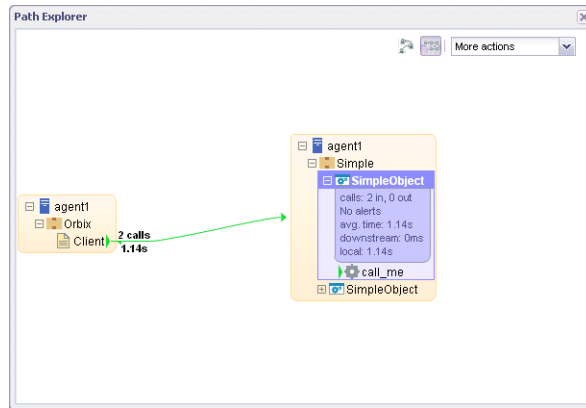


**Figure 15:** Traffic Between Packages

## Path Explorer

Figure 16 shows the Orbix `actional_demo` displayed in the **Path Explorer** view of the **Actional Management Server Administration Console**.

To view this screen, double click on the managed node shown in Figure 14. Alternatively, click the **Display Path Explorer** button at the top right of the **Network** view.



**Figure 16:** *Actional Server Path Explorer*

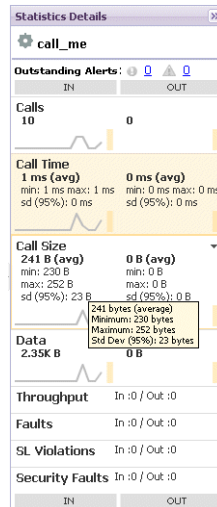
The **Path Explorer** view displays the relationships between different components in more detail. For example, you can view the call chain between services and consumers. Summary statistics are also displayed for the selected component.



## Statistics details

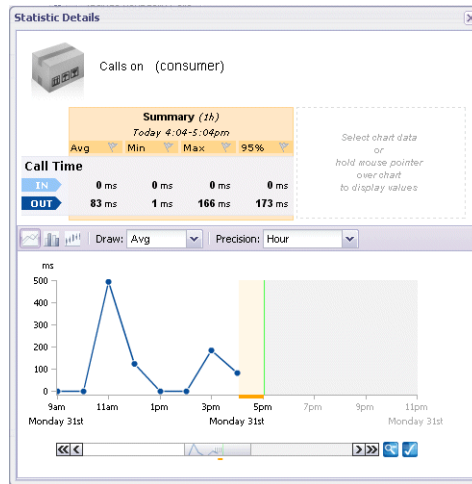
The **Statistics Details** pane on the right displays statistics gathered by the selected component. These include the number of incoming and outgoing calls, call time, call size, and so on. Alerts, faults and violations are also displayed.

For example, [Figure 17](#) shows the **Statistics Details** displayed on the right when the `call_me()` operation is selected in the **Path Explorer**.



**Figure 17:** Actional Server Statistics Details

Double clicking on a particular statistic in this view (for example, **Call Size**) displays a summary chart. For example, [Figure 18](#) shows a **Call Time** summary chart for the consumer.



**Figure 18:** Actional Server Statistics Chart

## Server manifest

The Actional server manifest (`LG_Header`) is a unique ID used by the Actional server to correlate information it receives from agents about interactions between different applications. For example, when you run the client application in the Orbix `actional_demo`, the following `LG_Header` is output on the command line:

```
Interaction=CgIEAUD6LU2sLiQBBwAAAA==;
Locus=4/LcwqvldfxotEoegsSGg==;
Flow=CgIEAUD6LU2sLiQBBgAAAA==;
UpstreamOpID=xPnAfuwlTEV7QGYoGRBgYA==;
CallerAddress=10.2.4.1;
```

For more details, see [“Actional server manifest” on page 31](#).

## Further information

For detailed information on using Actional SOA management tools, see the Actional product documentation.

---

# Monitoring Orbix Domain Services

---

## Overview

Orbix configuration domain services can be integrated with Actional automatically using the **Orbix Configuration** tool. These include services such as the Orbix configuration repository, locator daemon, node daemon, and so on. No manual configuration updates are required. For more details, see [“Configuring an Orbix Domain” on page 36](#)

This section shows examples of monitoring Orbix domain services in Actional SOA management tools.

---

## Starting Orbix services

To start your Orbix configuration domain services, perform the following steps:

1. Set your Orbix domain environment, for example:

```
c:\orbix\etc\bin>actional-cfr-domain_env.bat  
Setting environment for domain actional-cfr-domain
```

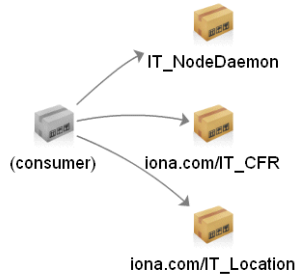
You must have configured your domain to be monitored by Actional (see [“Configuring an Orbix Domain” on page 36](#)).

2. Start your domain services, for example:

```
c:\orbix\etc\bin>start_actional-cfr-domain_services.bat  
Orbix services logging to: C:\orbix\var\actional-cfr-domain\logs  
Starting iona_services.config_rep.Hostname  
Starting iona_services.locator.Hostname  
Starting iona_services.node_daemon.Hostname  
Finished.
```

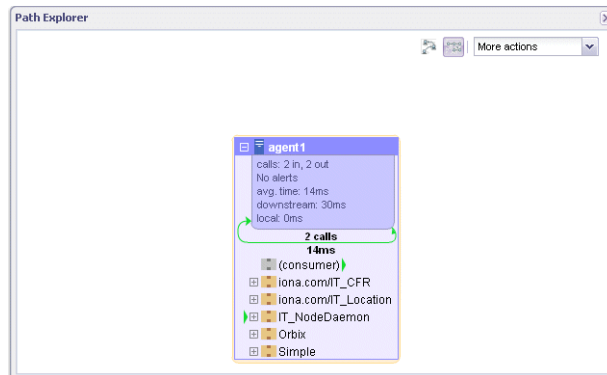
## Monitoring Orbix services

Figure 19 shows the traffic between packages for the Orbix configuration domain services. The services displayed are the node daemon, configuration repository, and locator daemon.



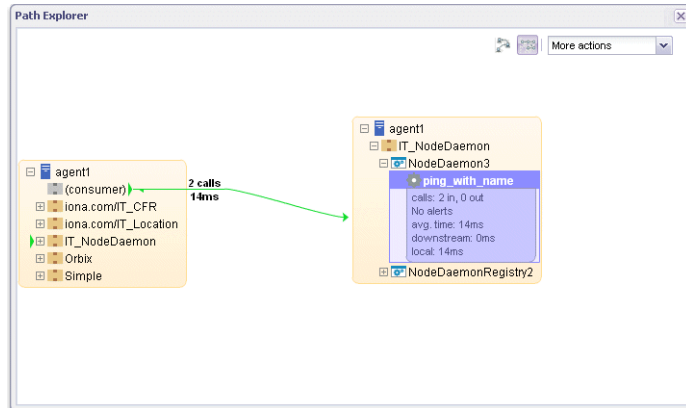
**Figure 19:** Traffic Between Domain Services Packages

Figure 20 shows the running Orbix domain services displayed in the **Path Explorer** view.



**Figure 20:** Domain Services in Path Explorer

Figure 21 shows the call displayed for the node daemon `ping_with_name()` operation:



**Figure 21:** *Node Daemon Operation*

### Further information

For detailed information on using Actional SOA management tools, see the Actional product documentation.

# Auditing Orbix Applications

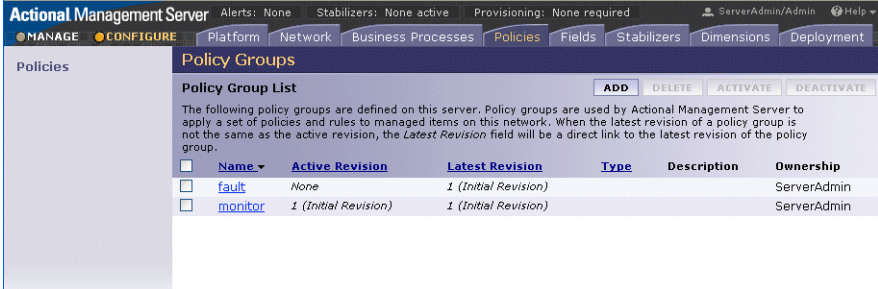
## Overview

This section shows some simple examples of auditing the Orbix `actional_demo` and Orbix domain services.

## Actional policy groups

Policy groups are used by Actional server to apply a set of policies and rules to managed items on your network. Policies and rules can be used to raise alerts on certain failure reasons. For example, when an Orbix operation takes too long to return, or when a specified IDL exception or fault is raised.

Figure 22 shows some example policy groups that have been defined in the **Policies** view.



<input type="checkbox"/>	Name	Active Revision	Latest Revision	Type	Description	Ownership
<input type="checkbox"/>	<a href="#">fault</a>	None	<a href="#">1 (Initial Revision)</a>			ServerAdmin
<input type="checkbox"/>	<a href="#">monitor</a>	<a href="#">1 (Initial Revision)</a>	<a href="#">1 (Initial Revision)</a>			ServerAdmin

Figure 22: Actional Policy Groups

## Viewing audit logs

When you have defined policies for your network, you can use them to audit and monitor alerts on certain failure reasons (for example, when a specified IDL exception or fault is raised).

Figure 23 shows some example audit logs for the Orbix `actional_demo` in the **Logs** view.

Date	Host Name	Service	Operation	Request ID	Call Status	Failure Reason	Response Time (ms)	Authenticated Security ID
2009-10-07 14:40:21,215	smccarthy.emea.progress.com	SimpleObject	call_me	n/a	SUCCEEDED	n/a	186	n/a
2009-10-07 14:40:21,153	smccarthy.emea.progress.com	Client	n/a	n/a	SUCCEEDED	n/a	230	n/a
2009-10-07 14:40:19,012	smccarthy.emea.progress.com	SimpleObject	call_me	n/a	SUCCEEDED	n/a	2094	n/a
2009-10-07 14:40:16,199	smccarthy.emea.progress.com	Client	n/a	n/a	SUCCEEDED	n/a	4938	n/a

Figure 23: Actional Demo Audit Logs

Figure 24 shows an example audit log record displayed on clicking on an entry for the Orbix `actional_demo` in Figure 23.

The screenshot displays the 'Audit Logs' section of the Actional Management Server. The left sidebar shows a navigation menu with 'Audit Logs' selected. The main content area shows 'Audit Log Record 1 of 8' with the following details:

Field	Value
Interaction ID:	CgIEAcfbhE6sLiQBEGAAAA==
Date:	2009-10-07 14:40:21.215
Host Name:	smccarthy.emea.progress.com
Group:	Simple
Group Revision:	
Service:	SimpleObject
Operation:	call_me
URL Path:	/
Request ID:	
Request Size (bytes):	236
Request Data:	
Request Attachments:	none
Request Message Fields:	none
Call Status:	SUCCEEDED
Failure Reason:	
Response Time (ms):	186
Reply Size (bytes):	24

**Figure 24:** Actional Demo Audit Log Record

The **Interaction ID** displayed at the top of the screen is used by the Actional server to correlate information it receives, from multiple agents, about interactions between different services. For more details, see [“Actional server manifest”](#) on page 31.



Figure 25 shows some example audit logs for Orbix configuration domain services in the **Logs** view. The Orbix service displayed in this example is the Orbix node daemon.

Audit Logs								
Audit Log Report								
Date	Host Name	Service	Operation	Request ID	Call Status	Failure Reason	Response Time (ms)	Authenticated Security ID
2009-10-13 12:18:49.837	smccarthy.emea.progress.com	NodeDaemon3	ping_with_name	n/a	SUCCEEDED	n/a	14	n/a
2009-10-13 12:18:19.806	smccarthy.emea.progress.com	NodeDaemon3	ping_with_name	n/a	SUCCEEDED	n/a	14	n/a
2009-10-13 12:17:49.728	smccarthy.emea.progress.com	NodeDaemon3	ping_with_name	n/a	SUCCEEDED	n/a	15	n/a
2009-10-13 12:17:19.696	smccarthy.emea.progress.com	NodeDaemon3	ping_with_name	n/a	SUCCEEDED	n/a	15	n/a
2009-10-13 12:16:49.665	smccarthy.emea.progress.com	NodeDaemon3	ping_with_name	n/a	SUCCEEDED	n/a	14	n/a
2009-10-13 12:16:19.634	smccarthy.emea.progress.com	NodeDaemon3	ping_with_name	n/a	SUCCEEDED	n/a	14	n/a
2009-10-13 12:15:49.603	smccarthy.emea.progress.com	NodeDaemon3	ping_with_name	n/a	SUCCEEDED	n/a	14	n/a
2009-10-13 12:15:19.571	smccarthy.emea.progress.com	NodeDaemon3	ping_with_name	n/a	SUCCEEDED	n/a	15	n/a

Figure 25: Domain Services Audit Logs

Figure 26 shows an example audit log record displayed on clicking an entry for the Orbix node daemon in Figure 25.

Audit Logs	
Audit Log Record 401 of 3172	
Interaction ID:	TFFJ3I5LkSLK45br7EXUg==
Date:	2009-10-13 12:18:49.837
Host Name:	smccarthy.emea.progress.com
Group:	IT_NodeDaemon
Group Revision:	
Service:	NodeDaemon3
Operation:	ping_with_name
URL Path:	IT_NodeDaemon/NodeDaemon3
Request ID:	
Request Size (bytes):	232
Request Data:	
Request Attachments:	none
Request Message Fields:	none
Call Status:	SUCCEEDED
Failure Reason:	
Response Time (ms):	14
Reply Size (bytes):	12
Reply Data:	

**Figure 26:** *Node Daemon Log Record*

### Further information

For detailed information on using Actional SOA management tools, see the Actional product documentation.

# Glossary

---

## A

### **Actional agent**

Run on each host that you wish to manage, and used to provide instrumentation data back to the Actional server. It includes two main components: an analyzer, and one or more interceptors. The analyser gathers and evaluates data such as records, statistics, and alerts. The interceptors collect data about service traffic from an application server, and apply policies to that traffic.

### **Actional server**

A central management server that manages nodes containing an Actional agent. The Actional server correlates the data it receives from each of its agents, and distributes policies to those agents. It enables an administrator to analyze service network data and create system-wide policies.

### **Actional server manifest**

A token sent by the Actional server sends with a request document to provide information about the request's origin and the business flow that the request belongs to. The Actional server manifest (`LG_Header`) is used by the Actional server to correlate information it receives, from multiple agents, about interactions between different services. For this reason, the server manifest is sometimes referred to as a correlation ID.

### **administration**

All aspects of installing, configuring, deploying, monitoring, and managing a system.

### **ART**

Adaptive Runtime Technology. A modular, distributed object architecture that supports dynamic deployment and configuration of services and application code. ART provides the foundation for Orbix and Artix software products.

## C

### **CFR**

See [configuration repository](#).

**client**

An application (process) that typically runs on a desktop and requests services from other applications that often run on different machines (known as server processes). In CORBA, a client is a program that requests services from CORBA objects.

**configuration**

A specific arrangement of system elements and settings.

**configuration domain**

Contains all the configuration information that Orbix ORBs, services and applications use. Defines a set of common configuration settings that specify available services and control ORB behavior. This information consists of configuration variables and their values. Configuration domain data can be implemented and maintained in a centralized Orbix configuration repository or as a set of files distributed among domain hosts. Configuration domains enable you to organize ORBs into manageable groups, bringing scalability and ease-of-use to large environments. See also [configuration file](#) and [configuration repository](#).

**configuration file**

A file that contains configuration information for Orbix components within a specific configuration domain. See also [configuration domain](#).

**configuration repository**

A centralized store of configuration information for all Orbix components within a specific configuration domain. See also [configuration domain](#).

**configuration scope**

Orbix configuration is divided into scopes. These are typically organized into a root scope and a hierarchy of nested scopes, the fully-qualified names of which map directly to ORB names. By organizing configuration properties into various scopes, different settings can be provided for individual ORBs, or common settings for groups of ORB. Orbix services, such as the naming service, have their own configuration scopes.

## **CORBA**

Common Object Request Broker Architecture. An open standard that enables objects to communicate with one another regardless of what programming language they are written in, or what operating system they run on. The CORBA specification is produced and maintained by the OMG. See also [OMG](#).

### **CORBA naming service**

An implementation of the OMG Naming Service Specification. Describes how applications can map object references to names. Servers can register object references by name with a naming service repository, and can advertise those names to clients. Clients, in turn, can resolve the desired objects in the naming service by supplying the appropriate name. The Orbix naming service is an example.

### **CORBA objects**

Self-contained software entities that consist of both data and the procedures to manipulate that data. Can be implemented in any programming language that CORBA supports, such as C++ and Java.

### **CORBA transaction service**

An implementation of the OMG Transaction Service Specification. Provides interfaces to manage the demarcation of transactions and the propagation of transaction contexts. Orbix OTS is such as service.

### **correlation ID**

See [Actional server manifest](#).

---

## **D**

### **deployment**

The process of distributing a configuration or system element into an environment.

---

## **G**

### **GIOP**

General Inter-ORB Protocol. The general CORBA standard messaging protocol, defined by the OMG, for communications between ORBs and distributed applications. The implementation of GIOP for TCP/IP is IIOP. See [IIOP](#).

**H**

---

**HTTP**

HyperText Transfer Protocol. The underlying protocol used by the World Wide Web. It defines how files (text, graphic images, video, and other multimedia files) are formatted and transmitted. Also defines what actions Web servers and browsers should take in response to various commands. HTTP runs on top of TCP/IP.

---

**I****IDL**

Interface Definition Language. The CORBA standard declarative language that allows a programmer to define interfaces to CORBA objects. An IDL file defines the public API that CORBA objects expose in a server application. Clients use these interfaces to access server objects across a network. IDL interfaces are independent of operating systems and programming languages.

**IFR**

See [interface repository](#).

**IIOP**

Internet Inter-ORB Protocol. The CORBA standard messaging protocol, defined by the OMG, for communications between ORBs and distributed applications. IIOP is defined as a protocol layer above the transport layer, TCP/IP.

**implementation repository**

A database of available servers, it dynamically maps persistent objects to their server's actual address. Keeps track of the servers available in a system and the hosts they run on. Also provides a central forwarding point for client requests. See also [location domain](#) and [locator daemon](#).

**IMR**

See [implementation repository](#).

**instrumentation**

Code instructions that monitor specific components in a system (for example, instructions that output logging information on screen). When an application contains instrumentation code, it can be managed using a management tool such as Actional.

**installation**

The placement of software on a computer. Installation does not include configuration unless a default configuration is supplied.

**Interface Definition Language**

See [IDL](#).

**interceptor**

An Actional interceptor collects data about service traffic from an application server, and applies policies to that traffic. It sits in the flow between the application logic and the consumers and providers of other services. It intercepts all inbound and outbound calls, and feeds information about those calls to an Actional agent.

An Orbix interceptor is an object that ORB services and transports implement to process operation invocations. Orbix interceptors are arranged in a chain, with each interceptor caching a reference to the next interceptor in the chain.

**interface repository**

Provides centralized persistent storage of IDL interfaces. An Orbix client can query this repository at runtime to determine information about an object's interface, and then use the Dynamic Invocation Interface (DII) to make calls to the object. Enables Orbix clients to call operations on IDL interfaces that are unknown at compile time.

**invocation**

A request issued on an already active software component.

**IOR**

Interoperable Object Reference. See [object reference](#).

## L

---

**LG\_Header**

See [Actional server manifest](#).

**location domain**

A collection of servers under the control of a single locator daemon. Can span any number of hosts across a network, and can be dynamically extended with new hosts. See also [locator daemon](#) and [node daemon](#).

**locator daemon**

A server host facility that manages an implementation repository and acts as a control center for a location domain. Orbix clients use the locator daemon, often in conjunction with a naming service, to locate the objects they seek. Together with the implementation repository, it also stores server process data for activating servers and objects. When a client invokes on an object, the client ORB sends this invocation to the locator daemon, and the locator daemon searches the implementation repository for the address of the server object. In addition, enables servers to be moved from one host to another without disrupting client request processing. Redirects requests to the new location and transparently reconnects clients to the new server instance. See also [location domain](#), [node daemon](#), and [implementation repository](#).

## N

---

**naming service**

See [CORBA naming service](#).

**node**

An Actional node is defined as a system on the current network. A node with an Actional agent installed is referred to as an instrumented node or a managed node.

**node daemon**

An Orbix node daemon starts, monitors, and manages Orbix servers on a host machine. Every machine that runs an Orbix server must run a node daemon.



---

**O****object reference**

Uniquely identifies a local or remote object instance. Can be stored in a CORBA naming service, in a file or in a URL. The contact details that a client application uses to communicate with a CORBA object. Also known as interoperable object reference (IOR) or proxy.

**OMG**

Object Management Group. An open membership, not-for-profit consortium that produces and maintains computer industry specifications for interoperable enterprise applications, including CORBA. See [www.omg.com](http://www.omg.com).

**ORB**

Object Request Broker. Manages the interaction between clients and servers, using the Internet Inter-ORB Protocol (IIOP). Enables clients to make requests and receive replies from servers in a distributed computer environment. Key component in CORBA.

**OTS**

See [CORBA transaction service](#).

---

**P****POA**

Portable Object Adapter. Maps object references to their concrete implementations in a server. Creates and manages object references to all objects used by an application, manages object state, and provides the infrastructure to support persistent objects and the portability of object implementations between different ORB products. Can be transient or persistent.

**protocol**

Format for the layout of messages sent over a network.

---

**S****server**

An application that provides services to clients. CORBA servers act as containers for CORBA objects, allowing clients to access those objects using IDL interfaces.

**service context**

A GIOP service context is a general mechanism for including out-of-band data in a GIOP request or reply message. Service contexts in GIOP are analogous to headers in other protocols such as HTTP.

**SSL**

Secure Sockets Layer protocol. Provides transport layer security—authenticity, integrity, and confidentiality—for authenticated and encrypted communications between clients and servers. Runs above TCP/IP and below application protocols such as HTTP and IIOP.

**T**

---

**TCP/IP**

Transmission Control Protocol/Internet Protocol. The basic suite of protocols used to connect hosts to the Internet, intranets, and extranets.

**TLS**

Transport Layer Security. An IETF open standard that is based on, and is the successor to, SSL. Provides transport-layer security for secure communications. See also [SSL](#).

# Index

## A

- Actional agent 26, 30, 52
- Actional Agent Interceptor SDK 26
- Actional Client Security Enforcement 52
- Actional Flex Point 52
- Actional interceptor 30
- Actional intermediary 26
- Actional Management Server 52
- Actional Management Server Administration
  - Console 25, 27, 62
- Actional Point of Operational Visibility 52
- actional-sdk.jar 40
- Actional server 25
- Actional server, configuration 53
- Actional server manifest 30, 31
- Adobe Flash 24
- alerts 24
- analyser 26
- Apache Derby 26, 53
- Apache Tomcat 25
- Audit agent events 58
- audit logs 71

## B

- binding 32

## C

- C++ 24
- com.actional.lg.interceptor.config 59
- correlation ID 31

## D

- database 26, 53
- DB2 26
- Dcom.actional.lg.interceptor.config 43
- default polling 57
- dependency mapping 24
- developers 19
- Djava.endorsed.dirs 43
- Domain Defaults 37

## E

- enable\_actional.tcl 39, 41, 44, 47
- endorsed directories 43
- Event Logs 59
- Expert Mode 37, 42, 45

## F

- Flash 24

## G

- GIOP service context 32
- group 28

## H

- host 28

## I

- INCOMING 59
- instrumented node 25
- Interaction ID 72
- interceptor chain 41, 44
- interceptors 26, 30, 42, 79
- Interceptor SDK 40
- interface 28
- itconfigure 45
- IT\_MONITORING 42, 45

## J

- Java handlers 41, 44
- JBoss 26
- Jetty 24
- JSP 24

## L

- LG\_Header 30, 31
- LG\_INTERCEPTORCONFIG 44, 59
- log filter 42, 45

## M

- managed node 25, 54

## INDEX

managed node, configuration 55  
module 28  
monitoring plug-in 42, 45  
MSDE 26

## N

Network tab 60, 62  
Network view 56  
NGSO mapping 28  
node 28

## O

OpenEdge 26  
operation 28  
Oracle 26  
Orbix Configuration GUI 45  
ORBMON 42, 45  
orbmon 42, 45  
OUTGOING 59  
Override Agent Database 55

## P

Path Explorer 64, 68  
Policy Evaluation Interval 57  
policy groups 70  
PostgreSQL 26  
provisioning 56

## R

REPLY 59  
REQUEST 59  
request-level interceptor 32  
response time 24

## S

Server Collection Interval 57  
server manifest 31  
service 28  
service context 32  
SOAP over HTTP 52  
SQL Server 26  
Statistics Details 65  
Statistics Gathering 57  
system administrators 19  
system architects 19

## T

Tomcat 25

## U

Uplink.cfg 38, 42, 44, 57  
Uplink Dir 38

## W

WebLogic 26  
WebSphere 26