

ABAQUS 2016

INSTALLATION AND LICENSING GUIDE



3DEXPERIENCE®

Abaqus

Installation and Licensing

Guide

Trademarks and Legal Notices

Abaqus, the 3DS logo, and SIMULIA are commercial trademarks or registered trademarks of Dassault Systèmes or its subsidiaries in the United States and/or other countries. Use of any Dassault Systèmes or its subsidiaries trademarks is subject to their express written approval. Other company, product, and service names may be trademarks or service marks of their respective owners.

Abaqus and this documentation may be used or reproduced only in accordance with the terms of the software license agreement signed by the customer, or, absent such an agreement, the then current software license agreement to which the documentation relates.

This documentation and the software described in this documentation are subject to change without prior notice.

Dassault Systèmes and its subsidiaries shall not be responsible for the consequences of any errors or omissions that may appear in this documentation.

© Dassault Systèmes, 2015

Third-Party Copyright Notices

Certain portions of Abaqus contain elements subject to copyright owned by the entities listed below.

© DISTENE
© Elysium Inc.
© Intel Corp.
Abaqus solver technology utilizes routines in Numerical Recipes: The Art of Scientific Computing, published by Cambridge University Press, and are used by permission.
First Edition FORTRAN : "Copyright(C)1986 Numerical Recipes Software" First Edition C : "Copyright(c)1987,1988 Numerical Recipes Software" Second Ed.FORTRAN : "Copyright(c)1986,1992 Numerical Recipes Software" Second Ed.C: "Copyright(c)1987-1992 Numerical Recipes Software"

Abaqus Licensed Programs may include open source software components. Source code for these components is available if required by the license.

The open source software components are grouped under the applicable licensing terms. Links to common license terms are included below.

IP Asset Name	IP Asset Version	Copyright notice
FreeType	2.1.10	Copyright© 1996-2002, 2006 by David Turner, Robert Wilhelm, and Werner Lemberg. Copyright © 2006 The Free Type Project (www.freetype.org). All rights reserved.
Libpng	1.0.3	Copyright © 2004, 2006-2014 Glenn Randers-Pehrson depending on the asset version.
Metis	4.0.1	Copyright 1998, Regents of the University of Minnesota
Natural Order Sort	Version 2004-10-10 mbp	Copyright © 2000, 2004 by Martin Pool.
OpenSSL for Python-2.7	0.9.8	Copyright © 1998-2008 The OpenSSL Project. All rights reserved. Copyright © 1995-1998 Eric Young (eay@cryptsoft.com) All rights reserved.
PETSc	2.3.3-p15; Released Sept-23-2008	COPYRIGHT © 1995-2004 UNIVERSITY OF CHICAGO
PBMPlus	8.3	Copyright © 1998-2000 Jef Poskanzer et al.
Profile-Pstats for Python-2.7	(no version)	Copyright © 1994 by InfoSeek Corporation. All rights reserved.
Python Console for Python-2.7	1.1a1	Copyright © 1999-2001 by Secret Labs AB Copyright © 1999-2001 by Fredrik Lundh
TCK/TK for Python-2.7	8.5	Copyright © Regents of the University of California, Sun Microsystems, Inc., Scriptics Corporation, and other parties.
TIFF	8.5	Copyright © 1991-1997 Sam Leffler Copyright © 1991-1997 Silicon Graphics, Inc.
Trace for Python-2.7	(no version)	Copyright ©2000, Mojam Media, Inc., all rights reserved. Copyright © 1999, Bioreason, Inc., all rights reserved. Copyright © 1995-1997, Automatrix, Inc., all rights reserved. Copyright © 1991-1995, Stichting Mathematisch Centrum, all rights reserved.
Unzip (from Info-ZIP)	6.0	Copyright © 1990-2009 Info-ZIP. All rights reserved.
uu for Python-2.7	(no version)	Copyright© 1994 by Lance Ellinghouse, Cathedral City, California Republic, United States of America. All rights reserved.
Zip (from Info-ZIP)	3.0	Copyright © 1990-2009 Info-ZIP. All rights reserved.
Zlib	1.2.8	Copyright © 1995-2004 Jean-Loup Gailly and Mark Adler
Under Apache 1.1:		
Jakarta	2.07	Copyright 1999–2004, The Apache Software Foundation
Under Apache 2.0:		

SCXML	Working Draft 16 December 2010	Copyright 1999–2005, The Apache Software Foundation
Xerces-J	2.6.2	Copyright © 2004, The Apache Software Foundation
Under “Artistic License”, Version 1.0:		
PERL	5.6.0	Copyright (C) 1993-2005, by Larry Wall and others.
Under BSD 2-Clause:		
Pychecker for Python-2.7	0.8.17	Copyright © 2000-2001, MetaSlash Inc. All rights reserved.
Under BSD 3-Clause:		
_random for Python-2.7	(no version)	Copyright © 1997-2002 Makoto Masumoto and Takuji Nishimura. All rights reserved.
_wincon.c for Python-2.7	(no version)	Copyright © 1999-2001 by Secret Labs AB. Copyright © 1999-2001 by Fredrik Lundh.
MAGMA (Matrix Algebra on GPU and Multicore Architectures)	1.4	Copyright © 2009-2013 by Innovative Computing Laboratory Electrical Engineering and Computer Science Department, University of Tennessee
TCP Server Socket Library	(no version)	Copyright © 2000, Jasen Plietz. All rights reserved.
Modelisar FMI	2.0	Copyright © 2008-2010, MODELISAR consortium. All rights reserved.
NumPy for Python-2.7	1.6.2	Copyright © 2005-2013, NumPy Developers.
Socket for Python-2.7	(no version)	Copyright © 1995, 1996, 1997, and 1998 WIDE Project. All rights reserved.
Visualization Toolkit (VTK)	5.10	Copyright © 1993-2008 Ken Martin, Will Schroeder, Bill Lorensen
Under MIT License:		
asyncore-asynchat for Python-2.7	(no version)	Copyright ©1996 by Sam Rushing. All rights reserved.
Cookie for Python-2.7	(no version)	Copyright © 2000 by Timothy O'Malley. All rights reserved.
ctypes for Python-2.7	(no version)	Copyright © 2000-2006 by Thomas Heller. All rights reserved.
libffi for Python-2.7	(no version)	Copyright © 1996-2003 by Red Hat, Inc. All rights reserved.
Mesa	6.0.1	Copyright © 1999-2007 by Brian Paul All Rights Reserved.
pyexpat for Python-2.7	2.0.0	Copyright © 1998, 1999, 2000 Thai Open Source Software Center Ltd and Clark Cooper
XML ExPat Parser	1.95.4	Copyright © 1998, 1999, 2000 Thai Open Source Software Center Ltd
Xmlrpclib for Python-2.7	1.0.1	Copyright © 1999-2002 by Secret Labs AB Copyright © 1999-2002 by Fredrik Lundh
Under Python Free Software License:		
Python	2.7.3	Copyright © 2001-2012 Python Software Foundation. All rights reserved.
Python Extensions for Windows for Python-2.7	Build 217	Copyright © Mark Hammond

Additional license terms:

FOX Toolkit Library License Addendum

1. License. The FOX Toolkit Library ("The Library") is licensed under GNU Lesser General Public License as published by the Free Software Foundation, version 3 of the License, or (at your option) any later version.
2. Relinking Exemption. You may distribute a combined work using a statically linked, unmodified copy of the FOX Library under terms of your choice, without the relinking requirement stipulated under the GNU Lesser Public License, subject to the following conditions:
 - a. This static relinking exemption covers only the FOX Toolkit Library. Other libraries which the FOX Library may need are covered by their own respective licenses.
 - b. Modification of the configure scripts, makefiles, or installation tools of the FOX Library to support a specific platform does not constitute creating a modified copy based on the FOX Library.
 - c. Programs or binaries statically linked with the FOX Library must be identified as such by including, in the Documentation or by other means (for example in the About Box or Online Help), the following statement:

"This software uses the FOX Toolkit Library (<http://www.fox-toolkit.org>)."
 - d. Subclassing from Objects or Widgets supplied by the Library involves no modifications to the source code of the Library itself, and does not constitute creating a modified copy based on the Library.
3. This FOX Toolkit Library License Addendum is NOT propagated to derivative works. When making derivative works based on the FOX Library, this license addendum ceases to apply, and code will revert to the simple GNU Lesser General Public License. The Relinking Exemption only applies to unmodified versions of the FOX Library.
4. If you do not accept or are unable to meet the conditions under (2), you may continue to distribute the combined work under the original GNU Lesser General Public License.

FreeType BSD-Like License:

0. Definitions

Throughout this license, the terms 'package', 'FreeType Project', and 'FreeType archive' refer to the set of files originally distributed by the authors (David Turner, Robert Wilhelm, and Werner Lemberg) as the 'FreeType Project', be they named as alpha, beta or final release.

'You' refers to the licensee, or person using the project, where 'using' is a generic term including compiling the project's source code as well as linking it to form a 'program' or 'executable'.

This program is referred to as 'a program using the FreeType engine'.

This license applies to all files distributed in the original FreeType Project, including all source code, binaries and documentation, unless otherwise stated in the file in its original, unmodified form as distributed in the original archive. If you are unsure whether or not a particular file is covered by this license, you must contact us to verify this.

The FreeType Project is copyright (C) 1996-2000 by David Turner, Robert Wilhelm, and Werner Lemberg. All rights reserved except as specified below.

1. No Warranty

THE FREETYPE PROJECT IS PROVIDED 'AS IS' WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL ANY OF THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY DAMAGES CAUSED BY THE USE OR THE INABILITY TO USE, OF THE FREETYPE PROJECT.

2. Redistribution

This license grants a worldwide, royalty-free, perpetual and irrevocable right and license to use, execute, perform, compile, display, copy, create derivative works of, distribute and sublicense the FreeType Project (in both source and object code forms) and derivative works thereof for any purpose; and to authorize others to exercise some or all of the rights granted herein, subject to the following conditions:

- Redistribution of source code must retain this license file ('FTL.TXT') unaltered; any additions, deletions or changes to the original files must be clearly indicated in accompanying documentation. The copyright notices of the unaltered, original files must be preserved in all copies of source files.
- Redistribution in binary form must provide a disclaimer that states that the software is based in part of the work of the FreeType Team, in the distribution documentation. We also encourage you to put an URL to the FreeType web page in your documentation, though this isn't mandatory.

These conditions apply to any software derived from or based on the FreeType Project, not just the unmodified files. If you use our work, you must acknowledge us. However, no fee need be paid to us.

3. Advertising

Neither the FreeType authors and contributors nor you shall use the name of the other for commercial, advertising, or promotional purposes without specific prior written permission.

We suggest, but do not require, that you use one or more of the following phrases to refer to this software in your documentation or advertising materials: 'FreeType Project', 'FreeType Engine', 'FreeType library', or 'FreeType Distribution'.

As you have not signed this license, you are not required to accept it. However, as the FreeType Project is copyrighted material, only this license, or another one contracted with the authors, grants you the right to use, distribute, and modify it. Therefore, by using, distributing, or modifying the FreeType Project, you indicate that you understand and accept all the terms of this license.

4. Contacts

There are two mailing lists related to FreeType:

- freetype@nongnu.org

Discusses general use and applications of FreeType, as well as future and wanted additions to the library and distribution. If you are looking for support, start in this list if you haven't found anything to help you in the documentation.

- freetype-devel@nongnu.org

Discusses bugs, as well as engine internals, design issues, specific licenses, porting, etc.

Our home page can be found at <http://www.freetype.org>

Info-ZIP (Unzip and Zip):

This is version 2009-Jan-02 of the Info-ZIP license. The definitive version of this document should be available at <ftp://ftp.info-zip.org/pub/infozip/license.html> indefinitely and a copy at <http://www.info-zip.org/pub/infozip/license.html>.

Copyright (c) 1990-2009 Info-ZIP. All rights reserved.

For the purposes of this copyright and license, "Info-ZIP" is defined as the following set of individuals:

Mark Adler, John Bush, Karl Davis, Harald Denker, Jean-Michel Dubois, Jean-loup Gailly, Hunter Goatley, Ed Gordon, Ian Gorman, Chris Herborth, Dirk Haase, Greg Hartwig, Robert Heath, Jonathan Hudson, Paul Kienitz, David Kirschbaum, Johnny Lee, Onno van der Linden, Igor Mandrichenko, Steve P. Miller, Sergio Monesi, Keith Owens, George Petrov, Greg Roelofs, Kai Uwe Rommel, Steve Salisbury, Dave Smith, Steven M. Schweda, Christian Spieler, Cosmin Truta, Antoine Verheijen, Paul von Behren, Rich Wales, Mike White.

This software is provided "as is," without warranty of any kind, express or implied. In no event shall Info-ZIP or its contributors be held liable for any direct, indirect, incidental, special or consequential damages arising out of the use of or inability to use this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the above disclaimer and the following restrictions:

1. Redistributions of source code (in whole or in part) must retain the above copyright notice, definition, disclaimer, and this list of conditions.
2. Redistributions in binary form (compiled executables and libraries) must reproduce the above copyright notice, definition, disclaimer, and this list of conditions in documentation and/or other materials provided with the distribution. Additional documentation is not needed for executables where a command line license option provides these and a note regarding this option is in the executable's startup banner. The sole exception to this condition is redistribution of a standard UnZipSFX binary (including SFXWiz) as part of a self-extracting archive; that is permitted without inclusion of this license, as long as the normal SFX banner has not been removed from the binary or disabled.

Altered versions--including, but not limited to, ports to new operating systems, existing ports with new graphical interfaces, versions with modified or added functionality, and dynamic, shared, or static library versions not from Info-ZIP--must be plainly marked as such and must not be misrepresented as being the original source or, if binaries, compiled from the original source. Such altered versions also must not be misrepresented as being Info-ZIP releases--including, but not limited to, labeling of the altered versions with the names "Info-ZIP" (or any variation thereof, including, but not limited to, different capitalizations), "Pocket UnZip," "WiZ" or "MacZip" without the explicit permission of Info-ZIP. Such altered versions are further prohibited from misrepresentative use of the Zip-Bugs or Info-ZIP e-mail addresses or the Info-ZIP URL(s), such as to imply Info-ZIP will provide support for the altered versions.

3. Info-ZIP retains the right to use the names "Info-ZIP," "Zip," "UnZip," "UnZipSFX," "WiZ," "Pocket UnZip," "Pocket Zip," and "MacZip" for its own source and binary releases.

LibPng:

COPYRIGHT NOTICE, DISCLAIMER, and LICENSE:

If you modify libpng you may insert additional notices immediately following this sentence.

libpng versions 1.0.7, July 1, 2000, through 1.0.13, April 15, 2002, are Copyright (c) 2000-2002 Glenn Randers-Pehrson and are distributed according to the same disclaimer and license as libpng-1.0.6 with the following individuals added to the list of Contributing Authors

Simon-Pierre Cadieux
Eric S. Raymond
Gilles Vollant

and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs. This library is provided with all faults, and the entire risk of satisfactory quality, performance, accuracy, and effort is with the user.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are Copyright (c) 1998, 1999 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-0.96, with the following individuals added to the list of Contributing Authors:

Tom Lane
Glenn Randers-Pehrson
Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are Copyright (c) 1996, 1997 Andreas Dilger

Distributed according to the same disclaimer and license as libpng-0.88, with the following individuals added to the list of Contributing Authors:

John Bowler
Kevin Bracey
Sam Bushell
Magnus Holmgren
Greg Roelofs
Tom Tanner

libpng versions 0.5, May 1995, through 0.88, January 1996, are Copyright (c) 1995, 1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

Andreas Dilger
Dave Martindale
Guy Eric Schalnat
Paul Schmidt
Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.
2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.
3. This Copyright notice may not be removed or altered from any source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

A "png_get_copyright" function is available, for convenient use in "about" boxes and the like:

```
printf("%s",png_get_copyright(NULL));
```

Also, the PNG logo (in PNG format, of course) is supplied in the files "pngbar.png" and "pngbar.jpg (88x31) and "pngnow.png" (98x31).

Libpng is OSI Certified Open Source Software. OSI Certified Open Source is a certification mark of the Open Source Initiative.

Glenn Randers-Pehrson
randeg@alum.rpi.edu
April 15, 2002

Natural Order Sort:

Copyright © 2000, 2004 by Martin Pool.

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

This license applies only to the C implementation. You are free to reimplement the idea from scratch in any language.

OpenSSL for Python-2.7:

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.

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 (ey@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

Original SSLeay License

Copyright © 1995-1998 Eric Young (ey@cryptsoft.com)

All rights reserved.

This package is an SSL implementation written by Eric Young (ey@cryptsoft.com). The implementation was written so as to conform with Netscape's SSL.

This library is free for commercial and non-commercial use as long as the following conditions are adhered to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:
"This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)" The word 'cryptographic' can be left out if the routines from the library being used are not cryptographic related :-).
4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement:
"This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The license and distribution terms for any publically available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution license [including the GNU Public License.]

PETSc:

This program discloses material protectable under copyright laws of the United States. Permission to copy and modify this software and its documentation is hereby granted, provided that this notice is retained thereon and on all copies or modifications. The University of Chicago makes no representations as to the suitability and operability of this software for any purpose. It is provided "as is" without express or implied warranty. Permission is hereby granted to use, reproduce, prepare derivative works, and to redistribute to others, so long as this original copyright notice is retained.

Software authors
Mathematics and Computer Science Division
Argonne National Laboratory,
Argonne IL 60439 FAX: (630) 252-5986

Any questions or comments on the software may be directed to petsc-maint@mcs.anl.gov.
Argonne National Laboratory with facilities in the state of Illinois, is owned by The United States Government, and operated by the University of Chicago under provision of a contract with the Department of Energy.

DISCLAIMER

THIS PROGRAM WAS PREPARED AS AN ACCOUNT OF WORK SPONSORED BY AN AGENCY OF THE UNITED STATES GOVERNMENT. NEITHER THE UNITED STATES GOVERNMENT NOR ANY AGENCY THEREOF, NOR THE UNIVERSITY OF CHICAGO, NOR ANY OF THEIR EMPLOYEES OR OFFICERS, MAKES ANY WARRANTY, EXPRESS OR IMPLIED, OR ASSUMES ANY LEGAL LIABILITY OR RESPONSIBILITY FOR THE ACCURACY, COMPLETENESS, OR USEFULNESS OF ANY INFORMATION, APPARATUS, PRODUCT, OR PROCESS DISCLOSED, OR REPRESENTS THAT ITS USE WOULD NOT INFRINGE PRIVATELY OWNED RIGHTS. REFERENCE HEREIN TO ANY SPECIFIC COMMERCIAL PRODUCT, PROCESS, OR SERVICE BY TRADE NAME, TRADEMARK, MANUFACTURER, OR OTHERWISE, DOES NOT NECESSARILY CONSTITUTE OR IMPLY ITS ENDORSEMENT, RECOMMENDATION, OR FAVORING BY THE UNITED STATES GOVERNMENT OR ANY AGENCY THEREOF. THE VIEW AND OPINIONS OF AUTHORS EXPRESSED HEREIN DO NOT NECESSARILY STATE OR REFLECT THOSE OF THE UNITED STATES GOVERNMENT OR ANY AGENCY THEREOF.

PBMPlus:

Copyright 1998 - 2000 Jef Poskanzer et al.

Permission to use, copy, modify, and distribute its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appears in all copies and that both the copyright notice and this permission notice appear in supporting documentation. This software is provided "as is" without express or implied warranty.

Profile-Pstats for Python-2.7:

Copyright 1994, by InfoSeek Corporation, All rights reserved.

Written by James Roskind

Permission to use, copy, modify, and distribute this Python software and its associated documentation for any purpose (subject to the restriction in the following sentence) without fee is hereby granted, provided that the above copyright notice appears in all copies, and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of InfoSeek not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. This permission is explicitly restricted to the copying and modification of the software to remain in Python, compiled Python, or other languages (such as C) wherein the modified or derived code is exclusively imported into a Python module.

INFOSEEK CORPORATION DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL INFOSEEK CORPORATION BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Python Console for Python-2.7:

Copyright (c) 1999-2001 by Secret Labs AB

Copyright (c) 1999-2001 by Fredrik Lundh

By obtaining, using, and/or copying this software and/or its associated documentation, you agree that you have read, understood, and will comply with the following terms and conditions:

Permission to use, copy, modify, and distribute this software and its associated documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appears in all copies, and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Secret Labs AB or the author not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

SECRET LABS AB AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL SECRET LABS AB OR THE AUTHOR BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

TCL/TK for Python-2.7:

This software is copyrighted by the Regents of the University of California, Sun Microsystems, Inc., Scriptics Corporation, and other parties. The following terms apply to all files associated with the software unless explicitly disclaimed in individual files.

The authors hereby grant permission to use, copy, modify, distribute, and license this software and its documentation for any purpose, provided that existing copyright notices are retained in all copies and that this notice is included verbatim in any distributions. No written agreement, license, or royalty fee is required for any of the authorized uses. Modifications to this software may be copyrighted by their authors and need not follow the licensing terms described here, provided that the new terms are clearly indicated on the first page of each file where they apply.

IN NO EVENT SHALL THE AUTHORS OR DISTRIBUTORS BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE, ITS DOCUMENTATION, OR ANY DERIVATIVES THEREOF, EVEN IF THE AUTHORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE AUTHORS AND DISTRIBUTORS SPECIFICALLY DISCLAIM ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. THIS SOFTWARE IS PROVIDED ON AN "AS IS" BASIS, AND THE AUTHORS AND DISTRIBUTORS HAVE NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

GOVERNMENT USE: If you are acquiring this software on behalf of the U.S. government, the Government shall have only "Restricted Rights" in the software and related documentation as defined in the Federal Acquisition Regulations (FARs) in Clause 52.227.19 (c) (2). If you are acquiring the software on behalf of the Department of Defense, the software shall be classified as "Commercial Computer Software" and the Government shall have only "Restricted Rights" as defined in Clause 252.227-7013 (c) (1) of DFARs. Notwithstanding the foregoing, the authors grant the U.S. Government and others acting in its behalf permission to use and distribute the software in accordance with the terms specified in this license.

TIFF:

Copyright (c) 1991-1997 Sam Leffler
Copyright (c) 1991-1997 Silicon Graphics, Inc.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that (i) the above copyright notices and this permission notice appear in all copies of the software and related documentation, and (ii) the names of Sam Leffler and Silicon Graphics may not be used in any advertising or publicity relating to the software without the specific, prior written permission of Sam Leffler and Silicon Graphics.

THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

IN NO EVENT SHALL SAM LEFFLER OR SILICON GRAPHICS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE."

Trace for Python-2.7:

Copyright 2000, Mojam Media, Inc., all rights reserved. Author: Skip Montanaro

Copyright 1999, Bioreason, Inc., all rights reserved. Author: Andrew Dalke

Copyright 1995-1997, Automatrix, Inc., all rights reserved. Author: Skip Montanaro

Copyright 1991-1995, Stichting Mathematisch Centrum, all rights reserved.

Permission to use, copy, modify, and distribute this Python software and its associated documentation for any purpose without fee is hereby granted, provided that the above copyright notice appears in all copies, and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of neither Automatrix, Bioreason or Mojam Media be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

uu for Python-2.7:

Copyright 1994 by Lance Ellinghouse
Cathedral City, California Republic, United States of America.
All Rights Reserved

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of Lance Ellinghouse not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

LANCE ELLINGHOUSE DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL LANCE ELLINGHOUSE CENTRUM BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Preface

This section lists various resources that are available for help with using Abaqus Unified FEA software.

Support

Both technical software support (for problems with creating a model or performing an analysis) and systems support (for installation, licensing, and hardware-related problems) for Abaqus are offered through a global network of support offices, as well as through our online support system. Contact information for our regional offices is accessible from **SIMULIA→Locations** at www.3ds.com/simulia. The online support system is accessible by selecting the **SUBMIT A REQUEST** link at **Support - Dassault Systèmes** (<http://www.3ds.com/support>).

Online support

Dassault Systèmes provides a knowledge base of questions and answers, solutions to questions that we have answered, and guidelines on how to use Abaqus, Engineering Process Composer, Isight, Tosca, fe-safe, and other SIMULIA products. The knowledge base is available by using the **Search our Knowledge** option on www.3ds.com/support (<http://www.3ds.com/support>).

By using the online support system, you can also submit new requests for support. All support/service requests are tracked. If you contact us by means outside the system to discuss an existing support problem and you know the support request number, please mention it so that we can query the support system to see what the latest action has been.

Training

All SIMULIA regional offices offer regularly scheduled public training classes. The courses are offered in a traditional classroom form and via the Web. We also provide training seminars at customer sites. All training classes and seminars include workshops to provide as much practical experience with Abaqus as possible. For a schedule and descriptions of available classes, see the **Training** link at www.3ds.com/products-services/simulia (www.3ds.com/products-services/simulia) or call your support office.

Feedback

We welcome any suggestions for improvements to Abaqus software, the support tool, or documentation. We will ensure that any enhancement requests you make are considered for future releases. If you wish to make a suggestion about the service or products, refer to www.3ds.com/simulia. Complaints should be made by contacting your support office or by visiting **SIMULIA→Quality Assurance** at www.3ds.com/simulia (www.3ds.com/simulia).

Contents

1. Overview	
2. Installing Abaqus	
Installation procedures	2.1
Abaqus documentation installation details	2.2
Abaqus FLEXnet licensing installation details	2.3
Abaqus product installation details	2.4
Uninstalling Abaqus products or licensing	2.5
Uninstalling Abaqus documentation	2.6
3. Abaqus licensing	
FLEXnet network licensing	3.1
Updating an Abaqus FLEXnet license file	3.2
Upgrading a FLEXnet license server and administration tools	3.3
Redundant FLEXnet license server configurations	3.4
FLEXnet license server manager lmgrd	3.5
FLEXnet Licensing administration tools	3.6
Using the FLEXnet options file	3.7
Reporting FLEXnet license usage data	3.8
Using the ds1sstat utility for a Dassault Systèmes license server	3.9
Using the reporttool utility	3.10
4. Customizing the Abaqus environment	
Using the Abaqus environment files	4.1
Defining analysis batch queues	4.2
Examples	4.3
5. Customizing Abaqus/CAE and Abaqus/Viewer	
Customizing the user interface	5.1
Configuring printers	5.2
Tuning graphics cards	5.3
A. Abaqus release directories and files	
Installation subdirectories	A.1
Command-line batch files and links	A.2
Documentation files	A.3

CONTENTS

B. Configuring Abaqus documentation

- Using an existing non-Abaqus web server B.1
- Setting the documentation URL after installation B.2

C. Accessing remote file systems for installation and execution

- Running Abaqus remotely on Linux C.1
- Using a network ODB connector C.2

D. Verification procedure

E. Troubleshooting Abaqus FLEXnet licensing

- General questions E.1
- FLEXnet license server operation E.2
- Special situations E.3
- Other licensing questions and problems E.4

1. Overview

This guide contains information about installing and customizing Abaqus 2016.

The installation of Abaqus involves the following tasks:

- Installing the Abaqus documentation.
- Installing an Abaqus license server.
- Installing Abaqus products.

For a description of the installation procedures, see Chapter 2, “Installing Abaqus.”

Network licensing and procedures to administer the license server are described in Chapter 3, “Abaqus licensing.”

The environment file provides options for customizing an Abaqus installation. Specifying the file system to be used to store scratch files is particularly important. For details on how to set local defaults and adjust the system configuration to run Abaqus jobs efficiently, see Chapter 4, “Customizing the Abaqus environment.”

For more information on customizing the Abaqus user interface, printer configuration, and graphic card tuning, see Chapter 5, “Customizing Abaqus/CAE and Abaqus/Viewer.”

2. Installing Abaqus

This chapter describes the installation procedures for Abaqus documentation, licensing, and products. A complete Abaqus installation requires the use of several installers for different product components.

Before starting the Abaqus installers, you should confirm that your computer is compatible with this release. Compare your system's configuration to the Abaqus system requirements described in the Program Directory. The Program Directory is additional documentation, similar to release notes, available at <http://media.3ds.com/support/progdir>. Choose **SIMULIA** as the product **Line** and **Established Products 2016** as the **Level**, then choose **Prerequisites - Abaqus** in the left-hand pane. A support contract and private login account are required for access (see https://iam.3ds.com/self_service/login/service/websupport/).

Please also check the Program Directory for additional information, especially pertaining to maintenance (hot fix) releases.

To start the installers on Windows, you must belong to the Administrators group or have the privileges assigned to the Administrators group.

A complete Abaqus installation includes several components. **You must perform the installations in the following order:**

1. *Abaqus documentation* (optional)
2. *License server*: You can use either FLEXnet licensing or Dassault Systèmes licensing, depending on which type of license file you have. See “Dassault Systèmes license server installation,” Section 2.1.4 or “FLEXnet Licensing installation,” Section 2.1.5.
3. *Abaqus services*: Abaqus/Standard solver, Abaqus/Explicit solver, and Abaqus/CFD solver. This software includes the SIMULIA Co-Simulation Engine and the foundation of all Abaqus APIs. See “Abaqus product installation,” Section 2.1.6.
4. *CAA developer software for Abaqus APIs*: output database (ODB) C++ API, user subroutines API, Abaqus Scripting Interface API, and SIMULIA Co-Simulation Engine C++ API. This software includes libraries and **.h** header files. The CAA (Component Application Architecture) API software is needed for using user subroutines and compiling and building ODB, SIM, or results file postprocessing applications. See “Abaqus product installation,” Section 2.1.6.
5. *Abaqus/CAE*: interactive GUI; also the command-line interface to Abaqus solvers (see “Using Abaqus commands,” Section 2.1.2). See “Abaqus product installation,” Section 2.1.6.

Important: You must *always* install both the Abaqus services (solvers) and Abaqus/CAE. They cannot be installed and used separately or individually.

During the installation procedures, default values are supplied whenever possible, and it is highly recommended that you use the defaults unless you have a good reason to change them. It is also recommended that you always install the CAA (API) developer software, even if you are not sure that you will need it; otherwise, you may find that some components need to be installed later before you can upgrade to a maintenance release (fix pack / hot fix). When you later install maintenance releases for Abaqus 2016, you will not be able to add any new product installer options.

Abaqus documentation should be installed on at least one computer on your network. Abaqus documentation should be installed before licensing and products because the product installer uses the documentation URL to configure network access to the online documentation.

2.1 Installation procedures

Follow the instructions below to download, extract, and run the installers.

2.1.1 Downloading and extracting the Abaqus media

If you download the Abaqus installation media, you must extract (uncompress) the archive files into a common directory before running the installers.

1. Download the media archive files to any location. The files are **.tar** format on both Windows and Linux. If you have the standard Abaqus-only media, the file names are similar to the following:

```
xxx_Abaqus.1-3.tar
xxx_Abaqus.2-3.tar
xxx_Abaqus.3-3.tar
```

The three **.tar** files must be uncompressed (combining their contents) before running any of the installers.

If you have purchased the SIMULIA extended products portfolio, the media consists of five **.tar** files. These media contain several additional products and utilities (in addition to Abaqus): Isight, Tosca, fe-safe, and CAD associative interfaces.

2. Create a directory into which you will extract the archive files.

In the instructions throughout this guide, *unload_dir* represents the full path to this extraction directory; for example, **C:\unload** on Windows or **/tmp/unload/** on Linux.

3. Extract each of the **.tar** files into the common *unload_dir* directory. On Windows, you can use any unzip utility; for example, the open-source 7-Zip utility, which handles the **.tar** format. The 7-Zip overwrite mode does not matter because there are no file or directory (folder) conflicts to merge within the combined directory structure.

You can follow the detailed steps listed below, if needed:

Windows platforms

1. Download all of the **.tar** files to a single folder.
2. Navigate to this folder in Windows Explorer.
3. Select all of the **.tar** files (either using the mouse to drag-select, Ctrl-A to select all, or click-select the first file and Shift-click-select the last).
4. With all files selected, right-click on the selection and choose **7-Zip > Extract Here**.

Linux platforms

1. Download all of the `.tar` files to a single directory.
2. In a shell/terminal/console, change directory (`cd`) to the download directory and execute the command `tar -xf filename` on each of the files, for example:

```
tar -xf filename.1-3.tar
tar -xf filename.2-3.tar
tar -xf filename.3-3.tar
```

For the standard Abaqus-only media, the contents of your `unload_dir` directory should be as shown in Figure 2–1.

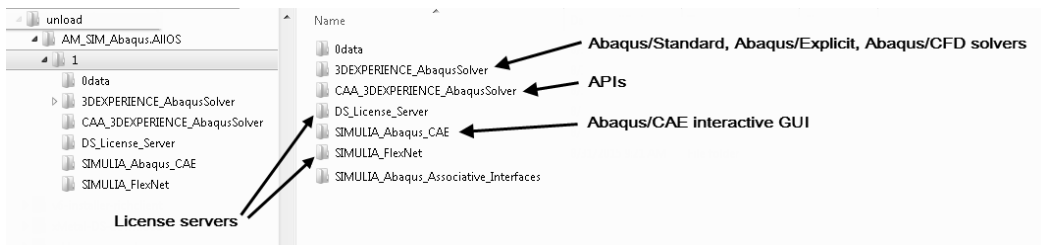
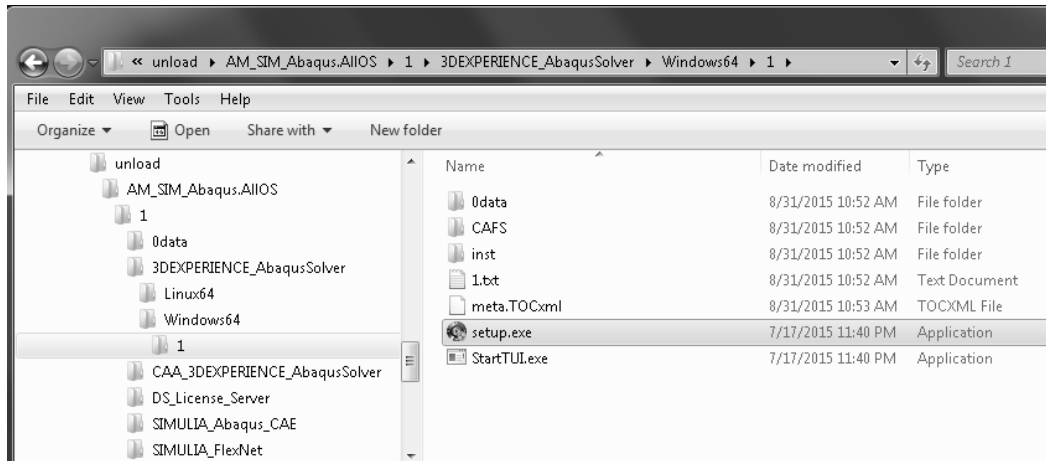


Figure 2–1 Contents of `unload_dir` directory for the standard Abaqus-only media.

Each subdirectory under `unload_dir\AM_SIM_Abaqus.AIOS\1\` contains an installer. Each installer executable installs different components. For example, to install the Windows version of the Abaqus services (solvers), you run the following `setup.exe` file:

```
unload_dir\AM_SIM_Abaqus.AIOS\1\3DEXPERIENCE_AbaqusSolver\
Windows64\1\setup.exe
```

INSTALLING Abaqus



To start the same installer for Linux, run the following shell script:

```
unload_dir/AM_SIM_Abaqus.AllOS/1/3DEXPERIENCE_AbaqusSolver/  
Linux64/1/StartGUI.sh
```

If you have the SIMULIA extended products media, there are several additional products and utilities delivered (in addition to Abaqus): Isight, Tosca, fe-safe, and CAD associative interfaces. The extended products media contains three product/component directories instead of one:

```
unload_dir/media_name/1/      Same contents as Abaqus-only media  
unload_dir/media_name/2/      Tosca and Isight  
unload_dir/media_name/3/      fe-safe
```

Tosca must be configured with the Abaqus/CAE installation to allow the Abaqus Optimization module to work correctly; the Tosca installer handles this setting. Tosca must be installed after Abaqus/CAE.

The download media for the documentation are split into two **.tar** archive files. As with the software media, you must extract these **.tar** files into a common, merged directory before starting the documentation installer.

2.1.2 Using Abaqus commands

The Abaqus/CAE installer creates two commands to run Abaqus from the command line: **abaqus** and **abq2016**. The Windows batch files or Linux links are created in a directory named **Commands**. On both Windows and Linux, the installer prompts you to choose the location for this directory.

On Windows, the default location is under the folder indicated by the environment variable **SystemDrive**:

```
%SystemDrive%\SIMULIA\Commands\
```

The standard default location of this folder is:

```
C:\SIMULIA\Commands\
```

For a list of all Abaqus command-line options, use the **help** command-line option with either the **abaqus** or **abq2016** command. For a detailed description of Abaqus command-line options, see Chapter 3, “Job Execution,” of the Abaqus Analysis User’s Guide.

On Windows platforms the **Commands** directory path is added automatically to each user’s **PATH** environment variable during installation. On Linux platforms the **Commands** directory should be prepended to each user’s path manually.

For hot fix (maintenance) releases, the release-specific command has the hot fix number appended. For hot fix 3, for example, the command should be named **abq2016hf3**.

2.1.3 Documentation installation

Online documentation for all Abaqus releases, including general and extended functionality releases, is provided in two formats: HTML and PDF. There are several options for installing the documentation and making it available to your Abaqus users:

- Install HTML and PDF on a server on your network, and use web server software to serve both formats to users (this is the most common choice).
- Install multiple copies of the PDF files on individual users’ machines, where they can view it locally using a PDF viewer such as Adobe Acrobat Reader.
- Skip the documentation installer, and simply copy the PDF files from the DVD to any computers or disks.

You must have write permission in the directory where documentation is installed. Administrator privileges are required to run the installer on Windows platforms.

The commands and options used to install documentation are described below. For more details about the documentation installation options, see “Abaqus documentation installation details,” Section 2.2.

To install Abaqus documentation:

1. Download the documentation media **.tar** files to any location. Extract the two **.tar** files into a common directory, combining their contents, similar to the software product **.tar** files.
2. From the extraction directory, run the following executable file to start the documentation installer:

Linux platforms

```
extract_dir/AM_DOC_SIM_EstPrd.AllOS/1/  
DOC_SIMULIA_Abaqus_fe-safe/1/Abaqus_2016/setup  
[-replay full_path_to_replay_file]
```

Windows platforms

```
extract_dir\AM_DOC_SIM_EstPrd.AllOS\1\  
DOC_SIMULIA_Abaqus_fe-safe\1\Abaqus_2016\setup.exe  
[-replay full_path_to_replay_file]
```

Command option

-replay *full_path_to_replay_file*

Use this option to perform additional identical documentation installations without having to reenter the same information for each installation. For more information, see “Repeating documentation installations on multiple computers,” Section 2.2.2.

3. Follow the instructions in the dialog boxes that appear to complete the installation. Choose which formats of documentation to install:
 - **HTML and PDF**
 - **PDF only**
4. If you installed the **HTML and PDF** format and selected the “Abaqus web server” or “no web server” option, you are provided with a URL that points to your HTML documentation when the installer completes. This URL is also written to a log file in the following location:

doc_install_dir/Documentation/installation_info/v2016/info.log

If you selected the “existing non-Abaqus web server” option, a sample URL is written to the **info.log** file. To access the HTML documentation from the Abaqus products, you must provide a valid URL during each subsequent Abaqus product installation.

5. If you installed only the PDF documentation, the path to the **index.pdf** file is shown.

To copy the PDF documentation (without running the installer):

1. Find the following directory:

```
extract_dir/AM_DOC_SIM_EstPrd.AllOS/1/  
DOC_SIMULIA_Abaqus_fe-safe/1/Abaqus_2016/pdf_docs/
```

Copy the entire contents of this directory.

2. To view the documentation, open the file **index.pdf** in a PDF viewer or web browser.

You may copy the PDF documentation in accordance with the following terms:

- Use of this material is governed by your Abaqus software license agreement.
- Keep the documentation confidential and utilize your best efforts to prevent unauthorized disclosure or use of any Abaqus Program including the Abaqus documentation for such program.
- Treat all Abaqus Programs including the Abaqus documentation with the same degree of care that you treat like information of your own that you do not want publicly disclosed or the subject of unauthorized access or use.
- Do not make or permit to be made any more copies of any Abaqus Program including the Abaqus documentation than are authorized by your Abaqus software license agreement.

2.1.4 Dassault Systèmes license server installation

You can use either FLEXnet or Dassault Systèmes licensing with Abaqus, Tosca, Isight, and fe-safe, depending on which type of license file you have. You can choose which type of license server to install. For Dassault Systèmes licensing, you must install the Dassault Systèmes license server (DSLS) software on any computer with a supported platform (operating system).

A new license file allows you to run all maintenance deliveries of the current version as well as the next two major releases. You must request a new license file for DSLS in order to run Abaqus 2016 from the command line.

The DSLS software package includes the *Dassault Systèmes License Server Installation and Configuration Guide (DSLS.pdf)*, which contains instructions for configuring the license server and administering licenses. Refer to this guide for all DSLS licensing operations. You can find the **DSLS.pdf** file in the following location:

```
unload_dir/AM_SIM_Abaqus.AllOS/1/DS_License_Server/1/DSLS.pdf
```

To install the Dassault Systèmes license server, you must be an Administrator on Windows or root on Linux. The Dassault Systèmes license server is supported only with Abaqus 6.12 and higher.

Virtual machines, such as VMware, are not supported. It is not possible to install or run the Dassault Systèmes license server on a virtual machine.

The Dassault Systèmes license server should be installed on only one computer unless you are using a redundant license server configuration. The server computers that you choose for documentation and

INSTALLING Abaqus

licensing should be accessible by all computers on the network where you plan to run Abaqus and should be stable machines that are not frequently rebooted or shut down.

Automatic configuration is recommended for most users, particularly those whose primary responsibility is not computer system or network administration. You can allow the installer to install, configure, and start the Dassault Systèmes license server automatically.

After installation, you can use the **dslsstat** utility to check the DSLS status; see “Using the **dslsstat** utility for a Dassault Systèmes license server,” Section 3.9.

If you are using an older version of the DSLS server software on Windows (3DEXPERIENCE R2015x or lower), it is recommended that you upgrade to DSLS R2016x with this Abaqus release.

Installing the Dassault Systèmes license server on Linux or Unix

To start the DSLS installer on Linux or Unix, follow these steps:

1. Set the **DISPLAY** environment variable for your computer.
2. Change directory (**cd**) into one of the following OS/platform-specific directories:

```
/unload_dir/AM_SIM_Abaqus.AllOS/1/DS_License_Server/1/AIX/  
/unload_dir/AM_SIM_Abaqus.AllOS/1/DS_License_Server/1/SUN/  
/unload_dir/AM_SIM_Abaqus.AllOS/1/DS_License_Server/1/RedHat_Suse/
```

3. Execute the file **startInstLicServ**.
4. Follow the instructions in the dialog boxes that appear to complete the installation.

After the installation finishes, you must use the Dassault Systèmes License Administration Tool to enroll the Abaqus licenses on the Dassault Systèmes license server; see “Enrolling Product Licenses” in the *Dassault Systèmes License Server Installation and Configuration Guide*. You must enroll the licenses before continuing with the Abaqus product installation.

Installing the Dassault Systèmes license server on Windows

To start the DSLS installer on Windows, double-click or execute the following file:

```
unload_dir\AM_SIM_Abaqus.AllOS\1\DS_License_Server\1\SetupDSLsmsi.exe
```

Follow the instructions in the dialog boxes that appear to complete the installation.

After the installation finishes, you must use the Dassault Systèmes License Administration Tool to enroll the Abaqus licenses on the Dassault Systèmes license server; see “Enrolling Product Licenses” in the *Dassault Systèmes License Server Installation and Configuration Guide*. You must enroll the licenses before continuing with the Abaqus product installation.

2.1.5 FLEXnet Licensing installation

You must have an Abaqus 2016 FLEXnet license file to set up a SIMULIA FLEXnet license server. FLEXnet Abaqus 2016 licensing uses FLEXnet Licensing Version 11.6.1. A license file allows you to

run subsequent Abaqus 2016 maintenance deliveries, previous releases dating back to Abaqus 6.2-7, and two future major releases. For example, a new license file for Abaqus 2016 can be used to run all Abaqus 2016 maintenance deliveries and any earlier releases of Abaqus still installed on your network.

FLEXnet Abaqus licensing should be installed on only one computer unless you are using a redundant license server configuration. The server computers that you choose for documentation and licensing should be accessible by all computers on the network where you plan to run Abaqus and should be stable machines that are not frequently rebooted or shut down.

Before installing the Abaqus products, you must have a FLEXnet network license server that is compatible with Abaqus 2016 installed and running on your network.

Select the licensing installation you want to perform from the following scenarios:

- Automatic configuration is recommended for most users, particularly those whose primary responsibility is not computer system or network administration. Allow the installer to install, configure, and start the FLEXnet license server for you automatically. An existing license server using a version of FLEXnet Licensing prior to Version 11.6.1 is upgraded automatically during the installation procedure if it was previously installed and configured using the Abaqus licensing installation procedure. For redundant license servers the installation will install and configure the Abaqus license servers automatically, but you will have to manually start each server. Use this method for ease of maintenance so that future upgrades can be configured automatically.
- Manual configuration is required for combining FLEXnet license files with other vendors. You will need to configure and start the FLEXnet license server manually. Use this method to allow the system administrator more control.

Abaqus 2016 requires **lmgrd** Version 11.6.1 or higher. If you integrate your Abaqus license features with other products using a FLEXnet Licensing combined license file, ensure that your version of **lmgrd** meets this requirement. For more information, refer to the FLEXnet Licensing End User Guide Version 11.6, which is available from the **Licensing** section of the **Support** page at www.3ds.com/simulia.

The commands and options used to install the SIMULIA FLEXnet license server and to stop and remove a license server are described below. For a more complete discussion of the FLEXnet licensing installation options, see “Abaqus FLEXnet licensing installation details,” Section 2.3. To replace an existing or expired FLEXnet license file, see “Updating an Abaqus FLEXnet license file,” Section 3.2; for details on installing a redundant license server, see “Redundant FLEXnet license server configurations,” Section 3.4.

To install Abaqus FLEXnet licensing:

1. On Windows, double-click or execute the following file:

```
unload_dir\AM_SIM_Abaqus.ALLOS\1\SIMULIA_Flexnet\Windows64\
1\setup.exe
```

INSTALLING Abaqus

On Linux, execute the following file:

```
unload_dir/AM_SIM_Abaqus.AllOS/1/SIMULIA_Flexnet/Linux64/1/StartGUI.sh
```

2. If you have an existing SIMULIA FLEXnet license server on this system, the installer attempts to stop the license server; if it is unable to do so, it provides the option to install only the licensing administration tools. If you install only the FLEXnet licensing administration tools, you must manually configure the license server. If the installer is unable to stop the existing license server and you want to complete a full automatic license server installation, you need to stop the existing server using the procedure described below.
3. Follow the instructions in the dialog boxes that appear to complete the installation.
4. On Linux platforms, the SIMULIA FLEXnet license server does not restart automatically after a system reboot by default. If you are installing a new FLEXnet license server, you must add the Abaqus licensing command to your system startup script if you want to have the FLEXnet license server restart automatically. On Windows platforms, an automatically configured license server starts automatically upon reboot.

Log files are written to the following directory when the installation is complete:

```
flex_install_dir/InstallData/log/CODE/os/media_name/
```

The log files are helpful for troubleshooting problems, and they should not be removed.

Stopping and removing an existing FLEXnet license server

If you have an existing FLEXnet license server from a previous release of Abaqus running on your system, you will need to remove it before installing the new license server. The licensing installer will attempt to remove the existing server automatically. If it is unable to do so, you should remove the existing server using one of the following procedures:

Linux platforms

1. Enter the command

```
flex_install_dir/os/code/bin/lmdown -c [port]@license_server_host  
-vendor ABAQUSLM
```

where:

- *os* is either **win_b64** or **linux_a64**
- *license_server_host* is the hostname of the FLEXnet license server. If the license server is using a port that is not between 27000 and 27009, you must specify the port number as well.

2. Verify that the server has been stopped by running the command

```
flex_install_dir/os/code/bin/lmstat -c [port]@license_server_host
```

Verify that no **ABAQUSLM** vendor daemons are reported. An error status message may appear if the SIMULIA FLEXnet license server was successfully stopped and there are no other vendor daemons running on the specified host.

Windows platforms

In addition to terminating the FLEXnet license server, you must remove licensing as a Windows service.

1. From the **Start** menu, select **FLEXNet Licensing Utilities** to display the **LMTOOLS** dialog box.
2. Verify **Configuration using Services** is enabled on the **Service/License File** tabbed page.
3. Select the **Start/Stop/Reread** tab, select the SIMULIA FLEXnet license server and click **Stop Server**.
4. Select the **Config Services** tab, select the SIMULIA FLEXnet license server, and click **Remove Service**.
5. Select **File**→**Exit** to close the **LMTOOLS** dialog box.

2.1.6 Abaqus product installation

To install the Abaqus products, you must have a network license server that is compatible with Abaqus 2016 installed and running on your network. It is also recommended that you perform the documentation installation before installing Abaqus products to obtain the URL for the HTML documentation (or the base installation directory for PDF files) and to enable access to the documentation from Abaqus. You will be asked to provide the following information relating to the documentation and licensing installations:

- The hostname of your license server(s) (REQUIRED).
- One of the following (depending on which formats of documentation were installed):
 - The URL for the Abaqus HTML documentation (optional, but recommended). If you did not install the Abaqus HTML documentation before the products or if you do not specify the URL, you must perform additional steps to enable access to the HTML documentation from the Abaqus products (see “Setting the documentation URL after installation,” Section B.2).
 - The path to the base installation directory under which the PDF documentation files are installed (if only the PDF was installed or if you prefer PDF over HTML). Entering this path when the installer prompts you will configure the **abaqus doc** command to open PDF rather than HTML documentation. See “Abaqus documentation,” Section 3.2.12 of the Abaqus Analysis User’s Guide, for more information.

INSTALLING Abaqus

A complete Abaqus installation includes several components:

1. Abaqus services (Abaqus/Standard solver, Abaqus/Explicit solver, Abaqus/CFD solver, SIMULIA Co-Simulation Engine, and APIs foundation)
2. CAA developer software for Abaqus APIs
3. Abaqus/CAE (interactive GUI) and command-line interface to Abaqus solvers

It is strongly recommended that all components be installed.

The commands used to install the Abaqus components are described below. For a more complete discussion of the Abaqus product installation options, see “Abaqus product installation details,” Section 2.4.

To install the Abaqus services (solvers):

1. Execute one of the following commands:

Linux platforms

```
unload_dir/AM_SIM_Abaqus.AllOS/1/3DEXPERIENCE_AbaqusSolver/  
Linux64/1/StartGUI.sh
```

where *unload_dir* is the directory in which you extracted (uncompressed) the media archive files.

Windows platforms

```
unload_dir\AM_SIM_Abaqus.AllOS\1\3DEXPERIENCE_AbaqusSolver\  
Windows64\1\setup.exe
```

where *unload_dir* is the directory in which you extracted (uncompressed) the media archive files.

2. Choose an empty directory/folder in which to install the software. This directory must be completely empty with no files or subdirectories.
3. Choose the components to install:

```
Abaqus/Explicit Solver  
Abaqus/CFD (Flow) Solver  
Abaqus/Standard Solver  
Cosimulation Services  
ODB API Services
```

It is strongly recommended that all components be installed.

4. Follow the instructions in the dialog boxes that appear to complete the installation.

To install the Abaqus CAA APIs:

1. Execute one of the following commands:

Linux platforms

```
unload_dir/AM_SIM_Abaqus.AllOS/1/CAA_3DEXPERIENCE_AbaqusSolver/
Linux64/1/StartGUI.sh
```

Windows platforms

```
unload_dir\AM_SIM_Abaqus.AllOS\1\CAA_3DEXPERIENCE_AbaqusSolver\
Windows64\1\setup.exe
```

2. When you are prompted for the directory to install into, enter the directory in which you installed the Abaqus services (solvers). You can only install CAA APIs for the software products you have already installed.
3. Follow the instructions in the dialog boxes that appear to complete the installation.

To install Abaqus/CAE:

Important: Abaqus/CAE *must* be installed to complete the Abaqus solvers installation.

1. Execute one of the following commands:

Linux platforms

```
unload_dir/AM_SIM_Abaqus.AllOS/1/SIMULIA_Abaqus_CAE/
Linux64/1/StartGUI.sh
```

Windows platforms

```
unload_dir\AM_SIM_Abaqus.AllOS\1\SIMULIA_Abaqus_CAE\
Windows64\1\setup.exe
```

2. Choose an empty directory/folder in which to install the software. This directory must be completely empty with no files or subdirectories.
3. Follow the instructions in the dialog boxes that appear to complete the installation.
You will need to enter the path to the location where you installed the Abaqus services (solvers). This allows the two components to connect with one another.

Log files are written to the following directory when each installer finishes:

```
install_dir/InstallData/log/CODE/os/media_name/
```

The log files are helpful for troubleshooting problems, and they should not be removed.

2.2 Abaqus documentation installation details

The Abaqus online documentation is provided in two formats, HTML and PDF. Most installations share the HTML and PDF documentation from a single server. Both formats of the documentation can also be installed on as many computers as desired, for local access by individual users.

The commands and options used to install Abaqus documentation are listed in “Documentation installation,” Section 2.1.3.

Abaqus HTML documentation can be installed on platforms that are supported for Abaqus 2016 and should generally be installed on only one computer on your network. The server machine that you choose for the HTML and PDF documentation should be accessible by all computers where you plan to run Abaqus.


The Abaqus PDF documentation is not meant as a replacement for the HTML-format documentation. However, it has the following advantages:

- Requires less disk space.
- Convenient portable reference.
- Ability to print a range of pages, such as a section or chapter.
- Higher quality printed output than that available from the HTML documentation.
- Searchable alternative to the HTML documentation, particularly for situations where the HTML documentation is not searchable because it was installed with no web server.

Installing only the PDF documentation has the following disadvantages:

- The detailed, step-by-step instructions for using each of the Abaqus/CAE functions are available only in the HTML version of the Abaqus/CAE User’s Guide and as context-sensitive help. They do not appear in the PDF version of the guide.
- The hyperlinks in context-sensitive help will not function if you have not installed the Abaqus HTML documentation.
- Some of the items under the **Help** menu in Abaqus/CAE and Abaqus/Viewer will connect only to the HTML help, not to the PDF documentation.

The Abaqus PDF collection contains all the books in the HTML collection. The PDF version of all guides except the Abaqus Scripting Reference Guide and the Abaqus GUI Toolkit Reference Guide

includes a  icon in the top left corner of each page that allows you to print a selected section without specifying a page range. More information about navigating and printing the Abaqus PDF documentation is available in Chapter 5, “Overview of the Abaqus PDF documentation,” of Using Abaqus Online Documentation.

2.2.1 Configuration of documentation installation

When the Abaqus documentation installer runs, it determines the configuration of any existing Abaqus documentation:

- If you have the documentation from an older release installed (Abaqus 6.14, for example), the new Abaqus 2016 documentation is installed in the same location as the older versions.
The Abaqus installer creates a new subdirectory named **v2016** to hold the Abaqus 2016 documentation. If you are using the Abaqus web server software, you are prompted to choose whether or not to serve all releases of the documentation simultaneously.
- If documentation from a previous release of Abaqus is not installed, you are prompted for the installation directory. The default location is the current working directory on Linux platforms and **C:\SIMULIA** on Windows platforms. You should *not* install Abaqus documentation in a directory that indicates a specific Abaqus release number. The installation procedure creates a subdirectory named **Documentation** under the specified directory.

During an HTML (plus PDF) installation, you must select one of the following options:

- Install and start the Abaqus web server software (recommended option, to allow search functionality).
- Install the HTML (plus PDF) documentation and use an existing non-Abaqus web server. In this case you must configure your web server software manually to access the Abaqus HTML documentation (see “Configuring an existing non-Abaqus web server,” Section B.1.1).
- Install the HTML (plus PDF) documentation with no web server. In this case no search functionality is available in the HTML documentation. Select this option if you want to view the HTML documentation from your local system without executing an active web server.

With the exception of “no web server,” the installation option that you select applies to both the Abaqus 2016 HTML documentation and, if it exists, any previous release of Abaqus documentation. If you select the “no web server” option for the Abaqus 2016 installation, the Abaqus 2016 HTML documentation has no web server and the documentation from previous releases is not affected.

Note: On Windows platforms the Abaqus web server software will start up automatically after a reboot. On Linux platforms the Abaqus web server is started by the installation procedure, but you should include the script to restart the web server (*doc_install_dir/Documentation/installation_info/v2016/startServer*) in the system startup file for the server host machine to have the Abaqus web server restart automatically after a system reboot. Contact your system administrator or platform vendor for information on the system startup file for your computer.

When the installation is complete, log files are written to the following directory:

doc_install_dir/Documentation/installation_info/v2016/

The log files are helpful for troubleshooting problems, and they should not be removed. For more information on accessing and using the Abaqus HTML documentation, see “Abaqus documentation,” Section 3.2.12 of the Abaqus Analysis User’s Guide, and the online book Using Abaqus Online Documentation.

2.2.2 Repeating documentation installations on multiple computers

When you install documentation, a file called **installer.properties** is created in the *doc_install_dir/Documentation/installation_info/v2016/* directory. This file, called the replay file, contains information that was used during the documentation installation, such as the installation directory and web server configuration. The **-replay** command option is available to repeat a previous installation on identical platforms without reentering the same information for each installation. You can use this option to perform a silent installation (no graphical user interface is displayed). It performs an installation identical to the previous installation (same installation directory, same configuration, etc.) using information stored in the replay file.

The information that was entered for the first installation must be valid for subsequent installations that use the **-replay** option. All installation error checking is skipped when this option is used. You should make sure that you have write permissions and sufficient disk space in the installation directory before performing the installation. To use the **-replay** option, do the following:

1. Install documentation on a computer using the graphical user interface (without the **-replay** option). You can install either option: HTML and PDF, or PDF-only.
2. Copy the file **installer.properties** (the replay file) to the other computers on which you want to install Abaqus documentation (must be the same platform) or to a mounted network drive.
3. Install the documentation on another computer using the **-replay** option as follows:

Linux platforms

```
extract_dir/AM_DOC_SIM_EstPrd.AllOS/1/  
DOC_SIMULIA_Abaqus_fe-safe/1/Abaqus_2016/setup  
-replay full_path_to_replay_file
```

Windows platforms

```
extract_dir\AM_DOC_SIM_EstPrd.AllOS\1\  
DOC_SIMULIA_Abaqus_fe-safe\1\Abaqus_2016\setup.exe  
-replay full_path_to_replay_file
```

The documentation is installed silently.

4. Review the installation log files for errors:
 - **html_InstallLog.log** and **info.log** (for an installation of HTML and PDF)
 - **pdf_InstallLog.log** (for a PDF-only installation)

These files are located in the following directory:

doc_install_dir/Documentation/installation_info/v2016/

This step is optional but recommended.

5. Repeat the documentation installation using the **-replay** option on each of the remaining computers.

2.3 Abaqus FLEXnet licensing installation details

Abaqus 2016 uses FLEXnet Licensing Version 11.6.1. The licensing installation procedure can be used for the following scenarios:

- to install the FLEXnet Licensing administration tools and to configure and start the FLEXnet Abaqus 2016 license server automatically; or
- to install only the FLEXnet Licensing files.

An existing FLEXnet license server from a previous version of FLEXnet Licensing is upgraded automatically during the installation procedure if it was previously installed and configured using the Abaqus licensing installer. If the installer detects an existing license server on your machine from a version of FLEXnet Licensing higher than Version 11.6.1, you have the option of either overwriting the existing server or terminating the installation procedure. To update your FLEXnet license file without overwriting the existing license server, see the procedure in “Updating an Abaqus FLEXnet license file,” Section 3.2.

The commands and options used to install the SIMULIA FLEXnet license server or to stop and remove an existing license server are listed in “FLEXnet Licensing installation,” Section 2.1.5.

For redundant license server configurations, you can install the license administration tools and configure the license server, but you will need to start the license servers manually (see “Redundant FLEXnet license server configurations,” Section 3.4). If you choose to install only the license administration tools, you must start the license server manually using the FLEXnet license server manager **lmgrd** (see “FLEXnet license server manager lmgrd,” Section 3.5).

If you are installing an Abaqus 2016 FLEXnet license server, you must have an Abaqus 2016 license file saved on the computer on which you are installing the license server. The FLEXnet license file is the file containing the Abaqus 2016 license that was attached to the e-mail message sent to you.

Abaqus FLEXnet licensing needs to be installed on only one computer on your network unless you are using a redundant license server configuration. The disk space requirement for the licensing utilities ranges from 10–25 MB depending upon the platform. The computer that you choose for licensing should be accessible by all computers where you plan to run Abaqus and should be a stable machine that is not frequently rebooted or shut down. For additional information on licensing, see Chapter 3, “Abaqus licensing.”

You will be prompted for the following information:

- The location of the FLEXnet license file. The installation procedure will save the license file in the *flex_install_dir/os/code/bin/* directory using the default name **simulialm.lic**.
- The installation directory (*flex_install_dir*). The default location for the installation directory is the current working directory on Linux platforms and **C:\SIMULIA\License** on Windows platforms.

Note: On Windows platforms the FLEXnet license server is started by the installation procedure and will start up automatically upon reboot. On Linux platforms the license server is started by the installation procedure, but you should include the command to restart the license server in the system startup file for the host computer to have the Abaqus license server restart automatically after a system reboot. The command to restart the license server is written to the **licenseStartup.sh** file in the *flex_install_dir/os/code/bin/* directory.

Log files are written to the following directory when the installation is complete:

flex_install_dir/InstallData/log/CODE/os/media_name/

The log files are helpful for troubleshooting problems, and they should not be removed.

After the Abaqus documentation is installed and the license server is installed and running, you should proceed to the Abaqus product installation on each computer that you will use to run the products. You need not install the Abaqus products on the license server host unless you will run Abaqus on that computer.

2.4 Abaqus product installation details

It is recommended that you install Abaqus documentation before you install the Abaqus products; if you do not, you must perform additional steps to provide access to the HTML documentation from Abaqus.

To install the Abaqus products, an Abaqus network license server that is compatible with Abaqus 2016 must be installed and running; you must know the name of at least one active Abaqus license server to perform the product installation.

The commands and options used to install the Abaqus products are listed in “Abaqus product installation,” Section 2.1.6.

You must have write permission in the installation directory to install the Abaqus products. On Windows platforms Abaqus shortcuts and required environment variables, including **PATH**, are added to the **All User** profile.

2.4.1 Visual C++ and MPI libraries

When you install Abaqus on Windows, the Microsoft Visual C++ 2012 and 2010 runtime libraries are installed automatically. These libraries are required to run Abaqus. The following libraries are installed:

- 64-bit 2012 libraries
- 64-bit 2010 libraries

Message Passing Interface (MPI) components must be installed to 1) use MPI-based parallel job execution in Abaqus/Standard, 2) to use domain-level parallelization in Abaqus/Explicit, or 3) to run any job in Abaqus/CFD (regardless of the number of CPUs). If your Abaqus users will be running these types of simulations, you must have the required MPI components preinstalled or allow the Abaqus installer to install them for you:

- On Windows/x86-64, the Abaqus services installer automatically installs the Microsoft MPI 5.0 libraries, if necessary. If you have a 64-bit Windows HPC Server, however, this is not needed because the MPI components are included with the operating system.
- On Linux 64-bit (lnx86-64) operating systems, the Abaqus product installer installs the IBM Platform Computing MPI libraries.

For more information on MPI and thread-based parallel processing modes, see “Parallel execution,” Section 3.5 of the Abaqus Analysis User’s Guide.

Any other third-party software that will be used in conjunction with the Abaqus products (compilers, operating system patches, etc.) should be installed before installing the Abaqus products. See the Program Directory for more details.

2.4.2 Information to enter during Abaqus/CAE installation

When the Abaqus/CAE installer runs, you will be prompted for the items below.

- The installation directory (*cae_install_dir*). For example: **C:\ds**. This should be an empty directory.
- The directory in which you installed the Abaqus services (solvers).
- The license server host name. If a set of redundant servers will be used, enter the host names of the redundant servers in the boxes provided in the dialog box (see “Redundant FLEXnet license server configurations,” Section 3.4, for details on installing redundant license servers). The installation will define settings in the Abaqus environment file according to information detected in the Abaqus license file on the server. If the license server contains an academic teaching license, Abaqus will be configured to use this license by default (see “License management parameters,” Section 4.1.7).
- The URL for the Abaqus HTML documentation or path to the base installation directory under which the PDF documentation files are installed (if only PDF was installed or if you prefer PDF over HTML). Entering this path when the installer prompts you will configure the **abaqus doc** command to open PDF rather than HTML documentation. See “Abaqus documentation,” Section 3.2.12 of the Abaqus Analysis User’s Guide, for more information.

If you did not install the Abaqus HTML documentation before the products or if you do not specify the URL, you must perform additional steps to provide access to the HTML documentation (see “Setting the documentation URL after installation,” Section B.2).

- The location of the Abaqus start-in directory (Windows platforms only). Specify the location where user files created during Abaqus interactive sessions are stored and the home directory for the Abaqus command window. The default location is **C:\temp**. (This can be changed by users later; see “Common customizations on Windows platforms,” Section 5.1.2.) Ensure that the start-in directory has write permission for all users.
- If the installation program detects a wide area network (WAN) license file, you will be prompted for information about the geographic location of your computer. This information is used to automatically set the **computer_location** environment file parameter and enable license usage logging for report generation. For more information, see “Reporting FLEXnet license usage data,” Section 3.8.

2.5 Uninstalling Abaqus products or licensing

This section describes how to remove Abaqus FLEXnet licensing or product releases from your computer.

For instructions on how to remove the Dassault Systèmes license server, see “Uninstalling the Dassault Systèmes License Server” in the *Dassault Systèmes License Server Installation and Configuration Guide (DSLS.pdf)*.

2.5.1 Linux platforms

The following procedures explain how to remove Abaqus product releases or FLEXnet licensing from Linux platforms.

To remove Abaqus FLEXnet licensing:

This procedure must be performed on the license server host, and you must be logged in as root or the owner of the license server.

1. Before you can remove the license server and license files, you must terminate the license server using the FLEXnet Licensing utility **lmdown** (see “lmdown,” Section 3.6.2).
2. Create a backup of any files you wish to keep, such as your Abaqus license file (typically named **abaquslm.lic** or **simulialm.lic**) and the FLEXnet Licensing server debug log file (typically named **simulialm.log**).
3. Remove the files using the following procedure:

```
cd flex_install_dir
rm -rf License
```

4. Remove the license server restart command from the computer’s system startup file. Contact your system administrator or platform vendor for information on the system startup file for your computer.

To remove the Abaqus products:

You can remove a release of Abaqus by deleting the installation directories and all subordinate files. Your user account must have permission to delete the Abaqus directories. You must delete the installation directories for both the Abaqus solvers and Abaqus/CAE.

1. Type the commands

```
cd solvers_install_dir
cd ../..
```

2. Change the permissions of the directories and files by entering the command

```
chmod -R 755 SimulationComputeServices
```

3. Remove the installed files by typing the command

```
rm -rf SimulationComputeServices
```

2.5.2 Windows platforms

You can use the **Programs and Features** control panel or the silent uninstaller batch files to remove Abaqus products or FLEXnet licensing. The silent uninstallers allow you to automate these tasks from another batch/script file. The silent uninstallers are available only on Windows platforms. You must have Windows Administrator privileges to uninstall Abaqus components.

The uninstallers may be unable to remove some files; for example, files that are open in another application. In addition, the uninstallers remove only directories and files that were written by the Abaqus installers. You may want to delete any remaining directories and files manually.

To remove Abaqus FLEXnet licensing:

FLEXnet licensing can be removed in either of the following ways:

- Use the **Programs and Features** control panel to **Uninstall a program**.
- To use the FLEXnet silent uninstaller, execute the following batch file from the command line or from another batch/script file:

```
flex_install_dir\Uninstall.bat
```

You should create a backup of any files you wish to keep, such as your Abaqus license file (typically named **abaquslm.lic** or **simuliaalm.lic**) and the FLEXnet Licensing server debug log file (typically named **simuliaalm.log**).

To remove the Abaqus products:

Abaqus products can be removed in either of the following ways:

- Use the **Programs and Features** control panel to **Uninstall a program**.

You should uninstall all of the Dassault Systèmes Abaqus items listed:

```
Dassault Systemes Abaqus CAE Abaqus 2016
Dassault Systemes Simulation Services 3DEXPERIENCE R2016x
Dassault Systemes Software Prerequisites x86-64
Dassault Systemes Software VC10 Prerequisites x86-64
Dassault Systemes Software VC11 Prerequisites x86-64
```

- To use the Abaqus product silent uninstallers, execute the following batch files from the command line or from another batch/script file:

```
solvers_install_dir\Uninstall.bat
cae_install_dir\Uninstall.bat
```

2.6 Uninstalling Abaqus documentation

This section describes how to remove Abaqus documentation from your computer.

2.6.1 Linux platforms

The following procedures explain how to remove Abaqus documentation from Linux platforms.

To remove an installation of HTML and PDF documentation:

The Abaqus HTML and PDF documentation can be removed safely by using one of the following procedures to stop the web server and delete all of the installed documentation releases or a release-specific directory and all subordinate files. You must have administrator privileges to uninstall Abaqus components.

- To remove all installed releases of the Abaqus documentation:
 1. Set your current directory to the parent directory for the Abaqus documentation (*doc_install_dir*).
 2. If you use the monitor process to serve the HTML documentation (i.e., the Abaqus web server) or use the monitor process to perform searches (i.e., an existing non-Abaqus web server), stop the monitor process using the command

```
Documentation/bin/monitor -k
```

3. Remove the documentation directory by typing the command

```
rm -rf Documentation
```

4. Remove the web server restart command from the computer's system startup file. Contact your system administrator or platform vendor for information on the system startup file for your computer.

5. Additional steps may be necessary if you are using an existing non-Abaqus web server. For more information, consult the Dassault Systèmes Knowledge Base at www.3ds.com/support/knowledge-base.
- To remove a specific release of the Abaqus documentation:
 1. Set your current directory to the `doc_install_dir/Documentation` directory.
 2. If the release that you are removing uses the monitor process to serve the HTML documentation (i.e., the Abaqus web server) or uses the monitor process to perform searches (i.e., an existing non-Abaqus web server), stop the monitor process using the command

```
bin/monitor -k
```

3. Type the following commands

```
rm -rf installation_info/release  
rm -rf taxis/release  
rm -rf docs/release
```

where *release* refers to the release of the Abaqus documentation you wish to delete.

After you remove a release of the documentation, you can configure the web server to serve the remaining HTML collection using the following procedure:

1. Set your current directory to the `doc_install_dir/Documentation` directory.
2. Verify that the release number of the release that you wish to serve is referenced in the `taxis.cnf` file as follows:

```
Default Database = %INSTALLDIR%/taxis/release
```

where *release* refers to the documentation release you wish to serve.

3. Run the command

```
installation_info/release/initializeExecutable
```

4. To start the web server to serve the remaining documentation collection, run the command

```
installation_info/release/startServer
```

To remove a PDF-only documentation installation:

1. Set your current directory to the parent directory for the Abaqus documentation (`doc_install_dir`).
2. Remove the documentation directory by typing the command

```
rm -rf Documentation
```

2.6.2 Windows platforms

You can use the **Programs and Features** control panel or the silent uninstaller batch files to remove Abaqus documentation. The silent uninstallers allow you to automate these tasks from another batch/script file. The silent uninstallers are available only on Windows platforms. You must have Windows Administrator privileges to uninstall Abaqus components.

Both the Windows uninstall shortcuts and the silent uninstaller batch files generate a log file that you can review. The log file indicates whether the uninstall was successful and if you should reboot/restart your computer. This information is shown in the last two lines of the log file. For example,

```
Uninstall Status: SUCCESS
Restart Needed: YES_RECOMMENDED
```

The **Uninstall Status** line will indicate either **SUCCESS** or **INCOMPLETE**. The **Restart Needed** line will indicate either **YES_RECOMMENDED**, **YES_REQUIRED**, or **NO**.

The uninstallers may be unable to remove some files; for example, files that are open in another application. In addition, the uninstallers remove only directories and files that were written by the Abaqus installer. You may want to delete any remaining directories and files manually.

To remove an installation of HTML and PDF documentation:

The Abaqus HTML and PDF documentation can be removed in any of the following ways:

- From the **Start** menu, select **Programs**→**Abaqus release Documentation**→**Uninstall Documentation**, where *release* refers to the release of the Abaqus documentation you wish to delete.
- Use the **Programs and Features** control panel to **Uninstall a program**.
- To use the Abaqus documentation silent uninstaller, execute the following batch file from the command line or from another batch/script file:

```
doc_install_dir\Documentation\installation_info\release\  
html_uninstaller\silent_doc_uninstall.bat
```

The documentation uninstallers generate the following log file:

```
doc_install_dir\Documentation\installation_info\release\  
uninstall_log.log
```

Other releases of the Abaqus documentation on your system are not affected by this procedure.

To remove a PDF-only documentation installation:

The Abaqus PDF documentation can be removed in any of the following ways:

- From the **Start** menu, select **Programs**→**Abaqus release PDF Documentation**→**Uninstall Documentation**, where *release* refers to the release of the documentation you wish to delete.

- Use the **Programs and Features** control panel to **Uninstall a program**.
- To use the Abaqus documentation silent uninstaller, execute the following batch file from the command line or from another batch/script file:

```
doc_install_dir\Documentation\installation_info\release\  
html_uninstaller\silent_doc_uninstall.bat
```

The documentation uninstallers generate the following log file:

```
doc_install_dir\Documentation\installation_info\release\  
uninstall_log.log
```

Other releases of the Abaqus documentation on your system are not affected by this procedure.

3. Abaqus licensing

This chapter describes network licensing for Abaqus, including the following topics:

- Updating an Abaqus 2016 FLEXnet license file.
- Upgrading a FLEXnet license server and tools.
- Redundant FLEXnet license server configurations.
- The FLEXnet license server manager, licensing administration tools, and options file.
- Using the **ds1sstat** and **reporttool** utilities for a Dassault Systèmes license server (DSLS).

3.1 FLEXnet network licensing

Abaqus 2016 FLEXnet licensing is provided in a network licensing format. Network licensing allows a limited number of analysis jobs and interactive sessions to be run simultaneously on any supported computer connected over a network. Abaqus network licensing uses the FLEXnet network license manager from Flexera Software (formerly Acreoso Software) to control an Abaqus license server, which is a process running on a single computer (license server host) on a network. Abaqus products can run on any supported computer on the network, including the license server host, as long as the necessary tokens are available.

Note: A network card or on-board network chip is required for a computer to operate as a license server.

Multiple FLEXnet license servers can be selected to provide redundancy in case the license server or its host fails. For information on redundant license servers, see “Redundant FLEXnet license server configurations,” Section 3.4.

Any supported computer on the network can be used to run the FLEXnet license server as long as the computers that will run Abaqus can communicate with the license server. A computer running Windows can act as the license server host for a Linux network and vice versa.

Each Abaqus analysis or interactive session must contact the license server prior to beginning execution and periodically while Abaqus is running. Therefore, Abaqus must have uninterrupted communication with the license server throughout the analysis or interactive session. If the license server or the computer on which it runs fails, it should be restarted as quickly as possible to ensure minimal interruption of Abaqus jobs.

License activity is recorded in a log file, called the server log. Tools to manage licensing activities are described in “FLEXnet Licensing administration tools,” Section 3.6, and The FLEXnet Licensing End User Guide Version 11.6; the latter document is available for download from the **Licensing** section of the **Support** page at www.3ds.com/simulia.

Restrictions can be applied to tokens to control access to Abaqus. A number of tokens can be reserved for certain users, or access can be denied to specific users and computers. These restrictions are defined in the network license options file. See “Using the FLEXnet options file,” Section 3.7,

for more information. You can also customize the behavior of a license server through settings in the Abaqus environment file. See “License management parameters,” Section 4.1.7, for information on the environment file settings.

The FLEXnet license server continuously tracks which tokens on the network are “available” and which are “in use.” Prior to beginning an analysis or interactive session, the Abaqus job (client) requests tokens from the license server. If tokens are in the available pool, the license server transfers tokens from the available pool to the in-use pool, and the tokens remain in the in-use pool until the end of the analysis or interactive session.

If no tokens are available, an analysis job or interactive session can be placed in the license queue to wait for a token, depending on the settings of the **lmlicensequeueing** and **lminteractivequeueing** parameters; see “License management parameters,” Section 4.1.7. If no tokens are available and the **lmlicensequeueing** and **lminteractivequeueing** parameters are set to **OFF**, the analysis job or interactive session exits with an error message. At the normal completion of the analysis or interactive session, the Abaqus client notifies the license server and the server returns the tokens to the available pool. If the analysis or interactive session aborts, the tokens are returned immediately.

For help in resolving questions relating to FLEXnet licensing, see Appendix E, “Troubleshooting Abaqus FLEXnet licensing.”

3.2 Updating an Abaqus FLEXnet license file

You may need to update an Abaqus 2016 FLEXnet license file; for example, when you obtain a new license file to replace an expired license file or to revise license features. You may also need to replace a license file from a previous release of Abaqus. For example, if you manually upgraded an existing license server to a version of FLEXnet Licensing higher than Version 11.6.1, you need to update the license file for Abaqus 2016 without reinstalling and overwriting the existing licensing utilities.

You must have write permissions to update the license file. If your license file is independent of other products using FLEXnet Licensing, you can replace the current license file with the new one using two methods: restarting the license server or forcing the server to reread the license file.

To update an Abaqus FLEXnet license file and restart the license server:

1. Backup the current license file by saving it with a new name.
2. Copy the new license file to the same location as the original license file using the name of the original license file.
3. Run the FLEXnet Licensing tool **lmdown** (see “lmdown,” Section 3.6.2) to shut down the currently running Abaqus license server.
4. Restart the Abaqus license server. For Windows platforms use the method described in “Starting the FLEXnet server using **LMTOOLS**,” Section 3.6.9. For Linux platforms use the method described in “Abaqus FLEXnet licensing installation details,” Section 2.3.

To update and reread an Abaqus FLEXnet license file:

1. Backup the current license file by saving it with a new name.
2. Copy the new license file to the same location as the original license file using the name of the original license file.
3. Run the FLEXnet Licensing tool **lmreread** (see “lmreread,” Section 3.6.6) to force the license server to reread the new license file.

The license file that you update must maintain its file name and path name (location). If either the file name or path name change, you should uninstall licensing and then run the licensing installation procedure (see Chapter 2, “Installing Abaqus”). The default name of the Abaqus license file is **simulia.lic**. You can determine the name of the Abaqus license file by running the command

```
flex_install_dir/os/code/bin/lmstat -c [port]@license_server_host
```

where *license_server_host* is the hostname of the license server. If the license server is using a port that is not between 27000 and 27009, you must specify the port number as well.

If your Abaqus license features are integrated with other products using a combined FLEXnet license file, replace the Abaqus licensing information in the combined license file with the updated information. You must run the FLEXnet Licensing tool **lmreread** (see “lmreread,” Section 3.6.6) to force the license server to reread the new license file. For more information on combined license files, see The FLEXnet Licensing End User Guide Version 11.6, which is available for download from the **Licensing** section of the **Support** page at www.3ds.com/simulia.

3.3 Upgrading a FLEXnet license server and administration tools

Abaqus 2016 licensing requires FLEXnet Licensing Version 11.6.1; the license server must be upgraded if it is an earlier version. You can use the FLEXnet Licensing tool **lmver** or **lmstat** (see “lmstat,” Section 3.6.7) to check the version number of the license server.

If you need to upgrade the license server, you should use the licensing installation procedure to upgrade the license server automatically. If you want to configure the license server manually, you can use the licensing installation procedure to install only the FLEXnet Licensing administration tools. For more information, see “FLEXnet Licensing installation,” Section 2.1.5.

If your Abaqus license file is integrated with FLEXnet license files for other products using a combined license file, you must ensure that the **lmgrd** version meets the FLEXnet version compatibility rules. For more information, see the FLEXnet Licensing End User Guide Version 11.6, which is available for download from the **Licensing** section of the **Support** page at www.3ds.com/simulia.

3.4 Redundant FLEXnet license server configurations

Abaqus FLEXnet licensing offers two forms of server redundancy to guard against license disruptions: redundant license server triads and license file lists. A redundant license server triad consists of three license servers that function as one using the same license key. A license file list consists of multiple independent license servers, each with its own license key; token requests will check each license server in the list until available tokens are found. For a detailed comparison of the two redundancy configurations, see “Comparing multiple license servers to three-server redundancy” in the Dassault Systèmes Knowledge Base at www.3ds.com/support/knowledge-base. Configuration of license file lists is not covered in detail in this guide.

A redundant license server triad requires three computers. At least two license servers must be running for licenses to be granted. Network communications must be consistent and reliable among the three computers used in a redundant server triad. Redundant license server triads are considered to be an advanced use of FLEXnet Licensing. More detailed requirements for redundant license servers appear in the FLEXnet Licensing End User Guide Version 11.6, which is available for download from the **Licensing** section of the **Support** page at www.3ds.com/simulia.

You can install and configure a redundant FLEXnet license server triad using either an automatic or a manual process. The manual process is required if your servers use different port numbers; otherwise, the automatic installation is recommended. In both cases you will have to start the servers manually using the FLEXnet license server manager **lmgrd**. In their initial startup, the redundant license servers must be started simultaneously (within one minute of each other).

3.4.1 Installing a redundant FLEXnet license server triad automatically

Use the following procedure to automatically install and configure a redundant license server triad.

1. Copy the license file that you received to each of the three license server computers.
2. On one of the server computers, use the Abaqus licensing installation procedure to automatically install, configure, and start the Abaqus license server (see “FLEXnet Licensing installation,” Section 2.1.5).

Once the installation program detects a redundant license file, it will not be able to start the server. However, it will install and configure the license server automatically.

3. Follow the instructions that appear in the dialog boxes to complete the licensing installation. When prompted, enter the port number on which the servers will run. In addition, enter the hostnames or IP addresses of your servers next to the corresponding host ids.
4. Repeat Steps 2 and 3 for the remaining two server computers. The server information must be entered in the same order for each installation.

5. Start the servers using the procedure described in “Starting a redundant FLEXnet license server triad,” Section 3.4.3.

3.4.2 Installing a redundant FLEXnet license server triad manually

Use the following procedure to manually install and configure a redundant license server triad. This procedure is necessary if the three servers in your configuration use different ports.

1. On each of the three license server computers, use the Abaqus licensing installation procedure to install the Abaqus FLEXnet Licensing administration tools (see “FLEXnet Licensing installation,” Section 2.1.5).

The licensing tools are saved in the *flex_install_dir/os/code/bin/* directory.

2. Copy the license file that you received to the licensing directory on one of the server machines. Rename this license file **simulialm.lic**.
3. Edit the **simulialm.lic** file, and replace the string **this_host** with the actual hostname of your computer. Verify that the hostname corresponds to the host id provided. Use *hostname.domain_name* if you have client machines that are not local to the server network.
4. Add the network port number to the **SERVER** line. It is recommended that you do not use a port number between 27000 and 27009.
5. Repeat this process for each **SERVER** line.
6. Append the following keywords to the first **SERVER** line:

```
PRIMARY_IS_MASTER HEARTBEAT_INTERVAL=60
```

For example:

```
SERVER bifrost.simulia.com 69084992 2501 \  
    PRIMARY_IS_MASTER HEARTBEAT_INTERVAL=60  
SERVER tank.simulia.com 69094954 2501  
SERVER willow.simulia.com 69094344 2501  
VENDOR ABAQUSLM
```

where each server is configured to use port number 2501 and the numbers beginning with 690 are the host ids provided in the **simulialm.lic** file for each server host. The port number must be assigned for each server, but the number need not be the same on every server.

Note: Each line in the license file must be 80 characters or less. You can use a backslash character (\) to indicate a line continuation; keywords that appear on the line after a backslash are considered part of the previous line in the license file.

7. Save the **simulialm.lic** file, and copy the file to the licensing directory on each of the other two server computers.

8. On Windows platforms, you must install licensing as a service on each license server using the procedure described in “Installing FLEXnet licensing as a Windows service,” Section 3.6.8.
9. Start the license servers using the procedure described in “Starting a redundant FLEXnet license server triad,” Section 3.4.3.

3.4.3 Starting a redundant FLEXnet license server triad

After installing and configuring the license servers on all three computers, you must use the following procedure to start each of the servers within one minute of each other.

Linux platforms

Start the Abaqus license server by executing the following command from the *flex_install_dir/os/code/bin/* directory:

```
./lmgrd -c simulia.lic -l +simulia.log
```

Windows platforms

You must be logged in as Administrator to start the server.

1. From the **Start** menu, select **Programs→Abaqus Licensing→Licensing utilities**.
2. Verify that **Configuration using Services** is enabled on the **Service/License File** tabbed page, and select the **Start/Stop/Reread** tab.
3. Click **Start Server**.

3.5 FLEXnet license server manager lmgrd

The FLEXnet license server manager **lmgrd** handles the initial contact with the client application programs, passing the connection on to the appropriate vendor daemon. It also starts and restarts the vendor daemons.

The **lmgrd** program is the main daemon program for FLEXnet Licensing. When you invoke **lmgrd**, the program looks for a license file that contains information about vendors and features. On Linux platforms it is recommended that you run **lmgrd** as a non-privileged user (not root).

See The FLEXnet Licensing End User Guide Version 11.6 for more information on FLEXnet Licensing. This guide is available for download from the **Licensing** section of the **Support** page at www.3ds.com/simulia.

Syntax and Options

```
lmgrd [-c license_file_list] [-1 [+]debug_log_path] [-2 -p] [-local]
[-x lmdown] [-x lmremove] [-z] [-v] [-help]
```

-c *license_file_list*

Specifies the license files to use.

-l [**+**] *debug_log_path*

Writes debugging information to the file *debug_log_path*. This option uses the letter **l**, not the numeral 1. Prepending *debug_log_path* with the + character appends logging entries.

-2 -p

Restricts usage of **lmdown** (see “lmdown,” Section 3.6.2), **lmreread** (see “lmreread,” Section 3.6.6), and **lmremove** (see “lmremove,” Section 3.6.5) to a FLEXnet Licensing administrator who is by default root. If there is a Linux group called “lmadmin,” use is restricted to members of that group. If root is not a member of this group, root does not have permission to use any of the three tools listed above. If you use **-2 -p** when starting **lmgrd**, no user on Windows platforms can shut down the license server with **lmdown**.

-local

Restricts **lmdown** and **lmreread** to be run only from the same machine where **lmgrd** is running.

-x lmdown

Disables the **lmdown** command (no user can run **lmdown**). If **lmdown** is disabled, you need to stop **lmgrd** via **kill pid** (Linux platforms) or stop the **lmgrd** and vendor daemon processes through the Windows **Task Manager** or Windows service. On Linux platforms, verify that the **kill** command does not contain the **-9** option.

-x lmremove

Disables the **lmremove** command (no user can run **lmremove**).

-z

Runs in foreground. The default behavior is to run in the background. If **-l debug_log_path** is present, no windows are used. If no **-l** options are specified, separate windows are used for **lmgrd** and each vendor daemon.

-v

Prints **lmgrd** version number and copyright and exits (does not launch **lmgrd**).

-help

Displays usage information and exits.

3.6 FLEXnet Licensing administration tools

FLEXnet Licensing provides administration tools to help manage the network licensing activities. The Abaqus licensing installation procedure installs the license file and the Abaqus license server. The FLEXnet Licensing administration tools are installed in *flex_install_dir/os/code/bin/*. If you have

installed Abaqus products on the license server, you can access the **License** directory using the **abaqus** command. Running the command

```
abaqus licensing
```

without additional arguments displays a command usage summary of all available FLEXnet Licensing administration tools. The FLEXnet Licensing End User Guide Version 11.6 contains detailed information on the syntax and options of the FLEXnet Licensing administration tools. You can download this guide from the **Licensing** section of the **Support** page at www.3ds.com/simulia.

On Windows platforms the licensing tools are also available using the FLEXnet Licensing toolchest **LMTTOOLS**. The **LMTTOOLS** toolchest can be accessed from the **Start** menu or by executing one of the following commands:

```
abaqus licensing lmtools
```

or

```
flex_install_dir/os/code/bin/lmtools
```

The following sections describe the syntax and options of several commonly used FLEXnet Licensing administration tools and **LMTTOOLS** procedures:

- “lmdiag,” Section 3.6.1
- “lmdown,” Section 3.6.2
- “lmhostid,” Section 3.6.3
- “lmpath,” Section 3.6.4
- “lmremove,” Section 3.6.5
- “lmreread,” Section 3.6.6
- “lmstat,” Section 3.6.7
- “Installing FLEXnet licensing as a Windows service,” Section 3.6.8
- “Starting the FLEXnet server using **LMTTOOLS**,” Section 3.6.9

3.6.1 lmdiag

The **lmdiag** tool allows you to diagnose problems when you cannot check out a license.

Syntax and Options

```
lmdiag [-c license_file_list] [-n] [feature[:keyword=value]]
```

```
-c license_file_list
```

Diagnose the specified files.

-n

Run in non-interactive mode; **lmdiag** will not prompt for any input in this mode. In this mode extended connection diagnostics (see below) are not available.

feature

Diagnose this feature only.

keyword=value

If a license file contains multiple lines for a particular feature, you can select a particular line for **lmdiag** to report on.

For example,

lmdiag f1:HOSTID=12345678

attempts a checkout on the line with the hostid “12345678.” *keyword* can be one of the following:

- VERSION
- HOSTID
- EXPDATE
- KEY
- VENDOR_STRING
- ISSUER

If no *feature* is specified, **lmdiag** operates on all features in the license files in your list. **lmdiag** first prints information about the license, then attempts to check out each license. If the checkout succeeds, **lmdiag** indicates this. If the checkout fails, **lmdiag** gives you the reason for the failure. If the checkout fails because **lmdiag** cannot connect to the license server, you have the option of running “extended connection diagnostics.”

These extended diagnostics attempt to connect to each TCP/IP port on the license server machine and detect if the port number in the license file is incorrect. **lmdiag** indicates each TCP/IP port number that is listening. If it is an **lmgrd** process, **lmdiag** indicates this as well. If **lmdiag** finds the vendor daemon for the feature being tested, it indicates the correct port number for the license file to correct the problem.

See The FLEXnet Licensing End User Guide Version 11.6 for additional information. This guide is available for download from the **Licensing** section of the **Support** page at www.3ds.com/simulia.

To use LMTTOOLS (Windows platforms) to run **lmdiag**:

1. From the **Start** menu, select **Programs**→**Abaqus Licensing**→**Licensing utilities**.

2. Verify that **Configuration using Services** is enabled on the **Service/License File** tabbed page, and select the **Server Diags** tab.
3. Click **Perform Diagnostics**.

3.6.2 lmdown

The **lmdown** tool allows for the graceful shutdown of selected license daemons (both **lmgrd** and selected vendor daemons) on all machines.

Syntax and Options

```
lmdown -c license_file_list [-vendor vendor] [-q] [-all]
```

-c *license_file_list*

Use the specified license files. Specifying **-c license_file_list** is always recommended with **lmdown**.

-vendor *vendor*

Shut down only this vendor daemon. **lmgrd** continues running.

-q

Do not prompt or print a header. Otherwise, **lmdown** asks “Are you sure? [y/n]:.”

-all

If multiple servers are specified, automatically shut down all of them. **-q** is implied with **-all**.

You can protect the unauthorized execution of **lmdown** when you start up the license manager daemon **lmgrd**. Shutting down the servers causes users to lose their licenses. See the **-local**, **-2 -p**, and **-x** options in “FLEXnet license server manager lmgrd,” Section 3.5, for details about securing access to **lmdown**.

If **lmdown** encounters more than one server (for example, if **-c** specifies a directory with many ***.lic** files), a choice of license servers to shut down is presented.

Note: On Linux platforms, do *not* use **kill -9** to shut down the license servers. On Windows platforms, if you must use the **Task Manager** to kill the FLEXnet Licensing service, be sure to end the **lmgrd** process first, then all the vendor daemon processes.

To stop and restart a single vendor daemon, use **lmdown -vendor vendor**, then use **lmreread -vendor vendor** to restart the vendor daemon (see “lmreread,” Section 3.6.6).

When shutting down a three-server redundant license server, there is a one-minute delay before the servers shut down. **lmdown** shuts down all three license servers of a set of redundant license servers. If you need to shut down one of a set of redundant license servers (not recommended because if either of the remaining machines becomes unavailable, the license server will stop

servicing licenses), you must kill both the **lmgrd** and vendor daemon processes on that license server machine.

To use LMTOOLS (Windows platforms) to run lmdown:

1. From the **Start** menu, select **Programs→Abaqus Licensing→Licensing utilities**.
2. Verify that **Configuration using Services** is enabled on the **Service/License File** tabbed page, and select the **Start/Stop/Reread** tab.
3. Click **Stop Server**.

3.6.3 lmhostid

The **lmhostid** tool returns the FLEXnet Licensing host id of the current platform.

Syntax and Options

```
lmhostid [-n]
```

-n

Only the host id is returned as a string, which is appropriate to use with **HOSTID=** in the license file. Header text is suppressed.

Example

The following is an example of **lmhostid** output:

```
lmutil - Copyright (c) 1989-2008 Acresto Software  
The FLEXlm host ID of this machine is "69021c89"
```

To use LMTOOLS (Windows platforms) to run lmhostid:

1. From the **Start** menu, select **Programs→Abaqus Licensing→Licensing utilities**.
2. Verify that **Configuration using Services** is enabled on the **Service/License File** tabbed page, and select the **System Settings** tab.
The host id is listed under **Ethernet Address**.

3.6.4 Impath

The **Impath** tool allows direct control over FLEXnet license path settings. It is most useful for checking current license path settings for diagnostic purposes.

Syntax and Options

```
Impath {-add | -override} {vendor | all} license_file_list
```

-add

Prepends *license_file_list* to the current license file list or creates the license file list, if it does not exist, initializing it to *license_file_list*. Duplicates are discarded.

-override

Overrides the existing license file list with *license_file_list*. If *license_file_list* is the null string (""), the specified list is deleted.

- **lmpath -override all ""**
Deletes the value of **LM_LICENSE_FILE**.
- **lmpath -override vendor ""**
Deletes the value of **VENDOR_LICENSE_FILE**

vendor

A vendor daemon name. Affects the value of **VENDOR_LICENSE_FILE**.

all

Refers to all vendor daemons. Affects the value of **LM_LICENSE_FILE**.

license_file_list

A colon-separated list on Linux platforms or a semicolon-separated list on Windows platforms. If *license_file_list* is the null string (""), the specified entry is deleted.

Note: **lmpath** works by setting **\$HOME/.flexlmrc** on Linux platforms and the FLEXnet Licensing registry entry on Windows platforms.

To display the current license path settings, use the command

lmpath -status

The following information is displayed:

```
lmutil - Copyright (C) 1989-2008 Acresto Software  
Known Vendors:
```

```
demo: ./counted.lic:./uncounted.lic
```

```
Other Vendors:
```

```
/usr/local/flexlm/licenses/license.lic
```

Where the path is set to a directory, all of the *.lic files are listed separately.

To use LMTOOLS (Windows platforms) to run lmpath:

1. From the **Start** menu, select **Programs→Abaqus Licensing→Licensing utilities**.

2. Verify that **Configuration using Services** is enabled on the **Service/License File** tabbed page, and select the **Utilities** tab.
3. Click **List All Vendor Paths**.

3.6.5 lmremove

The **lmremove** tool allows you to remove a single user's license for a specified feature. If the application is active, it rechecks out the license shortly after it is freed by **lmremove**. If an Abaqus process terminates abnormally, it may not return license tokens to the license pool even though the tokens are no longer needed. In this situation **lmremove** can be used to return Abaqus/CAE and Abaqus/Viewer tokens to the license pool. The *user*, *user_host*, *display*, *server_host*, *port*, and *handle* information must be obtained from the output of **lmstat -a**. The **lmremove** tool should not be used to return Abaqus analysis job tokens to the license pool; see Appendix E, "Troubleshooting Abaqus FLEXnet licensing," for more information.

Note: The **lmremove** tool does not free licenses for use by other jobs. To temporarily free a license for use by another job, a running analysis job can be suspended using the Abaqus **suspend** utility (refer to "Job execution control," Section 3.2.43 of the Abaqus Analysis User's Guide, for details). A running analysis job can be terminated using the Abaqus **terminate** utility or the appropriate operating system utility to stop the executable for the analysis job. For an example of using the **lmremove** tool, see "lmstat," Section 3.6.7.

Syntax and Options

lmremove [-c *license_file_list*] *feature user user_host display*

or

lmremove [-c *license_file_list*] -h *feature server_host port handle*

-c *license_file_list*

Specifies the license files.

feature

Name of the feature checked out by the user.

user

Name of the user whose license you are removing, as reported by **lmstat -a**.

user_host

Name of the host the user is logged on to, as reported by **lmstat -a**.

display

Name of the display where the user is working, as reported by **lmstat -a**.

server_host

Name of the host on which the license server is running, as reported by **lmstat -a**.

port

TCP/IP port number where the license server is running, as reported by **lmstat -a**.

handle

License handle, as reported by **lmstat -a**.

The **lmremove** tool removes all instances of *user* on *server_host* and *display* from usage of *feature*. If the **-c license_file_list** option is specified, the indicated file is used as the license file. The **-h** variation uses *server_host*, *port*, and license *handle*, as reported by **lmstat -a**.

3.6.6 **lmreread**

The **lmreread** tool causes the license manager to reread the license file and start any new vendor daemons that have been added. In addition, all currently running vendor daemons are signaled to reread the license file and their end-user options files for changes in feature licensing information or option settings. If report logging is enabled, any report log data still in the vendor daemon's internal data buffer are flushed. **lmreread** recognizes changes to server machine host names but cannot be used to change server TCP/IP port numbers.

If the optional vendor daemon name is specified, only the named daemon rereads the license file and its end-user options file (in this case **lmgrd** does not reread the license file).

Syntax and Options

lmreread [**-c license_file_list**] [**-vendor vendor**] [**-all**]

-c license_file_list

Use the specified license files.

-vendor vendor

Only this vendor daemon rereads the license file. **lmgrd** restarts the vendor daemon if necessary.

-all

If more than one **lmgrd** is specified, instructs all **lmgrds** to reread.

You may want to protect the execution of **lmreread**. See the **-2 -p** and **-x** options in "FLEXnet license server manager **lmgrd**," Section 3.5, for details about securing access to **lmreread**.

To stop and restart a single vendor daemon, use **lmdown -vendor vendor**, then use **lmreread -vendor vendor**, which restarts the vendor daemon.

Note: If you use the `-c license_file_list` option, the license files specified are read by `lmreread`, not by `lmgrd`; `lmgrd` rereads the file it read originally.

To use LMTTOOLS (Windows platforms) to run `lmreread`:

1. From the **Start** menu, select **Programs→Abaqus Licensing→Licensing utilities**.
2. Verify that **Configuration using Services** is enabled on the **Service/License File** tabbed page, and select the **Start/Stop/Reread** tab.
3. Select **SIMULIA FLEXnet License Manager**, and click **ReRead License File**.

3.6.7 `lmstat`

The `lmstat` tool helps you monitor the status of all FLEXnet network licensing activities, including:

- Daemons that are running
- License files
- Users of individual features
- Users of features served by a specific vendor daemon

The `lmstat` tool prints information that it receives from the license server; therefore, it does not report on unserved licenses such as uncounted licenses. To report on an uncounted license, the license must be added to a served license file and the application must be directed to use the license server for that license file (via `@host`, `port@host`, or `USE_SERVER`). Queued users and licenses shared due to duplicate grouping are also not returned by `lmstat`.

Syntax and Options

```
lmstat [-a] [-c license_file_list] [-f [feature]] [-i [feature]] [-s [server]]
[-S [vendor]] [-t timeout_value]
```

-a

Displays all information.

-c *license_file_list*

Specifies the license files to use.

-f [*feature*]

Displays the users of *feature*. If *feature* is not specified, usage information for all features is displayed.

-i [*feature*]

Displays information from the **FEATURE/INCREMENT** line for the specified *feature* or for all features if *feature* is not specified.

-s [*server*]

Displays the status of all license files in **\$VENDOR_LICENSE_FILE** or **\$SLM_LICENSE_FILE** on *server* or on all servers if *server* is not specified.

-S [*vendor*]

Lists all users of the specified vendor's features.

-t *timeout_value*

Sets the connection timeout to *timeout_value*. This limits the amount of time **lmstat** spends attempting to connect to *server*.

Note: The **lmstat -a** command is a potentially expensive command. With many active users, this command generates a lot of network activity.

The **lmremove** tool requires the output of the **lmstat -a** command, as shown in the example below.

Example

The output for the command

```
flex_install_dir/os/code/bin/lmstat -a
```

looks similar to the following:

```
License server status: 27000@firestar
    License file(s) on firestar: flex_install_dir\win_b64\code\bin\simulia1m.lic:
    firestar: license server UP (MASTER) v11.6
Vendor daemon status (on firestar):
    ABAQUSLM: UP v11.6
Feature usage info:
Users of cae: (Total of 4 licenses issued; Total of 1 license
in use)
    "cae" v61.2, vendor: ABAQUSLM
    floating license
    smith watt watt (v61.2) (firestar/27000 101),
start Tue 3/1 9:29
```

where

smith	<i>user</i>	User name
watt	<i>user_host</i>	Host where user is running
watt	<i>display</i>	Display where user is running
v61.2	<i>release</i>	Release of feature
firestar	<i>server_host</i>	Host where license server is running

27000	<i>port</i>	Port on <i>server_host</i> where license server is running
101	<i>handle</i>	License handle
start Tue 3/1 9:29	<i>checkout_time</i>	Time that this license was checked out

Note: The **lmremove** tool does not free licenses for use by other jobs (see “lmremove,” Section 3.6.5, for more information).

To use the **lmremove** tool to free the license for a job run by user **smith**, you would use the command

```
abaqus licensing lmremove cae smith watt watt
```

where

cae	<i>feature</i>	Name of the feature checked out by the user
smith	<i>user</i>	Name of the user whose license you are removing, as reported by lmstat -a
watt	<i>user_host</i>	Name of the host the user is logged into, as reported by lmstat -a
watt	<i>display</i>	Name of the display where the user is working, as reported by lmstat -a

To use LMTTOOLS (Windows platforms) to run `lmstat -a`:

1. From the **Start** menu, select **Programs**→**Abaqus Licensing**→**Licensing utilities**.
2. Verify that **Configuration using Services** is enabled on the **Service/License File** tabbed page, and select the **Server Status** tab.
3. Verify that **Display Everything** is enabled, and click **Perform Status Enquiry**.

3.6.8 Installing FLEXnet licensing as a Windows service

You must be logged in as Administrator to install licensing as a Windows service. You use the installation procedure to install the licensing administration tools. You must use the following procedure to install and start the service:

1. To access **LMTTOOLS** from the **Start** menu, select **Programs**→**Abaqus Licensing**→**Licensing utilities**.
2. Verify that **Configuration using Services** is enabled on the **Service/License File** tabbed page, and select the **Config Services** tab.
3. In the **Service Name** text field, type **SIMULIA FLEXnet License Manager**.

- Specify the paths to the **lmgrd.exe** file, the license file, and the debug log file, as shown in Figure 3–1.

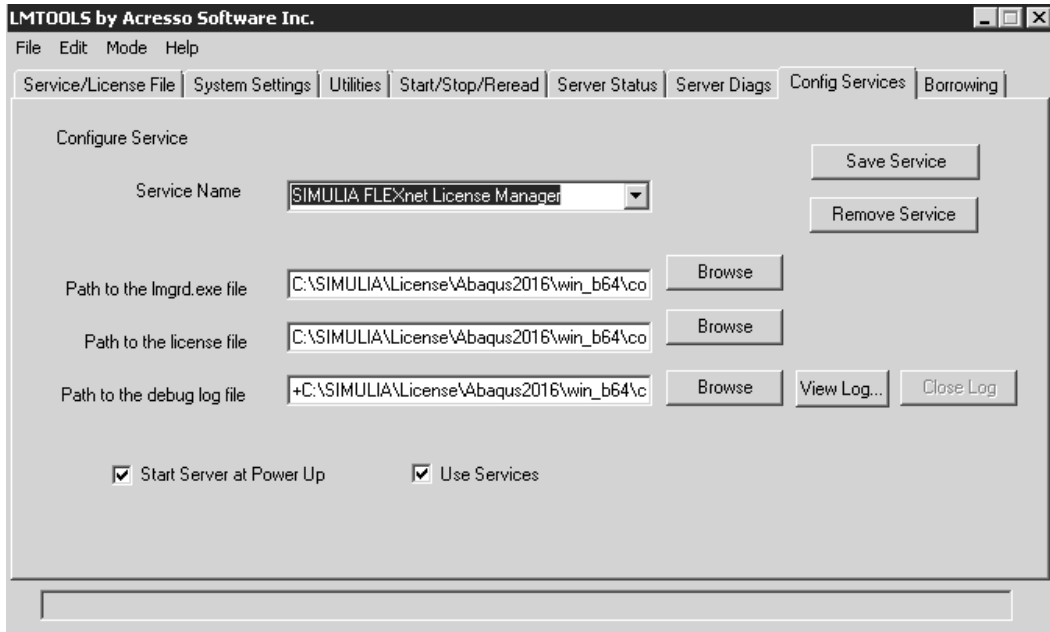


Figure 3–1 The **Config Services** tabbed page of **LMTOOLS**.

- Toggle on **Use Services**.
Start Server at Power Up becomes available.
- Toggle on **Start Server at Power Up**, and click **Save Service**.
- Select the **Start/Stop/Reread** tab, and click **Start Server**.
- From the main menu bar, select **File**→**Exit** to close the dialog box.

3.6.9 Starting the FLEXnet server using LMTOOLS

You must be logged in as Administrator to start the server.

- To access **LMTOOLS** from the **Start** menu, select **Programs**→**Abaqus Licensing**→**Licensing utilities**.
- Verify that **Configuration using Services** is enabled on the **Service/License File** tabbed page, and select the **Start/Stop/Reread** tab.

3. Click **Start Server**.
4. From the main menu bar, select **File**→**Exit** to close the dialog box.

3.7 Using the FLEXnet options file

The options file allows the license administrator to control various operating parameters of FLEXnet Licensing. Parameters, such as **EXCLUDE**, **INCLUDE**, **MAX**, and **RESERVE**, can be used to control Abaqus license usage. Usage can be controlled by user name, host name, display, IP address, or project. When specifying the features to control, you may specify features listed on a **FEATURE** line in the license file. Using a **PACKAGE** name in place of a **FEATURE** name applies the option to all of the components in the package. You should not specify individual **COMPONENTS** listed on the **PACKAGE** line, as it may lead to unexpected behavior. See Chapter 5 of the FLEXnet Licensing End User Guide Version 11.6 for detailed information on creating an options file. (This guide is available from the **Licensing** section of the **Support** page at www.3ds.com/simulia.)

The following is a sample options file:

```
INCLUDEALL INTERNET 192.168.0.*
RESERVE 5 abaqus USER robert
RESERVE 10 abaqus HOST demeter
MAX 5 abaqus USER jim
EXCLUDE abaqus USER khan
EXCLUDE viewer HOST guardian
```

In this options file:

- Connections are accepted only from machines with the IP addresses 192.168.0.*.
- Five tokens of feature **abaqus** are reserved for user *robert*.
- Ten tokens of feature **abaqus** are reserved for users on host *demeter*.
- The user *jim* is limited to five tokens of feature **abaqus** at a time.
- The user *khan* is excluded from accessing feature **abaqus**.
- The users on host *guardian* are excluded from accessing feature **viewer** (Abaqus/Viewer).

3.8 Reporting FLEXnet license usage data

This section describes methods for creating license usage reports. If your Abaqus FLEXnet license server is on a wide area network (WAN) and Abaqus users in different locations receive technical support and customer service from more than one regional SIMULIA office, you are required to submit license usage reports to SIMULIA. For more information about reporting requirements, see “License usage reporting for multi-territory accounts” in the Dassault Systèmes Knowledge Base at www.3ds.com/support/knowledge-base.

The reporting tools are available to all Abaqus licensees. Two utilities are provided with Abaqus to generate license usage reports:

- The **usage** utility (described in “Manually creating FLEXnet license usage reports,” Section 3.8.1) can be used to manually generate license usage reports for a specified range of dates.
- The scheduled license reporting utility (**schLicRpt**, described in “Scheduling automated FLEXnet license usage reporting,” Section 3.8.2) can be used to set up automated license usage reports on a monthly or quarterly basis.

The license usage reporting utilities use information from the FLEXnet debug log file. You can ensure that this log is created by including the **-l** option when starting the license server using **lmgrd** (see “FLEXnet license server manager lmgrd,” Section 3.5). To subtotal reports according to user location, you must set the **computer_location** environment file parameter. More details about the environment file parameters can be found in “License management parameters,” Section 4.1.7.

3.8.1 Manually creating FLEXnet license usage reports

The **usage** utility allows you to generate CSV-formatted reports detailing the usage of Abaqus licenses. Two types of usage reports are generated: summary usage reports and daily usage reports. Summary usage reports provide usage information for each Abaqus license feature subtotaled by user location or individual user. Summary usage reports list the total number of licenses requested, number of licenses granted, number of licenses denied, and total hours of license usage within a specified period. Daily usage reports list the peak and average number of analysis and Abaqus/CAE license tokens checked out on each day within a specified period. Daily usage reports provide aggregate data for all users and locations.

The license usage reports calculate token usage to two significant figures. If the total hours of license usage (in a summary report) or the average number of tokens checked out (in a daily report) is less than 0.005, the usage report lists these values as zero.

If the FLEXnet debug log file (or log files) is located on a remote server, you must copy it to a directory that is accessible to the local computer before running the **usage** utility.

Syntax and Options

```
abaqus licensing usage {-summary report_file | -daily report_file} [-nouser]
-log {debug_log_file | debug_log_directory} [-start start_date] [-end end_date]
```

-summary

Use this option to create a report summarizing usage of each license feature by user or location. The report is written to the path and file name specified by *report_file*.

-daily

Use this option to create a report providing peak and average daily usage of analysis and Abaqus/CAE license tokens. The report is written to the path and file name specified by *report_file*.

-nouser

This option suppresses usage data about individual users in a summary report. It reports aggregate data for the entire license server. If the **computer_location** environment file parameter is set, it also reports aggregate data for each location. This option has no effect when used in conjunction with the **-daily** option; daily reports do not include information about individual users or locations.

-log

Use this option to specify the path to the debug log file or to a directory that includes multiple debug log files; the utility reviews all log files located in the specified path for usage data within the specified reporting period.

-start

Use this option to specify the starting date and time for the reporting period. If the **-start** option is not specified, the reporting period begins with the oldest recorded item in the debug log files. The date and time must be entered in the following format:

dd-mmm-yyyy[_hh[:mm[:ss]]]

For example, **-start 01-jan-2008_09:00:00** indicates a starting time of 9:00 AM on January 1, 2008. Specifying a time is optional; the default start time is **00:00:00**. If the time is specified, the hour field (*hh*) must be entered in 24-hour format.

-end

Use this option to specify the ending date and time for the reporting period. If the **-end** option is not specified, the reporting period ends with the most recent recorded item in the debug log files. The date and time must be entered in the following format:

dd-mmm-yyyy[_hh[:mm[:ss]]]

For example, **-end 31-mar-2008_18:00:00** indicates an ending time of 6:00 PM on March 31, 2008. Specifying a time is optional; the default end time is **23:59:59**. If the time is specified, the hour field (*hh*) must be entered in 24-hour format.

3.8.2 Scheduling automated FLEXnet license usage reporting

Abaqus provides a utility that automatically generates both summary and daily FLEXnet license usage reports according to a monthly or quarterly schedule. The utility creates a script containing commands to generate the reports, then it installs a scheduled task (on Windows platforms) or a **crontab** entry (on Linux platforms) to execute the script on a monthly or quarterly basis. The debug log file can be located on a server that is remote from the computer on which automated license usage reporting is scheduled.

If desired, you can instruct the utility to automatically e-mail the reports to a specified address as soon as they are generated.

You should create a new directory dedicated to license usage reporting and run the scheduled license usage reporting utility from within this directory. The script for report generation (**usage_report.bat** or **usage_report.sh**), a utility configuration file (**schLicRpt.cfg**), and the generated reports are all saved to the directory in which you run the utility. Generated reports use the following file naming convention:

```
abaqus_usagen_siteID_date.csv
```

where *siteID* is the customer number that SIMULIA assigned to your site, and the *date* is written numerically in *yymmdd* format. The type of usage report is indicated by *n*: **1** indicates a summary usage report, and **2** indicates a daily usage report. For example, **abaqus_usage2_01ABCD_080515.csv** is the file name for a daily usage report created at site 01ABCD on May 15, 2008.

To run the utility, use the following command:

```
abaqus schLicRpt
```

The utility will prompt you for the required information, including the location of the debug log file, the frequency of report generation, and, if you choose to e-mail the reports, the fully qualified domain name of your SMTP server and the e-mail address to which reports are sent. If you run the utility again from the same directory, default settings will be provided during the configuration process based on the contents of the saved configuration file. You can, therefore, make minor modifications to the reporting configuration settings without reentering all of the required information.

You must have access to the **schTasks.exe** utility (on Windows) or **crontab** utility (on Linux) to run the scheduled license reporting utility. For best performance, it is recommended that you execute the utility on the Abaqus license server host computer, which is possible only if Abaqus products have been installed on this computer. The utility should be executed only on one computer within your site. Otherwise, you will generate duplicate usage reports and may degrade the performance of the license server when the reports are generated.

Use the following procedures to remove automatically generated reports from your system:

Linux platforms

To list current crontab entries, run the following command:

```
crontab -l
```

To edit the current crontab entries, run the following command:

```
crontab -e
```

To remove all of the current crontab entries, run the following command:

```
crontab -r
```

For more details on crontab, type **man crontab** from any prompt to view the crontab man page for your system.

Windows platforms

To list current scheduled tasks, run the following command:

```
schTasks /query
```

To remove the automated Abaqus license usage report, run the following command:

```
schTasks /delete /tn abaqus_usage_report
```

3.9 Using the `dslsstat` utility for a Dassault Systèmes license server

You can use the `dslsstat` utility to show basic status information about the Dassault Systèmes license server (DSLS); you can verify that the license server is running, is configured properly, and is serving the correct licenses. To see more detailed information about the license server, you can use the Dassault Systèmes License Administration Tool, which comes with the DSLS installation. See “Starting the License Administration Tool” in the *Dassault Systèmes License Server Installation and Configuration Guide (DSLS.pdf)*.

If you have installed Abaqus products on the network where the Dassault Systèmes license server resides, you can run the `dslsstat` utility using the `abaqus` command, as follows:

```
abaqus licensing dslsstat [-server machine:port] [-usage]
```

-server

Use this option to specify the name of the server machine and the port number on which the DSLS software is running; for example,

```
abaqus licensing dslsstat -server jupiter:4085
```

You can specify multiple servers to query by separating them with semicolons; for example,

```
abaqus licensing dslsstat -server jupiter:4085;mars:4085
```

On Linux, use single quotes to enclose the semicolon-separated list, for example:
`'jupiter:4085;mars:4085'`

If you omit the `-server` option on the command line, `dslsstat` attempts to determine which Dassault Systèmes license servers to query in two ways:

- Use any Dassault Systèmes license servers that Abaqus has been configured to use (usually configured during installation).
- Look in the system default location for the `DSLicSrv.txt` license client configuration file that describes which license servers to use. The locations of this file are described in “File Locations, Settings and Registry Entries” in the *Dassault Systèmes License Server Installation and Configuration Guide*.

If no servers are found, an error message is issued.

-usage

Use this option to show license usage details for individual users in the output. See the example below.

The results are returned in the command/terminal window; for example,

```
Server:  jupiter:4085
Name      : jupiter
Port      : 4085
Status    : Running
Type      : Standalone
Version   : 6.212.0
OS Name   : Windows 2008
OS Version : 5.2
OS Arch   : x86
Computer Id : TWK-4213101B783368E0
```

The information reported for each license server found includes:

- *Status*: **Running** or **Not Running**
- *Type*: **Standalone** or **Failover**

Information is also reported for each enrolled license feature found on the license servers; for example,

```
Licenses:
| Feature | Version | Model          | Number | InUse | Expires          | Server
| -----| -|-----| -|-----| -|-----| -----|
| QAE     | 11      | Token         | 50     | 0     | 31-Dec-2013 19:59:00 | jupiter
| QAQ     | 11      | ConcurrentUser | 1      | 0     | 31-Dec-2013 19:59:00 | jupiter
| QEX     | 11      | Token         | 50     | 0     | 31-Dec-2013 19:59:00 | jupiter
| QSD     | 11      | Named User    | 50     | 0     | 31-Dec-2013 19:59:00 | jupiter
```

The expiration date includes time in the local time zone.

If you include the **-usage** option, extra information is provided showing which users (login/username) are using particular license features; for example,

```
Licenses:
| Feature | Version | Model          | Number | InUse | Expires          | Server
| -----| -|-----| -|-----| -|-----| -----|
| QAE     | 11      | Token         | 50     | 0     | 31-Dec-2013 19:59:00 | jupiter
| QAQ     | 11      | ConcurrentUser | 1      | 0     | 31-Dec-2013 19:59:00 | jupiter
| QEX     | 11      | Token         | 50     | 0     | 31-Dec-2013 19:59:00 | jupiter
    jsmith on bigbird, granted on 18-Oct-2012 10:06:13
    tjones on kermit, granted on 18-Oct-2012 14:02:47
```

Each line of user information has the following format:

```
username on hostname, granted on date
```

For more information about the Abaqus licensing execution procedures, see “Licensing utilities,” Section 3.2.13 of the Abaqus Analysis User’s Guide.

3.10 Using the **reporttool** utility

You can use the **reporttool** utility to generate reports of Abaqus license usage history. The report tool reads data from the Dassault Systèmes license server log file or the FLEXnet debug log file and customizes the report according to your choices.

If you have installed Abaqus products on the network where the license server resides, you can run the **reporttool** utility using the **abaqus** command, as follows:

```
abaqus licensing reporttool -log logfile  
-logtype {dsls | flexnet}  
[-start start_date] [-end end_date]  
[-type {export | query}] -accessor accessors [-list_accessors]  
[-filter filter] [-list_filters] [-sort sorter] [-list_sorters]  
[-aggregator aggregators] [-list_aggregators] [-duration bucket_size]  
[-output report_file] [-format output_format] [-list_formats]  
[-custom customization_script] [-help]
```

For more information about the Abaqus licensing execution procedures, see “Licensing utilities,” Section 3.2.13 of the Abaqus Analysis User’s Guide.

In the context of this report tool, a license session is defined as a licensed job executing on an Abaqus product feature, which has a checkout time and a checkin time.

3.10.1 General options

The following command line options are available for the **reporttool** utility.

-log

Specify the path and file name of the log files to be read. The default location of a DSLS log file is **C:\ProgramData\DassaultSystemes\LicenseServer\LogFiles** on Windows or **/var/DassaultSystemes/LicenseServer/LogFiles/** on Linux. For FLEXnet, you must specify the debug log file. If the log file is located on a remote server, you must copy it to a directory that is accessible by your local computer. This option is required.

Multiple log files can be specified in two ways:

- Use the **-log** option multiple times: **-log first.log -log second.log**
- Give a comma-separated list of the file names: **-log first.log,second.log**

-logtype

Specify the type of license server being used with Abaqus: **ds1s** or **flexnet**. This option is required and is case-sensitive (must be lowercase).

-start

Specify the starting date and time for the reporting period. If the **-start** option is omitted, the reporting period begins with the oldest recorded item in the log files. The date and time must be specified in one of the following formats:

- **dd-*mmm*-yyyy_hh:mm:ss**
For example, **-start 01-jan-2012_09:00:00** indicates a starting time of 9:00 AM on January 1, 2012. Specifying a time is required, including hours, minutes, and seconds. The hour field (*hh*) must be entered in 24-hour format.
- **-xhours** or **-xdays**
Use this format to pick a time or day in the past. For example, **-start -6hours** specifies a start time of six hours ago.

-end

Specify the ending date and time for the reporting period. If the **-end** option is not specified, the reporting period ends with the most recent recorded item in the log files. The date and time must be specified in one of the following formats:

- **dd-*mmm*-yyyy_hh:mm:ss**
For example, **-end 31-mar-2012_18:00:00** indicates an ending time of 6:00 PM on March 31, 2012. Specifying a time is required, including hours, minutes, and seconds. The hour field (*hh*) must be entered in 24-hour format.
- **-xhours** or **-xdays**
Use this format to pick a time or day in the past. For example, **-end -3hours** specifies an ending time of three hours ago.

-type

Specify the type of report desired: **export** or **query**. If this option is omitted, the default is **export**. An export report provides the basic licensing session information, usually in tabular format or comma-separated values (CSV). The data can be optionally filtered or sorted. A query report lets you look at sessions in aggregate; you can operate on the raw data to calculate such things as maximum usage, peak usage, and averages. A query report divides the total time period into equal sized buckets. You choose the size of each time bucket with the **-duration** option. You can also

use an export report to save the raw data to a CSV file, import it into an Excel spreadsheet, and perform custom calculations using your own tools.

-accessor

Comma-separated list of accessors to be read from the log data. Accessors are the fields or columns of the output report. For example: **-accessor username,checkout,duration**. This option is required.

-list_accessors

Use this option to obtain a list of the available report accessors. The basic accessors are as follows:

- **duration**—duration of the license session, in seconds
- **username**—username
- **checkout**—checkout date
- **feature**—Dassault Systèmes license feature (trigram); for example, QSD
- **quantity**—number of licenses requested
- **project**—custom project names or numbers recorded from the **lmp** project environment file parameter (see “License management parameters,” Section 4.1.7)
- **session**—the internal session object that represents the licensing job; this is useful only when using the **peak** aggregator or designing your own custom aggregators

-filter

Filter the report data. The required format for this option is **-filter accessor:value**. The report output is filtered to include only data records for which *accessor=value*. You can include multiple *accessor:value* pairs in a comma-separated list; for example,

```
-filter username:tsmith,hostname:zulu
```

This example would produce a report showing license checkouts only from the user **tsmith** on the computer **zulu**. To create other filters, use the **-custom** option with a Python program.

-list_filters

Use this option to obtain a list of the available filters. The one built-in filter takes the form **-filter accessor:value**. If you create any custom filters using the **-custom** option, they will appear in this list.

3.10.2 Export options

The following command line options are available for sorting the output of the **reporttool** utility.

-sort

Sort the report data chronologically, from oldest to newest. The required format for this option is **-sort date**.

-list_sorters

Use this option to obtain a list of the available sorters. The one built-in sorter is **date**. If you create any custom sorters using the **-custom** option, they will appear in this list.

3.10.3 Query options

The following command line options are available for query reports from the **reporttool** utility.

-aggregator

Comma-separated list of aggregator functions to be applied to the accessor values in each time bucket. For example: **-aggregator max,average**. This option is required if **-type query** is used.

-list_aggregators

Use this option to obtain a list of the available aggregator functions. The basic aggregators are as follows:

- **max**—maximum value of an accessor in each time bucket
- **min**—minimum value of an accessor in each time bucket
- **sum**—sum of all accessor values in each time bucket
- **average**—average value of accessor in each time bucket
- **len**—number of items in each time bucket
- **peak**—maximum number of licenses in use; must operate on the **session** accessor

Not all aggregators will work with all assessors. For example, you can successfully calculate the maximum duration of sessions, but attempting to find the average value of usernames is meaningless.

-duration

The size of each time bucket for query reports, in minutes, hours, days, weeks, or months. If this option is omitted, the default is 24 hours. Examples: **-duration 1hour**, **-duration 2days**

3.10.4 Output options

The following command line options are available for formatting the output of the **reporttool** utility.

-output

Specify the path and file name of the file to which the report will be written. If this option is omitted or **-output -** is used, the default is to write to standard output (showing the report in your command prompt or shell window).

-format

Specify the output format to use.

-list_formats

Use this option to obtain a list of the available output formats. The basic formats are as follows:

- **table**—a plain table with columns for each accessor and/or aggregator selected
- **csv**—comma-separated values
- **json**—JavaScript Object Notation (see <http://json.org>)

3.10.5 Advanced options

-custom

Specify a Python program file containing your custom definitions.

3.10.6 Examples

Several examples are shown below to demonstrate different ways to use the **reporttool** utility.

The following example generates a simple export type of report showing checkout date, username, product feature, and quantity.

```
abaqus licensing reporttool -log today.log -logtype dsls -type export
-accessor checkout,username,feature,quantity
```

Checkout Date	Username	Feature	Quantity
2012-Dec-10 03:05:16	bbaggins	QEX	50
2012-Dec-10 03:33:50	gandalf	QAE	1
2012-Dec-10 04:08:34	fbaggins	QAE	1
2012-Dec-10 06:27:18	gollum	QEX	50

The following example generates a query report that calculates the average number of license tokens used in each 1-hour time bucket.

```
abaqus licensing reporttool -log today.log -logtype dsls -type query
-accessor quantity -aggregator average -duration 1hour
```

Bucket	Average
-----	-----

Abaqus LICENSING

```

2012-Sep-19 12:37:53 |      25
2012-Sep-19 13:37:53 |      19
2012-Sep-19 14:37:53 |       8
2012-Sep-19 15:37:53 |      18
2012-Sep-19 16:37:53 |      13
2012-Sep-19 17:37:53 |      34

```

The example below adds to the query report to calculate the total number of checkouts, the average checkout quantity, and the maximum checkout quantity for all sessions in **today.log**, grouped into 1-hour buckets.

```

abacus licensing reporttool -log today.log -logtype dsls -type query
-accessor quantity,quantity,quantity -aggregator count,average,max -duration 1hour

```

Bucket	Number of items	Average	Maximum
2012-Sep-19 12:37:53	108	25	80
2012-Sep-19 13:37:53	98	19	80
2012-Sep-19 14:37:53	46	8	66
2012-Sep-19 15:37:53	114	18	50
2012-Sep-19 16:37:53	129	13	50
2012-Sep-19 17:37:53	74	34	66
2012-Sep-20 07:37:53	49	49	66
2012-Sep-20 08:37:53	11	52	66
2012-Sep-20 09:37:53	2	50	50

The example below adds one more column to the query report to show the maximum duration of the sessions in each bucket.

```

abacus licensing reporttool -log today.log -logtype dsls -type query
-accessor quantity,quantity,quantity,duration -aggregator count,average,max,max
-duration 1hour

```

Bucket	Number of items	Average	Maximum	Maximum
2012-Sep-19 12:37:53	108	25	80	1497
2012-Sep-19 13:37:53	98	19	80	1220
2012-Sep-19 14:37:53	46	8	66	77
2012-Sep-19 15:37:53	114	18	50	482
2012-Sep-19 16:37:53	129	13	50	449
2012-Sep-19 17:37:53	74	34	66	3552
2012-Sep-20 07:37:53	49	49	66	3272

4. Customizing the Abaqus environment

This chapter describes how to use environment file parameters to customize the Abaqus execution procedure, including how to define analysis batch queues. Example files are provided at the end of the chapter.

4.1 Using the Abaqus environment files

The Abaqus execution procedure (see Chapter 3, “Job Execution,” of the Abaqus Analysis User’s Guide) reads environment files to determine the various parameters that are used to run a job.

When Abaqus starts, it searches three directories for the main environment file, **abaqus_v6.env**, in the following order:

1. The *solvers_install_dir/os/SMA/site/* subdirectory of the Abaqus services installation (see “Installation subdirectories,” Section A.1). An **abaqus_v6.env** file must exist in this directory.

The parameter settings in this file are ignored when a job is submitted to a remote queue; in this case, the settings in *solvers_install_dir/os/SMA/site/abaqus_v6.env* on the remote computer are used instead.

Starting in Abaqus 2016, the **abaqus_v6.env** file imports and uses any parameter definitions found in the following auxiliary environment files:

- **custom_v6.env** should contain your site-specific settings such as licensing and documentation parameters
- **basic_v6.env** contains general parameters common to all platforms
- **win86_64.env** or **lnx86_64.env** contain platform-specific parameters such as compiler and MPI settings
- **graphicsConfig.env** contains the **onCaeGraphicsStartup()** function to configure graphics-card-specific settings for Abaqus/CAE and Abaqus/Viewer

You can add your parameter settings directly in the *solvers_install_dir/os/SMA/site/abaqus_v6.env* file; but the recommended best practice is to place your site-specific settings in the *solvers_install_dir/os/SMA/site/custom_v6.env* file.

2. An **abaqus_v6.env** file in the user’s home directory. This environment file is optional and will affect all jobs submitted from the user’s account. Each user should include in this file only the parameters that they specifically want to change. This file should not be a complete copy of the *solvers_install_dir/os/SMA/site/abaqus_v6.env* file.

On Windows platforms, for Abaqus to locate the environment file in the user’s home directory, the full path to the user’s home directory must be specified using the **HOME** environment variable or a combination of the **HOMEDRIVE** and **HOMEPath** environment variables.

3. An **abaqus_v6.env** file in the current working directory. This environment file is optional and will affect all jobs submitted from the current working directory. Each user should include in this file only the parameters that they specifically want to change. This file should not be a complete copy of the *solvers_install_dir/os/SMA/site/abaqus_v6.env* file.

If the same parameter is defined in more than one of these environment files or is defined more than once within a file, the last definition encountered will be used.

An individual user can override the site-wide parameter settings in the *solvers_install_dir/os/SMA/site/custom_v6.env* and *solvers_install_dir/os/SMA/site/abaqus_v6.env* files by creating a new file named **abaqus_v6.env** in their home directory or in the current working directory. Any parameters set in these files will be read last by Abaqus (in the order described above), overriding the site-wide values.

If you are upgrading from an earlier release, do not simply include parameters from the **abaqus_v6.env** file from previous releases of Abaqus without checking that they are necessary.

4.1.1 Environment file syntax

The environment file for Abaqus uses Python syntax. See the Abaqus Scripting User's Guide for more detailed information on Python syntax.

Environment file entries have the following syntax:

```
parameter=value
```

All parameters must have a value. The following syntactic rules also apply:

- All parameters are case sensitive.
- A string value must be enclosed in a pair of double or single quotes.
- Comments are preceded by a number sign (#). All characters following a number sign on a line are ignored. Number signs within a quoted string are part of the string, not the beginning of a comment.
- Blank lines are ignored.
- Lists must be enclosed in square brackets ([]). Individual items in the list are separated by commas. Entries take the form:

```
parameter=[value1, value2, value3]
```

- Tuples must be enclosed in parentheses (()). Individual items in the tuple are separated by commas. If the tuple is enclosed in parentheses and contains only one value, a comma has to follow the value. Entries take the form:

```
parameter=(value1, value2)
```

- Embedded single quotes do not require special handling if they are placed within a double-quoted string. For example, "**my value's**" is translated as **my value's**. The same holds true for double quotes embedded in a single-quoted string. Quotes of the same type as the enclosing quotes

can be embedded if they are prefixed by the backslash (\) character. Strings in a list or a tuple must be enclosed in quotes.

- Triple-quoted (""") strings can span more than one line, and no special treatment of quotes within the string is necessary. Entries take the form:

```
parameter=""
multi-line
value
"""
```

Examples of many of the environment file parameters are available in the sample **abaqusinc.env** file found in the *solvers_install_dir/os/SMA/site/* subdirectory of the Abaqus/CAE installation. Care should be taken when merging customized **abaqus_v6.env** settings from an earlier release into the current release. Settings from the earlier release may not be compatible with the new release.

4.1.2 Memory and disk management parameters

Management of memory and disk resources for Abaqus/Standard and Abaqus/Explicit is discussed in detail in “Managing memory and disk use in Abaqus,” Section 3.4.1 of the Abaqus Analysis User’s Guide. The relevant parameters are listed here along with a single parameter, **abq_ker_memory**, that is used to manage memory in Abaqus/CAE and Abaqus/Viewer.

The available units for memory sizes are mb (megabytes) and gb (gigabytes). If the units are not specified, the size is assumed to be in megabytes.

abq_ker_memory

The maximum amount of memory that can be allocated by the Abaqus/CAE and Abaqus/Viewer kernel, specified in MB (megabytes). If the limit is exceeded, Abaqus/CAE displays an error message.

If the kernel memory size reaches the **abq_ker_memory** value or the virtual memory limit of the machine, the following message will be displayed: **Operation did not complete due to a memory allocation failure.**

For optimal performance, the memory limit should be set to a value less than the physical amount of memory on the machine. The minimum setting allowed is 256 MB.

scratch

Full path name of the directory to be used for scratch files. The default value on Linux is the value of the **\$TMPDIR** environment variable or **/tmp** if **\$TMPDIR** is not defined. On Windows the default value is the value of the **%TEMP%** environment variable or **\TEMP** if **%TEMP%** is not defined. During the analysis a subdirectory will be created under this directory to hold the analysis scratch files. The name of the subdirectory is constructed from the user’s user name, the job ID, and the job’s process number. The subdirectory and its contents are deleted upon completion of the analysis.

memory

Maximum amount of memory or maximum percentage of the physical memory that can be allocated during the input file preprocessing and during the Abaqus/Standard analysis phase. The default value is different on different platforms; for details, refer to the Dassault Systèmes Knowledge Base at www.3ds.com/support/knowledge-base.

4.1.3 Parallelization parameters

Parallelization in Abaqus is discussed in detail in the following sections:

- “Parallel execution in Abaqus/Standard,” Section 3.5.2 of the Abaqus Analysis User’s Guide
- “Parallel execution in Abaqus/Explicit,” Section 3.5.3 of the Abaqus Analysis User’s Guide
- “Parallel execution in Abaqus/CFD,” Section 3.5.4 of the Abaqus Analysis User’s Guide

The relevant parameters are listed here.

auto_convert

If this parameter is set equal to **ON** and an Abaqus/Explicit analysis is run in parallel with **parallel=domain**, the **convert=select**, **convert=state**, and **convert=odb** options will be run automatically at the end of the analysis if required. The default value is **ON**.

cpus

Number of processors to use during an analysis run if parallel processing is available. The default value for this parameter is 1.

domains

The number of parallel domains in Abaqus/Explicit. If the value is greater than 1, the domain decomposition will be performed regardless of the values of the **parallel** and **cpus** variables. However, if **parallel=DOMAIN**, the value of **cpus** must be evenly divisible into the value of **domains**. If this parameter is not set, the number of domains defaults to the number of processors used during the analysis run if **parallel=DOMAIN** or to 1 if **parallel=LOOP**.

gpus

Activate direct solver acceleration using GPGPU hardware in Abaqus/Standard. The value of this parameter should be the number of GPGPUs to use for an analysis. In an MPI-based parallel Abaqus/Standard analysis, this is the number of GPGPUs to use on each host.

max_cpus

Maximum number of processors allowed if parallel processing is available. If this parameter is not set, the number of processors allowed equals the number of available processors on the system.

mp_file_system

Type of file system available for an MPI-based parallel Abaqus analysis. The parameter must be set to a tuple; for example,

```
mp_file_system=(SHARED,LOCAL)
```

The first item in the tuple refers to the directory where the job was submitted, while the second refers to the job's scratch directory. If the file system hosting a directory is **LOCAL**, Abaqus will copy the required analysis files to the remote host machines and, at the end of the run, copy the output files back. In this case it must be possible to create the job's directory structure on all the hosts in **mp_host_list**. A **SHARED** file system means that the host machines share the same file system and file transfer is not necessary. With the recommended default (**DETECT, DETECT**) setting, Abaqus will determine the type of file system that exists. An MPI-based parallel Abaqus/Explicit analysis will use the scratch directory only if a user subroutine is used, whereas Abaqus/Standard normally writes large temporary files in this directory. Running on a local file system will generally improve the performance.

mp_host_list

List of host machine names to be used for an MPI-based parallel Abaqus analysis, including the number of processors to be used on each machine; for example,

```
mp_host_list=[['maple',1],['pine',1],['oak',2]]
```

indicates that, if the number of **cpus** specified for the analysis is 4, the analysis will use one processor on a machine called **maple**, one processor on a machine called **pine**, and two processors on a machine called **oak**. The total number of processors defined in the host list has to be greater than or equal to the number of **cpus** specified for the analysis. If the host list is not defined, Abaqus will run on the local system. When using a supported queuing system, this parameter does not need to be defined. If it is defined, it will get overridden by the queuing environment.

mp_mode

Set this variable equal to **MPI** to indicate that the MPI components are available on the system. Set **mp_mode=THREADS** to use the thread-based parallelization method. The default value is **MPI** where applicable.

mp_mpi_implementation

This variable determines the underlying MPI implementation to use. Generally, this variable does not need to be specified.

mp_mpirun_options

String of options that are passed to the MPI launcher for an MPI-based parallel Abaqus analysis. Generally this variable does not need to be specified.

mp_mpirun_path

A dictionary to define the full path to the MPI launcher for a given MPI implementation. For example, on Windows,

```
mp_mpirun_path={NATIVE: 'C:\\Program Files\\
Microsoft HPC Pack 2008 R2\\bin\\mpiexec.exe'}
```

mp_num_parallel_ftps

When performing parallel file staging using MPI-based parallelization, this parameter controls the number of simultaneous MPI file transfers. The first item controls the transfer of files to and from the temporary scratch directory. The second item controls the transfer of files to and from the analysis working directory. Setting either value to 1 disables the parallel file staging process. The use of file staging depends on the values specified in **mp_file_system**.

mp_rsh_command

Preferred command to open a remote shell on the machines specified by **mp_host_list**. Abaqus needs to open a remote shell to create and remove directories and files if the file system is not shared. The default value for this option is platform dependent; for example,

```
mp_rsh_command='ssh -n -l %U %H %C'
```

The following placemarkers are used:

- %U** Username.
- %H** The host where the remote shell is opened.
- %C** The command to be executed on the host.

Abaqus automatically uses secure copy (**scp**) to copy files to and from remote hosts if this parameter is set to use secure shell. By default, this parameter is ignored in favor of built-in MPI **rsh/scp** commands.

order_parallel

The default direct solver ordering mode in Abaqus/Standard if you do not specify the parallel ordering mode on the **abaqus** command line. If this parameter is set equal to **OFF**, the solver ordering will not be performed in parallel. If this parameter is set equal to **ON**, the solver ordering will be run in parallel. The default for parallel solver ordering is **ON**.

parallel

The default parallel equation solution method in Abaqus/Explicit if the user does not specify the parallel method on the **abaqus** command line. Possible values are **DOMAIN** or **LOOP**; the default value is **DOMAIN**.

standard_parallel

The default parallel execution mode in Abaqus/Standard if you do not specify the parallel mode on the **abaqus** command line. If this parameter is set equal to **ALL**, both the element operations and the solver will run in parallel. If this parameter is set equal to **SOLVER**, only the solver will run in parallel. The default parallel execution mode is **ALL**.

4.1.4 Job customization parameters**abq_cosimulation_lower_port**

This variable specifies the lowest port number in the range of TCP/UDP port numbers available for co-simulation between two Abaqus analyses; it is valid only for Abaqus/CAE. The default value is 48000.

abq_cosimulation_upper_port

This variable specifies the highest port number in the range of TCP/UDP port numbers available for co-simulation between two Abaqus analyses; it is valid only for Abaqus/CAE. If this value is not specified, it is set to 1000 more than **abq_cosimulation_lower_port**.

auto_calculate

If this parameter is set to **ANALYSIS**, all output database postprocessing in Abaqus/Standard is performed as part of analysis execution. If this parameter is set to **ON**, output database postprocessing is performed by the postprocessing calculator at the end of an analysis if the execution procedure detects that output database postprocessing is necessary. If this parameter is set to **OFF**, no output database postprocessing occurs even if the execution procedure detects that it is necessary to postprocess the output database file. The default value is **ANALYSIS**. In Abaqus/Explicit **auto_calculate=ANALYSIS** is equivalent to **auto_calculate=ON**.

average_by_section

This parameter is used only for an Abaqus/Standard analysis. If this parameter is set equal to **OFF**, the averaging regions for output written to the data (**.dat**) file and results (**.fil**) file are based on the structure of the elements. If this parameter is set equal to **ON**, the averaging regions also take into account underlying values of element properties and material constants. In problems with many section and/or material definitions the default value of **OFF** will, in general, give much better performance than the nondefault value of **ON**. See “Output to the data and results files,” Section 4.1.2 of the Abaqus Analysis User’s Guide, for further details on the averaging scheme.

cae_error_limit

This variable defines the maximum number of error messages that will be sent from an analysis job to Abaqus/CAE; it is valid only for Abaqus/CAE. The default value is 50.

cae_no_parts_input_file

This variable defines the format of the input file generated by Abaqus/CAE; it is valid only for Abaqus/CAE. If this variable is set to **ON**, Abaqus/CAE will generate an input file without parts and assemblies. The default value is **OFF**. For more information, see “Writing input files without parts and assemblies,” Section 9.10.4 of the Abaqus/CAE User’s Guide.

cae_warning_limit

This variable defines the maximum number of warning messages that will be sent from an analysis job to Abaqus/CAE; it is valid only for Abaqus/CAE. The default value is 200.

double_precision

The default precision version of Abaqus/Explicit to run if you do not specify the precision version on the **abaqus** command line. Possible values are **EXPLICIT** (only the Abaqus/Explicit analysis is run in double precision), **BOTH** (both the Abaqus/Explicit packager and analysis are run in double precision), **CONSTRAINT** (the constraint packager and constraint solver in Abaqus/Explicit are run in double precision, while the Abaqus/Explicit packager and analysis continue to run in single precision), or **OFF** (both the Abaqus/Explicit packager and analysis are run in single precision). The default is **OFF**.

max_history_requests

This parameter specifies the maximum number of history requests allowed in an Abaqus analysis. The default value is 10,000. History output in Abaqus is intended for relatively frequent output requests for small portions of a model and is displayed in X - Y plots in the Visualization module of Abaqus/CAE (Abaqus/Viewer). Requesting large amounts of history output will cause performance problems in both analysis and postprocessing of an Abaqus job. For vector- or tensor-valued output variables, each component is considered to be a single request. In the case of element variables, history output will be generated at each integration point. For example, requesting history output of the tensor variable S (stress) for a C3D10M element will generate 24 history output requests: (6 components) \times (4 integration points). When requesting history output of vector- and tensor-valued variables, it is recommended that individual components be selected where applicable. In cases where large amounts of history output are required, it is recommended that the data be written to the output database (.odb) as field output from which history data can be extracted using the Visualization module of Abaqus/CAE.

odb_output_by_default

If this parameter is set to **ON**, output database output will be generated automatically. If this parameter is set to **OFF**, output database request keywords must be placed in an input file to obtain output database output and to allow the analysis to be restarted. The default value is **ON**.

onCaeGraphicsStartup

Optional function to be executed before Abaqus/CAE or Abaqus/Viewer begins. This function allows the user to change the graphics options. See “Tuning graphics cards,” Section 5.3, for more information on this function. This function should not normally be changed.

onCaeStartup

Optional function to be executed before Abaqus/CAE begins. See “Customizing Abaqus/CAE startup,” Section 4.3.3, for examples of this function.

onDesignStartup

Optional function to be executed before Abaqus/Design begins.

onJobCompletion

Optional function to be executed after the Abaqus job completes. A function specified in the Abaqus environment file in the current directory will be executed first, followed by the function in the user’s home directory, and then the function in the Abaqus installation environment file. Multiple functions in the same environment file will result in only the last definition being used. See “Job variables,” Section 4.1.11, for a list of variables available to this function.

onJobStartup

Optional function to be executed before the Abaqus job begins. See “Job variables,” Section 4.1.11, for a list of variables available to this function.

printed_output

By default, the values of all *PREPRINT parameters are NO and no results are printed to the data file. Set the **printed_output** parameter equal to **ON** to obtain the same preprint information in the data file as if

```
*PREPRINT, CONTACT=YES, ECHO=YES, HISTORY=YES, MODEL=YES
```

were included in the input file. Setting **printed_output** equal to **ON** can also cause a large volume of tabular results to be printed to the data file (unless printed output control options are used to limit the output). If the input file is in terms of parts and assemblies, setting **printed_output** equal to **ON** will cause the part-assembly map to print out in the data file regardless of the settings on the *PREPRINT option; this allows the user to associate the printed output with the part-assembly defined in the input file. The default value for this variable is **OFF**.

run_mode

Default run mode (**INTERACTIVE**, **BACKGROUND**, or **BATCH**) if the user does not specify the run mode on the **abaqus** command line when running the analysis products. The default value is **BACKGROUND**. This variable should not be set to **BATCH** unless batch queues are defined.

split_dat

If this variable is set to **ON**, the data file will be split into two pieces. The output from the user input processing will be put in a file with a **.pre** extension. The analysis output file will still have a **.dat** file extension. The default value is **OFF**.

unconnected_regions

If this variable is set to **ON**, Abaqus/Standard will create element and node sets in the output database for unconnected regions in the model during a **datacheck** analysis. Element and node sets created with this option are named **MESH COMPONENT N**, where *N* is the component number.

4.1.5 System customization parameters

admin

This parameter prevents unauthorized modification of environment file parameters. Set this parameter equal to a list of environment file parameters that cannot be changed in a lower-level **abaqus_v6.env** file. Unless otherwise noted, all system and job customization parameters can be locked out. Commands in the installation directory have the highest precedence, followed by commands in the user's home directory, and then commands in the current working directory. Thus, an Abaqus user cannot change environment file commands that were locked out by the Abaqus account manager.

ask_delete

If this parameter is set to **OFF**, the user will not be asked whether old job files of the same name should be deleted; the files will be deleted automatically. The default value is **ON**.

compile_cpp

C++ compile command. The command used at SIMULIA is included in the platform-specific environment file (**win86_64.env** or **lnx86_64.env**) in the *solvers_install_dir/os/SMA/site/* subdirectory of the Abaqus services installation. This command should not normally be changed. It may be either a string or a tuple of strings. If the command is a tuple of strings, each string must represent a single command line argument. The values of the placemaker are determined by the Abaqus execution procedure or by the command line options and cannot be modified by the user. The values of the placemaker replace the placemaker in the **compile_cpp** string. The following placemaker is available:

%I Search directories for include files.

compile_fortran

Fortran compile command. The command used at SIMULIA is included in the platform-specific environment file (**win86_64.env** or **lnx86_64.env**) in the

solvers_install_dir/os/SMA/site/ subdirectory of the Abaqus services installation. This command should not normally be changed. It may be either a string or a tuple of strings. If the command is a tuple of strings, each string must represent a single command line argument.

The compilation of Fortran files using Fortran 90 freeform specifications is not supported by default. The **abaqus_v6.env** file in the */SMA/site/* subdirectory of the Abaqus installation includes comments that discuss the compile options.

file_format

Format of results file output (**ASCII** or **BINARY**). This parameter is valid only for Abaqus/Standard. The default value is **BINARY**.

link_exe

Command to link a postprocessing program. The command used at SIMULIA is included in the platform-specific environment file (**win86_64.env** or **lnx86_64.env**) in the *solvers_install_dir/os/SMA/site/* subdirectory of the Abaqus services installation. This command should not normally be changed. It can be either a string or a tuple of strings. If the command is a tuple of strings, each string must represent a single command line argument. The values of the placemarkers are determined by the Abaqus execution procedure or by the command line options and cannot be modified by the user. User-specified external libraries can be linked with the usual link commands. The following placemarkers are used in **link_exe**:

- %J** The job name (in this case the name of the executable to be created).
- %F** The name of the object file created from the user's source file.
- %M** The name of the internally created main object file.
- %B** The name of the shared library of utility functions.
- %O** The list of utility shared libraries for the output database application public interface.
- %L** The list of directories containing shared libraries (HP only).

link_sl

Command to link a shared library. The command used at SIMULIA is included in the platform-specific environment file (**win86_64.env** or **lnx86_64.env**) in the *solvers_install_dir/os/SMA/site/* subdirectory of the Abaqus services installation. This command, once its placemarkers have been exchanged, must be a valid command on the computer system where the shared library is linked. This command can be either a string or a tuple of strings. If the command is a tuple of strings, each string must represent a single command line argument. The values of the placemarkers are determined by the Abaqus execution procedure or by the command line options and cannot be modified by the user. The values of the placemarkers replace the placemarkers in the **link_sl** string. User-specified external libraries can be linked with the usual link commands. The following placemarkers are available:

- %F** The name of the compiled object file.
- %U** The name of the shared library.

CUSTOMIZING THE Abaqus ENVIRONMENT

- %A** The name of the archive of compiled object files to be linked into the shared library.
- %B** The name of the shared library of utility functions.
- %E** The name of the file containing the names of the symbols to be exported from the shared library.
- %L** The list of directories containing shared libraries (HP only).

nodb_cache_limit

Maximum size of the cache in the temporary file directory. Abaqus/CAE uses this cache for local data storage when you use a network ODB connector to read from a remote output database. Set the **nodb_cache_limit** parameter to the number of megabytes to which the cache size will be limited. The minimum value of **nodb_cache_limit** is 500, indicating that the cache size is limited to 500 MB. If you set the maximum cache size to be greater than the available free space, Abaqus/CAE reduces it to a value that is equal to the available free space.

plugin_central_dir

Full pathname of the directory containing Abaqus/CAE plug-ins. In most cases this is a directory at a central location that is accessible to all users at your site. For more information, see Chapter 81, “The Plug-in toolset,” of the Abaqus/CAE User’s Guide.

usub_lib_dir

Full path name of the directory containing optional user-defined libraries of Abaqus/Standard and/or Abaqus/Explicit user subroutines. Valid user subroutine libraries are platform specific, but the file base names are the same for all platforms. The base names are **standardU**, **explicitU**, and **explicitU-D**. Use this variable to avoid the cost of recompiling and/or relinking frequently used user subroutines. The **abaqus make** utility is used to create the shared libraries for use with this variable (see “Making user-defined executables and subroutines,” Section 3.2.18 of the Abaqus Analysis User’s Guide). User libraries created by the **user** option of the Abaqus/Standard and Abaqus/Explicit execution procedure will supersede any user libraries in this directory.

verbose

If this parameter is in the environment file, the execution procedure will print more information on job submission. Possible values are **ON**, **OFF**, **1**, **2**, and **3**. Set the value to **ON** or **1** to print the commands used to run application executables and some performance data. Set the value to **2** to print licensing transaction information. Set the value to **3** to print operating system environment settings. The output associated with the **verbose** parameter is written to standard output. The default value is **OFF**.

The following system customization parameters are used during documentation installation and to specify a web browser to display the Abaqus HTML documentation and context-sensitive help in Abaqus/CAE.

WARNING: *If you customize browser behavior for Firefox or you use a web browser not supported by Abaqus, you may encounter restrictive behavior that prevents the display of the documentation collection window. For example, you may be prompted to select a profile under which to run when attempting to access the HTML documentation with a copy of the web browser already running.*

browser_path

Full path to the web browser executable on Linux platforms. The value of this parameter can be either a string or a list of strings. If a list of strings is specified, the first string must be the full path to the web browser executable and subsequent strings are arguments to customize the browser behavior. If any argument strings are included, at least one of them must contain `%s` for which the full uniform resource locator (URL) to the Abaqus HTML documentation will be substituted.

Abaqus automatically configures supported browsers to correctly display the HTML documentation. The configurations are different for each browser. This parameter can be used in conjunction with **browser_type** to clarify the browser being used. If **browser_path** is set equal to a string and **browser_type** is not set, the system checks the specified browser path for Firefox. If Abaqus does not detect Firefox, Abaqus assumes that an unsupported browser will be used and does not perform an automatic configuration. This parameter is ignored on Windows platforms.

browser_type

Web browser on Linux platforms. To correctly display the HTML documentation, Abaqus automatically configures the browser according to the specified browser type. The possible settings are **FIREFOX** and **CUSTOM_BROWSER**. If you set **browser_type=CUSTOM_BROWSER** to use a web browser other than Firefox, no support or automatic configuration is provided. For more information, see “Using a web browser not supported by Abaqus,” Section 4.3.4.

This parameter can be used in conjunction with **browser_path** to directly specify an executable command for the browser. If **browser_type** is set to Firefox and **browser_path** is not set, the system searches the system path for Firefox. If the specified browser is not found, an error is displayed. This parameter is ignored on Windows platforms.

doc_root

Full uniform resource locator (URL) or path to the Abaqus HTML or PDF documentation. This variable is set during product installation when you provide the documentation URL and should not normally be changed. The required format is one of the following.

- For HTML documentation with a web server:

```
http://computername:port_number/v2016/
```

- For HTML documentation with no web server:

```
file:///doc_install_dir/Documentation/docs/v2016/index.html
```

CUSTOMIZING THE Abaqus ENVIRONMENT

- For PDF-only documentation, or if HTML and PDF were both installed but you prefer PDF, only the path to the installation base directory is required:

doc_install_dir

Setting **doc_root** to this file path and setting **doc_root_type** to '**pdf**' will configure the **abaqus doc** command to open PDF rather than HTML documentation.

doc_root_type

The documentation formats specified by the **doc_root** parameter. Set to the string value '**html**' (the default) for an installation of HTML and PDF, or set to the string value '**pdf**' for PDF-only (or if PDF is preferred over HTML). These values must be lowercase.

doc_resource

Full path to the Adobe Acrobat Reader executable (**acroread**) on Linux platforms. The value of this parameter must be a string. Setting this parameter allows the **abaqus doc** command to open PDF rather than HTML documentation.

The following parameters disallow the execution of the corresponding modules prior to testing for license activation through the startup file. They can be used to provide “friendly” messages if an attempt is made to execute an analysis module for which your site does not have a license. Use these parameters in conjunction with the **admin** parameter to ensure uniformity across your site.

no_aqua

Block execution of Abaqus/Aqua if value is set to **ON**.

no_background

Block background execution of Abaqus analysis jobs if value is set to **ON**.

no_batch

Block batch queue execution of Abaqus analysis jobs if value is set to **ON**.

no_cae

Block execution of Abaqus/CAE if value is set to **ON**.

no_design

Block execution of Abaqus/Design if value is set to **ON**.

no_explicit

Block execution of Abaqus/Explicit if value is set to **ON**.

no_interactive

Block interactive execution of Abaqus analysis jobs if value is set to **ON**.

no_standard

Block execution of Abaqus/Standard if value is set to **ON**.

no_viewer

Block execution of Abaqus/Viewer if value is set to **ON**.

4.1.6 Executable parameters

The Abaqus executables for the licensed modules are placed automatically into the `/code/bin/` subdirectory of the Abaqus installation. Prefix parameters (***_prefix**) can be set to `"time"` on Linux systems to determine the execution time for a particular stage of an Abaqus run.

exe_prefix

Optional executable prefix for all Abaqus analysis executables. The default value is an empty string.

explicit_prefix

Optional executable prefix for Abaqus/Explicit. The value given for this prefix will override the value given for **exe_prefix**.

explicit_dp_prefix

Optional executable prefix for the double precision version of Abaqus/Explicit. The value given for this prefix will override the value given for **exe_prefix**.

package_prefix

Optional executable prefix for the Abaqus/Explicit PACKAGE program. The value given for this prefix will override the value given for **exe_prefix**.

pre_prefix

Optional executable prefix for the Abaqus analysis input file processor. The value given for this prefix will override the value given for **exe_prefix**.

standard_prefix

Optional executable prefix for the Abaqus/Standard program. The value given for this prefix will override the value given for **exe_prefix**.

4.1.7 License management parameters

License management customization parameters control the behavior of the Abaqus license server based on current network conditions and user needs. The Abaqus license server is installed with default parameters that should be suitable for most users. The following parameters are provided for customization:

abaquslm_license_file

This parameter provides the host name of the computer running the Abaqus FLEXnet license server and is set during the product installation. This parameter does not apply to DSLS license servers. If a single server is used, this parameter should be set to *port@license_server_host*, where *port* is the port number and *license_server_host* is the name of the computer running the server. If the port number is between 27000 and 27009, you do not need to include it. If redundant servers are used, the parameter should be set to a list of the names of the redundant server hosts and their ports. For example, if the host names for a set of three redundant servers are maple, pine, and oak, the following parameter definition is appropriate:

```
abaquslm_license_file="27000@maple:27000@pine:27000@oak"
```

Use a colon to separate the host names on Linux platforms, and use a semicolon to separate the host names on Windows platforms.

academic

This parameter indicates whether an academic Abaqus client should use research or teaching license tokens. Setting this parameter to **TEACHING** will force the Abaqus client to use only teaching license tokens. Setting this parameter to **RESEARCH** or removing the parameter will force Abaqus to use only research license tokens. This parameter is set automatically during the product installation: if the license server contains an Abaqus teaching license file, the installation sets the parameter to **TEACHING**; otherwise, the parameter is set to **RESEARCH**.

cae_timeout

The number of minutes that an Abaqus/CAE or Abaqus/Viewer session will remain idle due to no user activity before returning its token to the license server. The default value is 60 minutes.

computer_location

A string that indicates the location of the local client computer. This parameter allows you to subtotal license usage reports by location. The license usage reporting utility compiles and organizes data according to the **computer_location** name. The default value is an empty string. If you do not change this default, the license usage report will not distinguish between different locations in the report.

dsls_license_config

Path to the Dassault Systèmes license server (DSLS) configuration file (**DSLicSrv.txt**). This file determines which Dassault Systèmes license servers to use with Abaqus. For example,

- **Linux platforms:**

```
/opt/simulia/license/DSLicSrv.txt
```

- **Windows platforms:** (must use double backslashes)

```
C:\\SIMULIA\\License\\DSLicSrv.txt
```

For more information about the **DSLicSrv.txt** file, see “Configuring Clients” in the *Dassault Systèmes License Server Installation and Configuration Guide (DSLS.pdf)*.

license_server_type

The type of license server software used by Abaqus clients. Possible values are **FLEXNET** (default) and **DSLS**. See “Dassault Systèmes license server installation,” Section 2.1.4, and “FLEXnet Licensing installation,” Section 2.1.5, for more information.

lmhanglimit

The number of minutes that an Abaqus client will wait in the license queue to obtain licenses if no licenses are currently available. The default value of 0 forces the job to remain in the license queue indefinitely unless it is killed by the user.

lminteractivequeuing

This parameter indicates whether an interactive Abaqus/CAE or Abaqus/Viewer session should queue for a license if one is not available. To allow Abaqus/CAE or Abaqus/Viewer sessions running interactively to queue for a license, set this parameter equal to **ON**. The default value is **OFF**. The **lmlicensequeuing** parameter is used for queuing sessions running without the graphical user interface.

lmlicensequeuing

This parameter indicates whether an Abaqus analysis job or an Abaqus/CAE or Abaqus/Viewer session using the **noGUI** option should queue for a license if one is not available. The default value is **ON**. If this parameter is set to **OFF**, jobs and Abaqus/CAE or Abaqus/Viewer sessions will terminate immediately if a license is not available. The **lminteractivequeuing** parameter is used for queuing sessions running interactively.

lmlog

This parameter indicates whether license usage data should be written to the FLEXnet debug log file. To use the Abaqus license usage reporting utility (see “Reporting FLEXnet license usage data,” Section 3.8), this parameter must be set to **ON**, which is the default value. To suppress license usage data in the debug log file, set this parameter equal to **OFF**.

lmproject

This optional parameter can be used to record information about internal project names or numbers for your company. The **lmproject** parameter can be set to any string value; for example, **lmproject="turbomachinery-project-23"**. This parameter can be set in the environment file in each users’ home directory and can be edited whenever necessary to change to a different project name.

The information about Abaqus license checkouts and associated project names is recorded on the license server and can be retrieved by obtaining historical reports using the accessor **project**. See “Using the **reporttool** utility,” Section 3.10.

lmqueuesleep

The number of seconds that an Abaqus client will wait before checking the license queue to see if enough tokens are free. The default value is 30 seconds, which is the minimum allowed. Increasing this value will decrease network traffic when license queuing occurs.

lmsvrdownlimit

The number of minutes that an Abaqus client will attempt to connect to the license server if the license server is currently unavailable. The default value of 0 forces the job to attempt to connect indefinitely unless it is killed by the user.

4.1.8 Object-oriented batch queue parameter

The object-oriented interface for defining an analysis batch queue is easily customized and extended. An analysis batch queue is available from the Abaqus execution procedure command line once it has been instantiated and inserted into the dictionary of analysis queues. See “Defining analysis batch queues,” Section 4.2, for more information.

queues

Dictionary of queue names and objects. Queue names and their corresponding object instances are inserted into the dictionary as key/value pairs. The special queue name **default** can be used to designate a default queue. The default queue is used when the **run_mode** is set to **BATCH** and the **queue** option is not specified on the command line. It is also used when the **queue** option is specified on the command line but the named queue is blank or does not exist in the **queues** dictionary.

4.1.9 String-based batch queue parameters

The string-based interface for defining an analysis batch queue allows access to a fixed number of options. The parameters below can be used to construct command strings to control the execution of an Abaqus analysis. The only requirement is the string must be a valid command on the computer system where it is executed. See “Defining a string-based analysis batch queue,” Section 4.3.7, for an example.

after_prefix

Optional prefix to be output as part of the submit command when the **after** option is specified on the command line. The default value is an empty string.

queue_cmd

Default command to be used to submit a batch job when the **run_mode** is set to **BATCH**. This parameter, once its placemarkers have been exchanged, must be a valid command on the computer system where it is executed. The values of the placemarkers are defined by input specified on the command line of the Abaqus execution procedure or by environment file parameters. The values of

the placemarkers replace the placemarkers in the **queue_cmd** string. The following placemarkers are available:

- %%** A percent (%) character.
- %A** The **after_prefix** string is substituted for **%A** when the **after** option is specified on the command line.
- %L** The log file name. This token will be replaced by *job-name*.**log**, where *job-name* is the value defined by the **job** command line option.
- %P** The **queue_prefix** string (see description below) is substituted for **%P** when the **queue** option is specified on the command line.
- %Q** The queue name from the **queue** command line option.
- %S** The command script file name. This token will be replaced by *job-name*.**com**, where *job-name* is the value defined by the **job** command line option.
- %T** The time from the **after** command line option.
- %J** The job name from the **job** command line option.
- %E** The token is substituted by the full path to the Python executable.
- %O** The full path to the output directory.

queue_name

List of names for batch commands, typically used for submitting jobs to queues other than the default (defined by **queue_cmd**). Then, elsewhere in the environment file, each of these command aliases must appear on the left side of the equal sign, with the desired command string on the right. This command string has the same format as the **queue_cmd** parameter. It can use the replaceable placemarkers in its construction as long as the final result is a valid system command.

queue_prefix

Optional prefix to be output as part of the submit command when a queue name is specified. The default value is an empty string.

4.1.10 Adams translation units parameters

The **abaqus adams** translator refers to the environment file parameters in this section to specify default values for the units used in the modal neutral (**.mnf**) file it creates. For more information about valid units specifications in ADAMS/View, see “Units” in “Translating Abaqus data to MSC.ADAMS modal neutral files,” Section 3.2.38 of the Abaqus Analysis User’s Guide. These parameters provide the default values only; if you specify a units preference using one of the parameters in the **abaqus adams** execution procedure, your selection overrides the default setting.

adams_units_family

The type of units system for the model. The possible values are **mmks**, **mks**, **cgs**, or **ips**, which correspond to the ADAMS/View options with the same names.

adams_length_units

The length units specified for the model in the modal neutral (**.mnf**) file.

adams_mass_units

The mass units for the model in the modal neutral (**.mnf**) file.

adams_time_units

The time units for the model in the modal neutral (**.mnf**) file.

adams_force_units

The force units for the model in the modal neutral (**.mnf**) file.

4.1.11 Job variables

The following variables can be used in an **onJobStartup** or **onJobCompletion** function:

id

The job identifier specified as the value of the **job** option on the command line.

savendir

The path name to the directory from which the job was submitted.

scremdir

The path name to the scratch directory.

analysisType

The type of analysis to be executed. Possible values are **EXPLICIT** and **STANDARD**.

In addition, for an MPI-based parallel job the following variables are available in an **onJobStartup** or **onJobCompletion** function:

host_list

List of host machine names that were used for the analysis, including the number of processors used on each machine. The format is identical to the **mp_host_list** environment variable (see “Parallelization parameters,” Section 4.1.3).

local_host

List of identifiers used to determine the host machine name from which the job was submitted (e.g., host name, IP address, aliases, etc.).

rsh_command

Command used to open a remote shell on the machines that were used during analysis. The format is identical to the **mp_rsh_command** environment variable (see “Parallelization parameters,” Section 4.1.3).

file_system

Tuple showing the type of file system used for the MPI-based parallel job. The first item in the tuple refers to the directory where the job was submitted, while the second refers to the job’s scratch directory. For MPI-based parallel Abaqus/Explicit analyses that do not use a user subroutine, the scratch directory will remain as **DETECT** unless it was set by the user.

cpus

Number of total processors used for the analysis summed across all host machines.

The following variables are available outside of the **onJobStartup** and **onJobCompletion** functions:

abaqus_version

A string that contains the Abaqus release.

analysisType

The type of analysis to be executed. Possible values are **EXPLICIT** and **STANDARD**.

applicationName

The name of the Abaqus execution procedure to be executed; e.g., **analysis**, **cae**, or **viewer**.

4.2 Defining analysis batch queues

Analysis batch queues are used to configure the way that Abaqus analysis jobs are run. They are particularly useful for integrating Abaqus with third-party batch queueing systems.

Two pieces of information are needed to run an Abaqus job: the syntax of the command used to execute the job and the job-specific information. The command used to execute the job is obtained from a queue definition in the Abaqus environment file. The job-specific information is obtained from the command line options and the analysis parameters defined in the Abaqus environment file. The command line options are described in Chapter 3, “Job Execution,” of the Abaqus Analysis User’s Guide.

The command syntax and the job-specific information are used to construct a command to run an Abaqus job in an analysis queue. An object-oriented interface and a string-based interface are available for defining the analysis batch queues. The object-oriented interface is preferred because it is easily customized and extended. This section discusses the object-oriented interface; see “String-based batch queue parameters,” Section 4.1.9, for information on the string-based interface.

A set of predefined (built-in) queue classes is available within an Abaqus environment file. These classes can be instantiated to create user-defined queue objects. See “Object-oriented batch queue parameter,” Section 4.1.8, for information on the **queues** environment file parameter. In addition to the predefined queue classes, users can create their own queue classes to customize the way analysis jobs are executed. The new queue classes can be derived from the predefined queue classes to minimize coding by the user. The **driverQueues** module must be imported to instantiate a predefined queue or to derive a custom queue class. The predefined classes are described in “Queue class,” Section 4.2.1, and “Predefined derived classes,” Section 4.2.2. In addition, examples that illustrate how to extend predefined classes and create custom queue classes are provided in “Using predefined derived classes,” Section 4.3.5, and “Deriving and using custom queue classes,” Section 4.3.6, respectively.

4.2.1 Queue class

The **Queue** class is an abstract base class. All other analysis batch queue classes are derived from it. The class has no explicit constructor or members. The following methods of the **Queue** class are common to all derived classes:

__repr__(...)

This method returns the class name as a string. The string is printed as a description for the queue when **abaqus information=environment** is executed. Derived classes should override this method to provide a useful description of the queue objects instantiated from them.

createScript(...)

This method creates a Python script named *job-name.com* in the current working directory. This script is used to run the analysis. This method is called by the analysis execution procedure prior to the **submit** method. If the *job-name.com* file cannot be written in the current working directory, a `FileCreationError` exception is raised. The following argument is required:

options A dictionary containing the analysis options.

getDriverName(...)

This method returns the name of the command used to invoke the Abaqus execution procedure.

getPython(...)

This method returns the absolute path to the Abaqus Python interpreter as a string.

getNumRequiredTokens(...)

This method returns the number of license tokens required for an analysis as an integer. The following argument is required:

options A dictionary containing the analysis options.

spawn(...)

This method executes a command in a new process, waits for it to complete, and returns an integer representing the exit status of the command. If the command cannot be executed, a `SpawnError` exception is raised. The following arguments are required:

cmd A valid command string to be executed in the new process. If analyses are submitted to this queue from Abaqus/CAE, the command must return the status immediately; otherwise, the ability to monitor the progress of the analysis jobs in Abaqus/CAE may fail. Examples of commands that return the status immediately are **qsub**, **bsub**, **at**, **batch**, etc.

env A dictionary of environment variables available to the process.

The following argument is optional:

verbose A Boolean specifying whether the command string is printed to **stdout**. The default value is OFF.

submit(...)

This abstract method must be implemented by a derived class. This method is called by the analysis execution procedure to submit the analysis to a queue. The **submit** method must return an integer; a value of 0 indicates success, and a nonzero value indicates failure. When this method is called, the analysis execution procedure supplies the following required arguments:

options A dictionary containing the analysis options.

env A dictionary of environment variables available to the process.

Most derived classes call the **spawn** method from this method and return its exit status.

4.2.2 Predefined derived classes

The following analysis batch queue classes are derived from the **Queue** base class:

AtQueue class

The **AtQueue** class executes an analysis using the Linux **at** command. This class overrides the following base class methods:

__repr__(...)

This method returns a string describing the class.

submit(...)

This method executes the Linux **at** command to run the *job-name.com* analysis script at the time specified on the command line **after** option. If the **after** option is not specified on the command line, a `QueueError` exception is raised.

BatchQueue class

The **BatchQueue** class executes an analysis using the Linux **batch** command. This class overrides the following base class methods:

__repr__(...)

This method returns a string describing the class.

submit(...)

This method executes the *job-name.com* analysis script under the Linux **batch** command.

HoldQueue class

The **HoldQueue** class creates a *job-name.com* file and exits. This class overrides the following base class methods:

__repr__(...)

This method is reimplemented to provide a useful description.

submit(...)

This method prints a message stating that the *job-name.com* script was not submitted and returns a value of 0.

LSFQueue class

The **LSFQueue** class submits an analysis to the LSF queue named when the object was instantiated. If a name was not specified, the analysis is submitted to the default LSF queue. The following constructor argument is optional:

name Name of a valid LSF queue.

This class overrides the following base class method:

submit(...)

This method invokes the LSF **bsub** command to submit the *job-name.com* analysis script to an LSF batch cluster and returns the exit status of the **bsub** command.

NQSQueue class

The **NQSQueue** class submits an analysis to the NQS queue named when the object was instantiated. If a name was not specified, the analysis is submitted to the default NQS queue. The following constructor argument is optional:

name Name of a valid NQS queue.

This class overrides the following base class method:

submit(...)

This method invokes the NQS **qsub** command to submit the *job-name.com* analysis script to an NQS system and returns the exit status of the **qsub** command.

PBSQueue class

Queues instantiated from the **PBSQueue** class will create a *job-name.pbs* script and run the command **qsub job-name.pbs**. The *job-name.inp* and *job-name.com* files will be copied to the execution host, where the *job-name.com* script will be executed. After job completion, all output files will be copied back to the submission host. The following constructor argument is optional:

name Name of a valid PBS queue.

This class overrides the following base class method:

submit(...)

This method invokes the PBS **qsub** command to submit the *job-name.pbs* script to a PBS system and returns the exit status of the **qsub** command.

4.3 Examples

The examples included in this section illustrate various uses of the Abaqus environment file.

4.3.1 Example environment file

An example Windows environment file is shown below. This file will work on Linux systems as well if you change the **scratch** directory setting appropriately. A sample environment file, **abaqusinc.env**, is included in the *solvers_install_dir/os/SMA/site/* subdirectory of the Abaqus/CAE installation to show the options used at SIMULIA.

```
scratch = "c:/temp"
if applicationName in ('analysis', 'datacheck', 'continue') :
    memory = "4 gb"

def onCaeStartup() :
    #Graphics preferences
    session.graphicsOptions.setValues(dragMode=AS_IS, displayLists=ON)
    # Print preferences
    session.printOptions.setValues(vpDecorations=OFF,
        vpBackground=OFF, rendition=COLOR,
        printCommand='lpr -S marley -P hp3')
    session.psOptions.setValues(date=OFF)
    # Job preferences
    def setJobPreferences(module, userData) :
```

```

    session.Queue(name='long', hostName='server',
        queueName='large', directory='/tmp')
addImportCallback('job', setJobPreferences)
# Visualization preferences
def setVisPreferences(module, userData):
    session.defaultOdbDisplay.contourOptions.setValues(
        renderStyle=SHADED, visibleEdges=EXTERIOR,
        contourStyle=CONTINUOUS)
addImportCallback('visualization', setVisPreferences)

```

The default compile and link environment variables for your computer have also been inserted in your site environment file.

4.3.2 Notifying users when a job is completed

The following is an example of how environment file commands can be used to notify Linux system users when their job is finished. The notification method used depends on how the job was run and if the user is logged in. If the job was run interactively, the user will not be notified that the job has finished. If the user is still logged in when the job completes, a message will be output to the screen. If the user has logged out by the time the job completes, a message will be mailed to the user. The syntax of the **mail** command varies from system to system. Please consult your system documentation to determine the appropriate commands.

```

def onJobCompletion():
    import os, re

    userName = os.environ['USER']
    msg = 'Job %s has completed' % id

    # Run "who" command, pipe the output, and read into a list
    whopipe = os.popen('who', 'r')
    output = whopipe.readlines()
    whopipe.close()

    # Find out if the user is logged in
    loggedIn = 'no'
    terminal = [ ]

    for line in output:
        columns = re.split('[ ]+', line) # Split into blank separated columns

```

```

name = columns[0]      # User name is in the first column
if name == userName:
    terminal.append(columns[1]) # Terminal at which user is logged in
    loggedIn = 'yes'

# Use "write" command if the user is logged in, use mail otherwise
if loggedIn == 'no':
    logFile = savedir + id + ".log"
    if os.path.exists(logFile):
        os.system('cat %s | Mail -s "%s" %s' % (logFile, msg, userName))
    else:
        os.system('Mail -s "%s" %s' % (msg, userName))
else:
    for termNum in terminal:
        os.system('echo "%s" | write %s %s' % (msg, userName, termNum))

```

4.3.3 Customizing Abaqus/CAE startup

The following example for the `onCaeStartup` parameter will establish viewport preferences and print options (including a print command), set up a remote execution queue for running Abaqus jobs, and set preferences for contour plots in the Visualization module:

```

def onCaeStartup():

    # Print preferences
    session.printOptions.setValues(vpDecorations=OFF,
        vpBackground=OFF, rendition=COLOR,
        printCommand='lpr -S server -P printer')
    session.psOptions.setValues(date=OFF, logo=OFF)

    def initQueues(*args):
        session.Queue(name='long', hostName='server',
            queueName='large', directory='/tmp')
    addImportCallback('job', initQueues)

    # Visualization preferences
    def setVisPreferences(module, userData):
        import visualization
        session.defaultOdbDisplay.contourOptions.setValues(
            renderStyle=SHADED, visibleEdges=EXTERIOR,

```

```

        contourStyle=CONTINUOUS)
    addImportCallback('visualization', setVisPreferences)

```

Indented text must be valid Python commands. For more queue examples, see Chapter 19, “The Job module,” of the Abaqus/CAE User’s Guide.

4.3.4 Using a web browser not supported by Abaqus

Abaqus provides support only for the Firefox web browser on Linux platforms; however, it does provide the ability to use a web browser not supported by Abaqus for viewing HTML documentation. Examples are shown for setting the **browser_type** and **browser_path** system customization parameters in this situation.

The first example illustrates the parameter settings to specify a web browser not supported by Abaqus:

```

browser_type = CUSTOM_BROWSER
browser_path = ['full_path_to_browser', 'argument1', 'argument2', etc.]

```

where the first string contains the full path to the web browser and subsequent strings are arguments to customize the browser behavior. Refer to the specific web browser documentation for valid arguments.

The second example illustrates the parameter settings to specify the Opera browser (not supported by Abaqus) as the web browser:

```

browser_type = CUSTOM_BROWSER
browser_path = ['/usr/local/bin/opera', '-newwindow' '%s']

```

where **/usr/local/bin/opera** is the full path to the browser and the argument **-newwindow** opens a new browser window using the full uniform resource locator (URL) to the Abaqus HTML documentation. Use the argument **-newpage** to open a new page.

The third example illustrates the parameter settings to specify the Konqueror browser (not supported by Abaqus) as the web browser:

```

browser_type = CUSTOM_BROWSER
browser_path = '/opt/kde3/bin/konqueror'

```

where **/opt/kde3/bin/konqueror** is the full path to the browser. By default, a new browser window opens using the URL to the Abaqus HTML documentation.

4.3.5 Using predefined derived classes

The following example illustrates the instantiation of some predefined derived classes and their insertion into the **queues** dictionary:

```

run_mode = BATCH

```

```

from driverQueues import *
queues['atq'] = AtQueue()
queues['batchq'] = BatchQueue()
queues['hold'] = HoldQueue()

```

To submit an analysis using one of the queues, specify the queue name as the value for the analysis execution procedure **queue** parameter.

4.3.6 Deriving and using custom queue classes

To derive a custom queue class, the **driverQueues** module must be imported and the class must inherit directly or indirectly from the **Queue** class. Derived queues must provide an implementation for the **submit** method. Derived class methods can raise exceptions as needed. The predefined **QueueError** exception is provided as a general-purpose exception.

The following examples illustrate the derivation and use of custom queue classes:

```

run_mode = BATCH
from driverQueues import *
#~~~~~
class NiceQueue(Queue):
    #~~~~~
    def __repr__(self):
        return 'Executes analysis using Linux nice command.'
    #~~~~~
    def submit(self, options, env):
        job = options['job']
        after = options.get('after', '')
        verbose = options.get('verbose', 0)
        if options.get('after', ''):
            # a descriptive string must be supplied as data when
            # raising a QueueError exception
            #~~~~~
            raise QueueError, \
                "after" is not a valid argument for this queue.'
        # run nice under bourne shell to eliminate platform dependencies
        cmd = "/bin/sh -c 'nice %s python ./%s.com 1>./%s.log 2>&1 &' \" \
            % (self.getDriverName(), job, job)
        return self.spawn(cmd, env, verbose)
#~~~~~
class LSF_ResvQueue(LSFQueue):

```

CUSTOMIZING THE Abaqus ENVIRONMENT

```
# For integration with LSF. This queue class supports cpu, memory,
# and license reservations.
#~~~~~
def __init__(self, name, memReserve=0, cpusReserve=0):
    LSFQueue.__init__(self, name)
    self.memReserve = memReserve
    self.cpusReserve = cpusReserve
#~~~~~
def __repr__(self):
    return 'Submits to LSF %s queue (run "bqueues -l %s" for description)' \
           % (self.name, self.name)
#~~~~~
def submit(self, options, env):
    job = options['job']
    verbose = options.get('verbose', 0)
    queue = self.name
    cpus = options.get('cpus', '1')
    if self.cpusReserve:
        cpus = self.cpusReserve
    resLst = [ ]
    # license reservation - For the following line to work, LSF
    # must be configured with a static or dynamic resource called
    # "abqtokens".
    #~~~~~
    resLst.append('abqtokens=%d' % self.getNumRequiredTokens(options))
    # memory reservation
    if self.memReserve:
        from math import ceil
        resLst.append('mem=%d' % int(ceil(self.memReserve/float(cpus))))
    resStr = ''
    if resLst:
        import string
        resStr = '-R rusage[%s]' % string.join(resLst, ':')
    bsub = 'bsub -q %s -J %s -n %s -o %s.log -N %s %s python %s.com' % \
          (queue, job, cpus, job, resStr, self.getDriverName(), job)
    return self.spawn(bsub, env, verbose)

# queue definitions
queues['default']      = NiceQueue()
queues['hold']         = HoldQueue()
```

```

queues['benchmark'] = LSF_ResvQueue(name='benchmark', cpusReserve=16)
queues['dedicated'] = LSF_ResvQueue(name='dedicated')
queues['a500M'] = LSF_ResvQueue(name='a500M', memReserve=500)
queues['a1500M'] = LSF_ResvQueue(name='a1500M', memReserve=1500)
queues['a3000M'] = LSF_ResvQueue(name='a3000M', memReserve=3000)
queues['a6000M'] = LSF_ResvQueue(name='a6000M', memReserve=6000)

```

4.3.7 Defining a string-based analysis batch queue

The following example illustrates the use of the environment file parameters for string-based analysis batch queue definition:

```

try:
    queue_name=list(queue_name)
except:
    queue_name = [ ]
queue_name=queue_name + ["aba_short", "aba_long", "hold "]
after_prefix="-a"
queue_prefix="-q"
queue_cmd="qsub -nr -me %P %Q %A %T -x -eo -o %L %S"
aba_short="qsub -nr -me -q short %A %T -x -eo -o %L %S"
aba_long="qsub -nr -me -q long %A %T -x -eo -o %L %S"
hold="printf 'Job %S not submitted\n' "

```

The **qsub** command used in this example is available only on certain computer systems. Other commands, such as **at** and **batch**, can be used to configure a queuing system on most Linux platforms. Please refer to your system documentation or contact your hardware vendor for information about queuing systems for your platform.

If the queue specified by the **queue** command line option matches one of the queue aliases in the **queue_name** parameter, that **queue** command is used in place of the default command **queue_cmd**. The following are legal command line options for the above example:

```

abaqus job=qt queue=normal
abaqus job=qt queue=aba_short after=10:00
abaqus job=qt queue=hold

```

The first of these three command line options does not match a defined queue, so the **queue_cmd** string is used to submit the job to the **normal** queue. This queue must have been set up by the systems manager prior to submission of the job. The actual command used to send the job to the **normal** queue for execution on Linux platforms is

```

qsub -nr -me -q normal -x -eo -o qt.log qt.com

```

The value of **%A** is not output if **after=time** is not specified on the command line.

The second option uses the string defined by **aba_short**, which submits the job to the system predefined **short** queue. The command executed by the Linux platform is

```
qsub -nr -me -q short -a 10:00 -x -eo -o qt.log qt.com
```

The last command line option creates the file **qt.com**, which contains the Abaqus job commands, and saves it in the current directory. The message **Job qt.com not submitted** is then written to the terminal. The job is not submitted to any queue.

4.3.8 Locking out modifications to environment file parameters

In the example below, all Abaqus jobs will run in batch mode by default, and execution of Abaqus/Aqua jobs is not allowed. The inclusion of the **admin** parameter prevents modification of these settings in lower-level environment files. If this parameter is part of the environment file in the installation directory, the values of **run_mode** and **no_aqua** will override any corresponding values in a user's local directory or command line. Therefore, this example constrains all jobs submitted at your site to run in batch mode.

```
run_mode = BATCH
no_aqua = ON
admin = ['run_mode', 'no_aqua']
```

The **no_aqua** parameter would typically be used to provide a “friendly” message to users if Abaqus/Aqua is not licensed at your site.

5. Customizing Abaqus/CAE and Abaqus/Viewer

This chapter describes user interface customization, printer configuration, and graphics card tuning for Abaqus/CAE and Abaqus/Viewer.

Platform dependencies sometimes exist for Abaqus/CAE and Abaqus/Viewer. These dependencies can change between releases; therefore, they are listed on the **Support** page at www.3ds.com/simulia, where the latest information is published.

5.1 Customizing the user interface

To customize the Abaqus/CAE and Abaqus/Viewer user interface, you can specify general display properties on Windows platforms. Settings on other platforms, such as Linux, may also affect the appearance of the user interface and some of its functions. You can also record all of your actions in the Abaqus/CAE or Abaqus/Viewer user interface in a file named **abaqus.guiLog**.

5.1.1 Hardware acceleration (all platforms)

With some graphics devices Abaqus/CAE and Abaqus/Viewer may fail when hardware acceleration is turned on. It is possible to turn off hardware acceleration if completely necessary, although it is not recommended. Disabling hardware acceleration will severely degrade graphics performance in Abaqus/CAE and Abaqus/Viewer. You can disable hardware acceleration using one of the following methods:

Linux platforms

Start Abaqus/CAE or Abaqus/Viewer using the **-mesa** option:

```
abaqus cae -mesa
abaqus viewer -mesa
```

Windows platforms

There are two ways to disable hardware acceleration on Windows platforms:

- Add the parameter **abaqus_no_hardware_acceleration=ON** to the Abaqus environment file; or
- Create a system environment variable using the following command:

```
set ABAQUS_NO_HARDWARE_ACCELERATION=1
```

5.1.2 Common customizations on Windows platforms

The following procedures explain how to specify some commonly desired settings on Windows platforms:

To change the “start-in” location for any Abaqus shortcut:

1. Use **Windows Explorer** to go to the directory where the Abaqus shortcuts are located. The shortcuts contained in this directory affect all users on the computer and may require special permission to change.
2. Click mouse button 3 on the shortcut (**Abaqus CAE**, **Abaqus Command**, or **Abaqus Viewer**) for which you wish to change the start-in location, and select **Properties**; then click the **Shortcut** tab.
3. In the text box labeled **Start in:**, set the full path to the directory you wish to use as the default location for saving the files created by that Abaqus product.

To stop Abaqus/CAE and Abaqus/Viewer windows from being erased when a dialog box is moved:

Menu items may vary depending upon your Windows operating system. For Windows 7 operating systems:

1. Select **Start→Control Panel**.
2. Type **Effects** in the search box and press **Return**.
3. Select **Adjust the appearance and performance of Windows**.
4. In the **Visual Effects** tab of the **Performance Options** dialog box, toggle off **Show window contents while dragging**.
5. Click **OK** to save your settings and to close the **Performance Options** dialog box.

To change the colors and fonts used in Abaqus/CAE and Abaqus/Viewer:

You can change the colors and fonts displayed in Abaqus/CAE and Abaqus/Viewer by applying a new color scheme to your session. Color schemes determine the colors and text settings that Windows uses to display each component in an application, such as its menus, dialog boxes, and title bar. For example, when the **Windows Standard** color scheme is selected, Windows displays white text in Tahoma font against a blue background in the title bar; and displays black text in Tahoma font on a white background in application menus.

You can also customize a color scheme by editing the color or text settings for any individual item in the color scheme. This customization enables you to change more specific settings, like increasing the text size in the title bar without changing the text size in other areas of the application.

Changes to the color and font settings affect all applications, not just Abaqus programs. Menu items may vary depending upon your Windows operating system. For Windows 7 operating systems:

1. Select **Start→Control Panel**, and type **Display** in the search box.
2. If desired, choose a new font size.
3. Select **Personalization**.

4. Choose a preset **Theme** from the list.
5. If desired, change the color or font settings for individual items in the selected color scheme:
 - a. Type **Window** in the search box. Select **Change window colors and metrics** in the **Personalization** area of the search results.
The **Window Color and Appearance** dialog box appears.
 - b. Select the **Item** for which you want to change color and font display.
The dialog box displays the current color and, if applicable, the font settings for the selected item. Windows excludes font settings for items that do not display text, like the active window border.
 - c. Choose a new item color from the **Color 1** list. For items like the active title bar that allow a gradient between two colors, you can also adjust the second color in the gradient by choosing a new color from the **Color 2** list.
 - d. Adjust the text settings for the selected item from the options at the bottom of the dialog box. You can choose a new **Font** from the list, click the **Size** arrows to increase or decrease text size, choose a new text **Color** from the list, and toggle the bold or italic formatting for the text in this item.
 - e. Repeat the previous two steps to change color and text display for other individual items in a color scheme.
 - f. Click **OK** to close the **Window Color and Appearance** dialog box.
6. Click **OK** to save your settings and to close the **Display Properties** dialog box.

To change the default fonts used in Abaqus/CAE and Abaqus/Viewer:

By default, Windows renders text in the Abaqus/CAE and Abaqus/Viewer viewport windows by referring to the fonts available in your system fonts directory. You can override this default behavior and use other fonts for your session by adding the parameter **hks_font_path** to your Abaqus environment file. Set this parameter to multiple, comma-delimited values if you want to set up multiple font directories for your session.

To display Chinese characters in Abaqus/CAE and Abaqus/Viewer:

You can enable a localized Chinese version of Abaqus/CAE and Abaqus/Viewer. This localized version displays Chinese characters for text in menus, dialog boxes, the Model and Results Trees, and the message area; text in the viewport is not localized.

Menu items may vary depending upon your Windows operating system. For Windows 7 operating systems:

1. Set the regional language settings for Windows to enable Chinese characters:
 - a. Select **Start**→**Control Panel**.

Enabling user interface recording at runtime

Start Abaqus/CAE or Abaqus/Viewer using the `-guiRecord` option:

```
abaqus cae -guiRecord
abaqus viewer -guiRecord
```

Enabling user interface recording using `ABQ_CAE_GUIRECORD`

You can specify user interface recording by setting `ABQ_CAE_GUIRECORD` from a command line, in the Abaqus environment file, or—on Windows platforms—as a system environment variable.

- To set the variable for a single session, enter the following command in a command prompt:

```
set ABQ_CAE_GUIRECORD=1
```

You must run Abaqus/CAE or Abaqus/Viewer from the same command prompt.

- To enable user interface recording as the default behavior, add the parameter `ABQ_CAE_GUIRECORD=ON` to the Abaqus environment file; or on Windows platforms, save it in the Windows system properties. Environment variables can be saved using the **Advanced** tabbed page of the **System Properties** dialog box (select **Start**→**Settings**→**Control Panel**, and double-click **System**).

When user interface recording is the default behavior, you can still disable it from the command line by starting Abaqus/CAE or Abaqus/Viewer using the `-guiNoRecord` option:

```
abaqus cae -guiNoRecord
abaqus viewer -guiNoRecord
```

5.2 Configuring printers

Print commands are not normally needed for Windows platforms. Abaqus/CAE and Abaqus/Viewer automatically detect and list any installed Windows printers. If you encounter problems using the print tool in Abaqus/CAE or Abaqus/Viewer, check the installed Windows printers on your machine and remove any printers that are no longer valid.

This section describes how Abaqus uses print commands. Print commands on Linux platforms should be identical to those used for other applications.

5.2.1 How Abaqus determines and uses the print command

When Abaqus/CAE or Abaqus/Viewer prints to a PostScript printer, it goes through the following steps:

Creating a temporary PostScript file

The temporary PostScript file contains all the PostScript code necessary to describe the page to be printed.

Retrieving the print command specified in the print dialog box

The command specified in the print dialog box can be any command that sends to the printer an unmodified copy of the file whose path is its last argument (some PostScript modifications such as the ones done by PostScript print managers are allowed).

To customize the default print command that is used by Abaqus/CAE or Abaqus/Viewer when it shows the print dialog box, add the following line to the **abaqus_v6.env** file in your home directory or in the site directory of the Abaqus release:

```
def onCaeStartup() :  
    session.printOptions.setValues (printCommand=  
        'print_command_and_arguments_here' )
```

Appending the name of the temporary file and invoking the resulting command

The name of the previously created temporary file is appended to the print command and the PostScript file is printed.

Deleting the temporary PostScript file

If the printer that you are using does not support print spooling, the temporary PostScript file may be deleted before the file is printed. To prevent the temporary PostScript files from being deleted, add the following line to the **abaqus_v6.env** file in your home directory or in the site directory of the Abaqus release:

```
def onCaeStartup() :  
    session.printOptions.setValues (deleteTemporaryFiles=False)
```

5.3 Tuning graphics cards

This section contains the information that you need to configure Abaqus/CAE and Abaqus/Viewer for a graphics adapter that is not yet qualified.

5.3.1 Why is tuning necessary?

SIMULIA tunes and qualifies a limited set of graphics adapters prior to each release. Tuning parameters for these graphics adapters are included in Abaqus. However, new graphics adapters and new drivers for existing graphics adapters become available between releases. Tuning may enable you to take advantage of these new adapters and drivers without waiting for a new release of Abaqus/CAE or Abaqus/Viewer.

Abaqus/CAE and Abaqus/Viewer use OpenGL for high-speed graphics rendering. While the OpenGL standard has strict conformance tests, some features are implementation dependent and require tuning to function correctly. Tuning a new graphics adapter or driver ensures that Abaqus/CAE and Abaqus/Viewer graphics are rendered correctly and that maximum rendering performance is obtained for each system.

You can find the latest information on qualified graphics adapters on the **Support** page at www.3ds.com/simulia. If you read the information on this page and follow the tuning procedures described in this section, you should be able to render Abaqus/CAE and Abaqus/Viewer graphics correctly and with maximum performance. If you continue to experience problems, you should contact your local technical support office for assistance.

5.3.2 How can I tune the parameters?

Abaqus/CAE and Abaqus/Viewer provide the following two methods for tuning graphics parameters:

- Select **View**→**Graphics Options** from the main menu bar. Abaqus displays the **Graphics Options** dialog box shown in Figure 5–1 from which you can select the desired settings. This approach allows you to select from only the most commonly used tuning parameters. The settings in the **Graphics Options** dialog box are described in Chapter 7, “Configuring graphics display options,” of the Abaqus/CAE User’s Guide.
- Use an Abaqus Scripting Interface command to select the desired settings. You can enter the command in the command line interface (CLI) and modify the values of the tuning parameters. This approach provides complete control of all the tuning parameters and is described in “Using the Abaqus Scripting Interface to tune the graphics parameters,” Section 5.3.3. This section also describes how you can use an Abaqus Scripting Interface command to obtain information about the graphics card that you are using.

Table 5–1 shows the tuning parameters that are available from Abaqus/CAE and Abaqus/Viewer. The table also shows the standard value of each parameter and whether you can use the **Graphics Options** dialog box to modify it. Certain parameters can be modified only before starting an Abaqus/CAE or Abaqus/Viewer session (see “Making your graphics configuration permanent,” Section 5.3.4, for details on modifying parameters on startup).

Table 5–1 Tuning parameters.

Parameter	Standard value	Modify using Graphics Options dialog box	Modify only on startup
<i>displayLists</i>	On	Yes	No
<i>antiAlias</i>	On	Yes	No
<i>translucencyMode</i> ²	More accurate than fast	Yes	No
<i>highlightMethod</i> ³	Hardware	Yes	No
<i>highlightMethodHint</i>	Hardware	Yes	No
<i>dragMode</i>	As-is	Yes	No
<i>autoFitAfterRotate</i>	Off	Yes	No

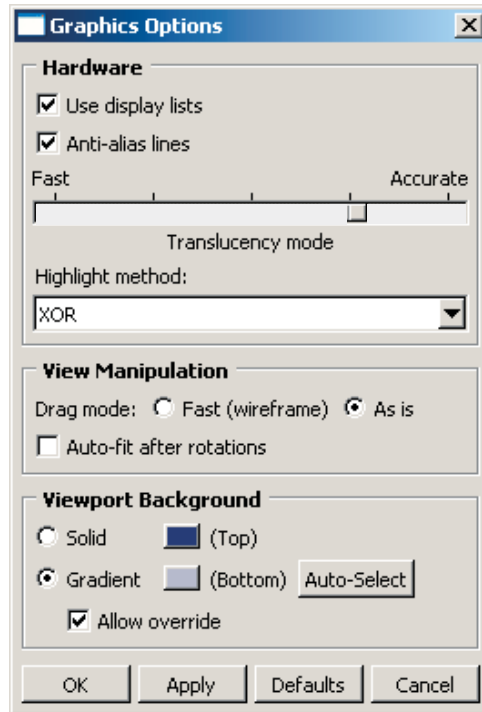


Figure 5–1 The Graphics Options dialog box.

Parameter	Standard value	Modify using Graphics Options dialog box	Modify only on startup
<i>backgroundColor</i>	#333366	Yes	No
<i>backgroundBottomColor</i>	#acacc1	Yes	No
<i>backgroundStyle</i>	Gradient	Yes	No
<i>backgroundOverride</i>	On	Yes	No
<i>doubleBuffering</i>	On	No	No
<i>polygonOffsetConstant</i>	0.0 to 100.0	No	No
<i>polygonOffsetSlope</i>	0.0 to 100.0	No	No
<i>printPolygonOffsetConstant</i>	0.0 to 100.0	No	No
<i>printPolygonOffsetSlope</i>	0.0 to 100.0	No	No
<i>textureMapping</i>	On	No	No
<i>printTextureMapping</i>	On	No	No

Parameter	Standard value	Modify using Graphics Options dialog box	Modify only on startup
<i>vertexArrays</i>	On	No	No
<i>vertexArraysInDisplayLists</i>	On	No	No
<i>backfaceCulling</i>	On	No	No
<i>directRendering</i>	On	No	Yes
<i>accelerateOffScreen</i>	Off	No	Yes
<i>backingStore</i>	On	No	No
<i>hardwareAcceleration</i> ⁴	On	No	Yes
<i>hardwareOverlay</i>	None	No	Yes
<i>hardwareOverlayAvailable</i> ⁵	None	No	N/A
<i>shadersAvailable</i> ⁵	None	No	N/A
<i>viewManipDisplayListThreshold</i> ¹	40	No	No
<i>contourRangeTexturePrecision</i>	5.0×10^{-6}	No	No

¹The threshold is only used in the Visualization module of Abaqus/CAE (Abaqus/Viewer) when display lists are enabled.

²Translucency mode settings for rendering of translucent objects range from 1 (optimized for performance) to 5 (optimized for accuracy). The default value is 4.

³The highlight method is indirectly set by setting the *highlightMethodHint* parameter. Abaqus uses this value to determine an appropriate setting for *highlightMethod*.

⁴Hardware acceleration is applicable only to Windows platforms.

⁵You cannot directly set the *hardwareOverlayAvailable* parameter or the *shadersAvailable* parameter. Abaqus automatically sets these parameters by detecting the available hardware on your system.

5.3.3 Using the Abaqus Scripting Interface to tune the graphics parameters

You can enter Abaqus Scripting Interface commands in the command line interface to tune your graphics parameters and to find out information about the graphics adapter installed on your system. This section explains how to use the Abaqus Scripting Interface to modify the graphics options; the Abaqus Scripting Interface is described in detail in the Abaqus Scripting User's Guide.

In general, you should use the default values for most of the parameters. However, Abaqus provides the capability to modify parameters to fix the following specific problems:

- The *hardwareAcceleration* parameter controls a number of different graphics tuning parameters and generally should not be modified. Hardware acceleration options are discussed in “Hardware acceleration (all platforms),” Section 5.1.1.
- The *hardwareOverlay* parameter is controlled by the *hardwareOverlayAvailable* parameter. If your system supports hardware overlay planes, Abaqus/CAE and Abaqus/Viewer will use them by default. If your system supports hardware overlay planes but viewports display a solid color and will not display a model, you may need to manually set *hardwareOverlay=OFF*.
- The *contourRangeTexturePrecision* parameter sets the tolerance used when computing the appropriate scale for transforming result (contour) values to texture values. When set too low, the “out of range” colors may be shown incorrectly for values near the range limits.
- Some graphics adapters do not support the use of textures to generate contour plots properly. If you experience problems displaying contour plots (for example, all contours appear gray or the system aborts), you need to set *textureMapping=OFF* to emulate texture mapping in software. Similarly, if you experience problems printing contour plots, you need to set *printTextureMapping=OFF*.
- Some graphics adapters do not fully support the use of vertex arrays to process information about vertices. Some specific problems indicate that vertex arrays are not fully supported: when you drag the radius of a circle in the Sketcher, the circle is not visible; when you display an *X-Y* plot, the axis labels are not visible; and some facets in the shaded display of a mesh are missing. If you experience any of these problems, set *vertexArraysInDisplayLists=OFF*. If this does not resolve the problem, suppress the use of vertex arrays altogether by setting *vertexArrays=OFF*.
- The *backfaceCulling* parameter controls the display of facets that are determined to be facing away from the viewer. If the front sides of elements appear to be missing in the display or if the display is incomplete, set *backfaceCulling=OFF*.
- You can disable direct rendering (set *directRendering=OFF*) for Linux systems that do not behave correctly when accessing the graphics hardware directly.
- You can disable hardware-accelerated off-screen rendering (set *accelerateOffScreen=OFF*) when you want printed images to be rendered without OpenGL hardware acceleration or if you experience problems with the Probe functionality in the Visualization module of Abaqus/CAE (Abaqus/Viewer).
- You can disable the backing store (set *backingStore=OFF*) when you want to conserve memory. When *accelerateOffScreen=ON*, the memory for the backing store is allocated from memory on the graphics card. When *OFF*, the memory for backing store is allocated from system memory. The backing store is generated by rendering the viewport to an off-screen area. Subsequent viewport refreshes are performed more quickly by copying the off-screen area to the viewport window. Even when *backingStore=ON*, the backing store will not be created if the viewport can be redrawn sufficiently quickly.
- The *translucencyMode* parameter determines whether Abaqus/CAE optimizes the rendering of translucent objects for performance, accuracy, or for a level in between. Lower values provide better performance, while higher values provide greater accuracy.

- The *polygonOffsetConstant* and *polygonOffsetSlope* parameters, which affect onscreen display, require manual tuning for each graphics adapter. On Linux systems the *printPolygonOffsetConstant* and *printPolygonOffsetSlope* parameters can generally be set equal to the same values as the corresponding onscreen display parameters. On Windows systems the *printPolygonOffsetConstant* and *printPolygonOffsetSlope* parameters do not generally need to be adjusted.
- The *viewManipDisplayListThreshold* parameter can be lowered if there is an unacceptable delay when initiating view manipulation operations in the the Visualization module. Increasing this value may increase the delay for large models but should produce improved graphics performance during the view manipulation. If set high with a large model, the delay can be many seconds and in excessive cases may exceed system graphics memory and result in an empty display (no visible model) for the view manipulation.

You can tune the graphics parameters using the following Abaqus Scripting Interface objects:

- **GraphicsOptions**: The members of the **GraphicsOptions** object determine the current graphics settings. These settings can be modified during a session using the **setValues** method. The arguments to the **setValues** method are described in “setValues,” Section 17.9.1 of the Abaqus Scripting Reference Guide.

You can view the current settings of the graphics parameters by entering the following command in the command line interface:

```
print session.graphicsOptions
```

The following output is typical:

```
{'accelerateOffScreen': OFF,
  'antiAlias': ON,
  'autoFitAfterRotate': OFF,
  'backfaceCulling': ON,
  'backgroundBottomColor': '#acacc1',
  'backgroundColor': '#333366',
  'backgroundOverride': ON,
  'backgroundStyle': GRADIENT,
  'backingStore': ON,
  'contourRangeTexturePrecision': 5.0e-06
  'directRendering': ON,
  'displayLists': ON,
  'doubleBuffering': ON,
  'dragMode': AS_IS,
  'graphicsDriver': OPEN_GL,
  'hardwareAcceleration': ON,
  'hardwareOverlay': OFF,
  'hardwareOverlayAvailable': False,
  'highlightMethod': SOFTWARE_OVERLAY,
```

```
'highlightMethodHint': (HARDWARE_OVERLAY,
SOFTWARE_OVERLAY, XOR, BLEND),
'polygonOffsetConstant': 2.0,
'polygonOffsetSlope': 0.75,
'printPolygonOffsetConstant': 1.0,
'printPolygonOffsetSlope': 0.75,
'printTextureMapping': ON,
'shadersAvailable': True,
'stencil': False,
'textureMapping': ON,
'translucencyMode': 3,
'vertexArrays': ON,
'vertexArraysInDisplayLists': ON,
'viewManipDisplayListThreshold': 40})
```

Note: Some of the parameters listed above have been deprecated. For information on accessing deprecated parameters, see “BackwardCompatibility object,” Section 53.4 of the Abaqus Scripting Reference Guide.

The following command uses the `setValues` method to modify some members of the GraphicsOptions object:

```
session.graphicsOptions.setValues(autoFitAfterRotate=ON,
dragMode=AS_IS)
```

You can enter this command in the command line interface.

- GraphicsInfo: The members of the GraphicsInfo object provide information about the graphics adapter installed on your system. This information may be useful for determining how to tune the graphics adapter. The members are described in “GraphicsInfo object,” Section 17.8 of the Abaqus Scripting Reference Guide. The members are for reference only and cannot be modified.

You can view the graphics information by entering the following command in the command line interface:

```
print session.graphicsInfo
```

The following output is typical on Windows platforms:

```
{'glRenderer': 'Quadro FX 560/PCI/SSE2',
'glVendor': 'NVIDIA Corporation',
'glVersion': (2, 0, '.3'),
'glxClientVendor': None,
'glxClientVersion': (None, None, None),
'glxServerVendor': None,
'glxServerVersion': (5, 1, None)}
```

Tuning the `polygonOffsetConstant` and `polygonOffsetSlope` parameters

If display lists are enabled, you will not see the effect of tuning these parameters; therefore, you must toggle off **Use display lists** before attempting to tune your graphics adapter. Alternatively, you can enter the following command in the command line interface:

```
session.graphicsOptions.setValues(displayLists=OFF)
```

Setting drag mode to `AS_IS` is helpful for fine tuning the parameters. Rotating the view interactively will show you if minor adjustments are necessary.


```
session.graphicsOptions.setValues(dragMode=AS_IS)
```

It is recommended that you tune *polygonOffsetConstant* first, then tune *polygonOffsetSlope*. To tune these parameters, you should first display the part that is generated by the example script in “Creating a part,” Section 3.1 of the Abaqus Scripting User’s Guide. To retrieve the script, use the following command in a command prompt window (operating system shell):

```
abaqus fetch job=modelAExample
```

Select **File**→**Run Script** from the main menu bar, select the example script from the **Run Script** dialog box that appears, and click **OK**. The example script creates a new viewport; however, Abaqus should display only one viewport while you are trying to tune the graphics parameters. Select any old viewports and delete them by selecting **Viewport**→**Delete Current** from the main menu bar.

To tune the `polygonOffsetConstant` parameter:

1. From the **Views** toolbar, apply the bottom view .
2. In the command line interface, enter the following commands:

```
session.graphicsOptions.setValues(polygonOffsetSlope=0.0)  
session.graphicsOptions.setValues(polygonOffsetConstant=0.0)
```

3. Display the bottom view again to refresh the display.
4. Examine the model for visible lines. If all lines are not visible, repeat Step 2, increasing the value of the polygon offset constant by a small increment; for example, `0.1`. The normal range for this parameter is between `0.5` and `1.5`, and two decimal places usually provide sufficient precision. You should attempt to find a value as small as possible that produces a correct display. Values that are too large may cause the lines to appear to float above the part. The following figures illustrate the lines that should be visible in your model.

Figure 5–2 illustrates the model with an incorrect value for *polygonOffsetConstant*; some lines are missing between the shaded areas.

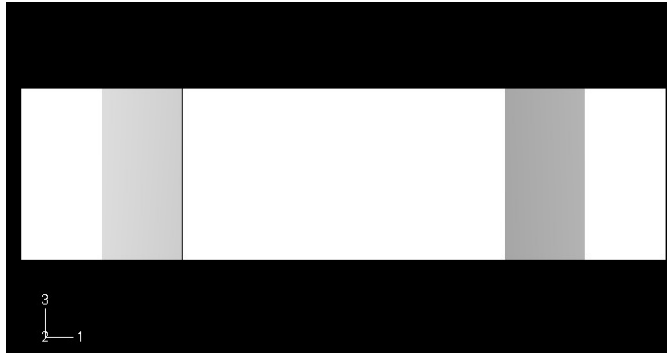


Figure 5–2 Incorrect value for *polygonOffsetConstant*.


Figure 5–3 illustrates the model with a correct value for *polygonOffsetConstant*; all the shaded areas are separated by lines.



Figure 5–3 Correct value for *polygonOffsetConstant*.

After you have derived a value for *polygonOffsetConstant*, you can tune *polygonOffsetSlope*.

To tune the *polygonOffsetSlope* parameter:

1. From the **Views** toolbar, apply the isometric view . This view shows edges at a 45° angle on at least one axis.
2. In the command line interface, enter the following command:
`session.graphicsOptions.setValues(polygonOffsetSlope=1.0)`
3. Examine the model for visible lines. If all lines are not visible, repeat Step 2, increasing the value of the polygon offset slope by a small increment; for example, **0.1** or **0.05**. The

normal range for this parameter is between **1.0** and **2.0**, and two decimal places usually provide sufficient precision. If the *polygonOffsetConstant* value is too low, it may force the *polygonOffsetSlope* to be high. High values of *polygonOffsetSlope* may cause the edges of hidden polygons to show through where they meet visible polygons. In this case, raise the *polygonOffsetConstant* value to get an acceptable value for *polygonOffsetSlope*.

Figure 5–4 illustrates the model with an incorrect value for *polygonOffsetSlope*; some line segments are missing between the shaded areas, there is a line missing inside the hole, and some lines appear dashed when they should appear solid.

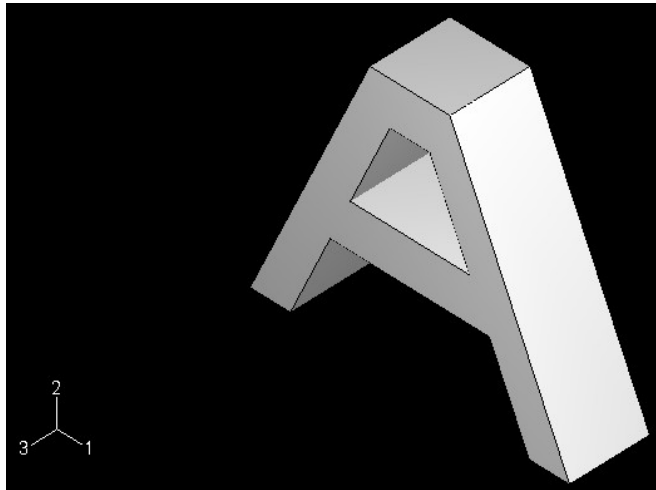


Figure 5–4 Incorrect value for *polygonOffsetSlope*.

Figure 5–5 illustrates the model with a correct value for *polygonOffsetSlope*; all the shaded areas are separated by solid lines.

Test the tuned values of *polygonOffsetConstant* and *polygonOffsetSlope* on several models to make sure the values are satisfactory. When you have finished tuning the graphics parameters, you should return your settings for display lists and drag mode to the original values.

When you are satisfied with the parameter settings, you should modify the environment file as described in “Making your graphics configuration permanent,” Section 5.3.4.

5.3.4 Making your graphics configuration permanent

Once you are satisfied with the values you have specified for the tuning parameters, you can make the changes permanent by including an **onCaeGraphicsStartup** function in your environment file (**custom_v6.env** or **abaqus_v6.env**). To avoid conflicts with other graphics settings, you should

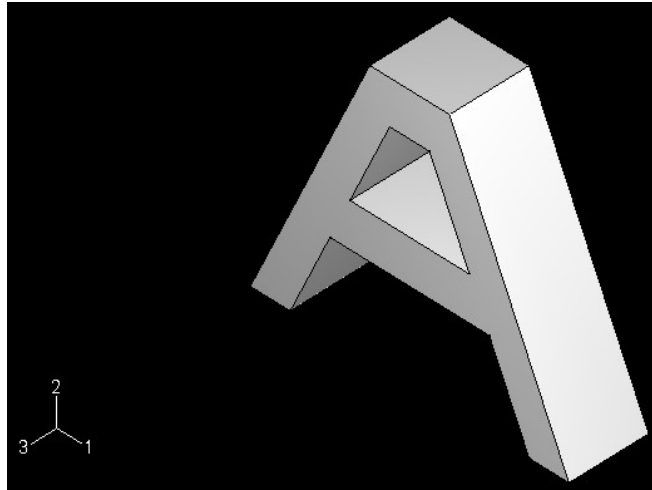


Figure 5–5 Correct value for *polygonOffsetSlope*.

add the customized **onCaeGraphicsStartup** function only to the environment file in your home directory (see “Using the Abaqus environment files,” Section 4.1, for details on environment file location and execution).

The members of the **DefaultGraphicsOptions** object determine the default graphics settings that are enabled when you start a session and when you click **Defaults** in the **Graphics Options** dialog box. You can view the default graphics settings by entering the following command in the command line interface:

```
print session.defaultGraphicsOptions
```

You use the **setValues** method in the environment file (**abaqus_v6.env**) to modify the members of the **DefaultGraphicsOptions** object. To set your default graphics options in the environment file, you must use the **session.defaultGraphicsOptions** object instead of the **session.graphicsOptions** object that you modified from the command line interface. The following example environment file configures your Abaqus/CAE and Abaqus/Viewer graphics settings:

```
def onCaeGraphicsStartup() :  
  session.defaultGraphicsOptions.setValues (  
      polygonOffsetConstant=1.0,  
      polygonOffsetSlope=1.2)
```

Appendix A: Abaqus release directories and files

The directory structure for Abaqus 2016 is different from the structure used in Abaqus 6.14 (and earlier).

A.1 Installation subdirectories

There are two separate software installations required for the Abaqus products. This guide refers to the base installation directories as:

solvers_install_dir

This directory contains the Abaqus services: Abaqus/Standard solver, Abaqus/Explicit solver, Abaqus/CFD solver, SIMULIA Co-Simulation Engine, and output database API.

cae_install_dir

This directory contains Abaqus/CAE.

The following subdirectories will be created in either the *solvers_install_dir* or *cae_install_dir*:

<i>cae_install_dir/os/CAEresources/</i>	Configuration files.
<i>cae_install_dir/os/code/</i> <i>solvers_install_dir/os/code/</i>	Abaqus executables and command procedures, third-party executables, and libraries.
<i>solvers_install_dir/os/code/include/</i>	Links to header files used for building postprocessing programs with the abaqus make utility.
<i>cae_install_dir/os/code/bin/</i>	The Abaqus/CAE executable.
<i>solvers_install_dir/os/code/bin/</i>	The Abaqus solvers' executables and command procedures.
<i>cae_install_dir/os/SMA/samples/</i>	Input files and directories associated with the installed products, including installation verification problems, timing test problems, files for the Introduction to Abaqus seminar, files for the Getting Started tutorial guides, files for application briefs, files used with the abaqus findkeyword utility, and files from the Example Problems, Benchmarks, and Verification Guides.

solvers_install_dir/os/SMA/site/

Site-specific files: the Abaqus environment files (**abaqus_v6.env** and **custom_v6.env**), a sample environment file (**abaqusinc.env**), include files (*file_name.inc*), a data file containing information used in the verification procedure (**chksum.dat**), and the platform computer tables.

cae_install_dir/os/tools/

Python interpreter.

cae_install_dir/os/code/python2.7/lib/abaqus_plugins/

Example Python modules for Abaqus/CAE plug-ins.

A.2 Command-line batch files and links

The installer creates a **Commands** directory containing links (Linux) to, or batch files (Windows) for, the driver command to run Abaqus.

See “Using Abaqus commands,” Section 2.1.2, for details.

A.3 Documentation files

The following directories will be created under the documentation installation, depending on which of the installation options are chosen:

Documentation	Default directory for all files associated with documentation.
installation_info	Directory containing the command to restart the web server and log files from the documentation installation procedure and Windows uninstaller (if used).
docs	Directory containing release-specific book files for an HTML plus PDF documentation installation.
pdf_docs	Directory containing release-specific book files for a PDF-only documentation installation.

An illustration of the documentation directory structure is shown in Figure A–1.

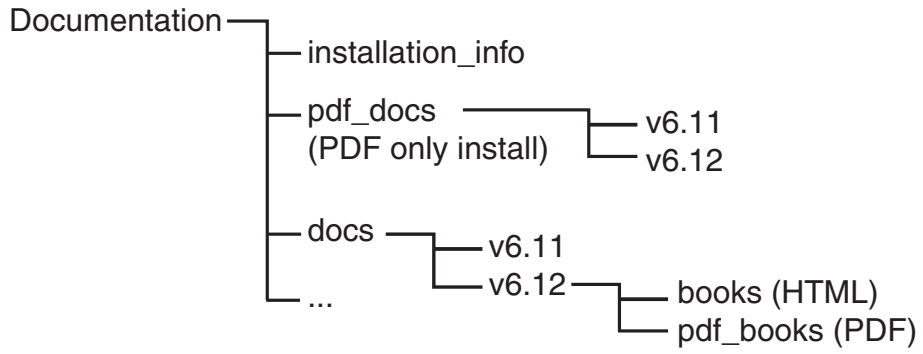


Figure A-1 Documentation installation.

Appendix B: Configuring Abaqus documentation

Abaqus HTML documentation is installed separately from the licensing and product installations and is viewed through a web browser. It is recommended that you install and start the Abaqus web server to serve the HTML (and PDF) files to your users' web browsers. You can install HTML documentation without a web server; however, in this case no search functionality is available. For information on the documentation installation options, see “Abaqus documentation installation details,” Section 2.2.

This appendix contains information related to configuring the web server and Abaqus HTML documentation. You must configure the Abaqus HTML documentation if you use an existing non-Abaqus web server during the HTML documentation installation procedure, if you install Abaqus products before installing Abaqus HTML documentation, or if you do not specify the URL for HTML documentation during the product installation.

For the latest information on supported web servers and configuring existing non-Abaqus web servers, see the Dassault Systèmes Knowledge Base at www.3ds.com/support/knowledge-base.

Note: When you install the Abaqus HTML documentation, the PDF files are also installed. Both the HTML and PDF versions are available through a web browser, via a link from the HTML collection home page (**`index.html`**) to the PDF collection index file (**`index.pdf`**).

B.1 Using an existing non-Abaqus web server

When you install Abaqus documentation and you choose to use an existing non-Abaqus web server, you must configure your server manually to access the HTML (and PDF) documentation. If you want to access the HTML documentation from the **Start** menu (Windows platforms only), you must modify the **`documentation.htm`** file after you configure the server (see “Setting the documentation URL in the **`documentation.htm`** file,” Section B.1.2).

B.1.1 Configuring an existing non-Abaqus web server

You can configure an existing non-Abaqus web server to access the Abaqus HTML documentation using the following procedures. For the latest information about non-Abaqus web server configurations, see “How to install Abaqus HTML documentation using an existing web server” in the Dassault Systèmes Knowledge Base at www.3ds.com/support/knowledge-base.

Linux platforms

1. Install the Abaqus HTML documentation in the SIMULIA parent directory (see “Installation procedures,” Section 2.1).

APPENDIX B: CONFIGURING Abaqus DOCUMENTATION

2. Create an alias in your web server's configuration file for the root directory of the HTML documentation. For example, on an Apache web server you must add

```
Alias /v2016 "doc_install_dir/Documentation/docs/v2016"
```

where *doc_install_dir* refers to the HTML documentation installation parent directory.

3. Create an alias in your web server's configuration file for the Taxis executable. For example, on an Apache web server you must add

```
ScriptAlias /taxis/ "doc_install_dir/Documentation/scripts/taxis/"
```

4. Restart your web server to reread the web server's configuration file.

5. Access the Abaqus HTML documentation with the following URL:

```
http://computername:port_number/v2016
```

where *computername* is the fully qualified domain name of your web server and *port_number* is the port number that your web server is running on (*:port_number* needs to be specified only if your web server is not running on the default port of 80).

Windows platforms

1. Install the Abaqus HTML documentation in the SIMULIA parent directory. The recommended location for the SIMULIA parent directory is **C:\SIMULIA**.
2. Create an alias in your web server's configuration file for the root directory of the HTML documentation. For example, on an Apache web server you must add

```
Alias /v2016 "doc_install_dir/Documentation/docs/v2016"
```

where *doc_install_dir* refers to the HTML documentation installation parent directory.

3. Create an alias in your web server's configuration file for the Taxis executable. For example, on an Apache web server you must add

```
ScriptAlias /taxis/ "doc_install_dir/Documentation/scripts/  
taxis.exe/"
```

4. Restart your web server to reread the web server's configuration file.

5. Access the Abaqus HTML documentation with the following URL:

```
http://computername:port_number/v2016
```

where *computername* is the fully qualified domain name of your web server and *port_number* is the port number that your web server is running on (*:port_number* needs to be specified only if your web server is not running on the default port of 80).

6. To access the Abaqus HTML documentation from the **Start** menu, see the instructions in "Setting the documentation URL in the **documentation.htm** file," Section B.1.2.

B.1.2 Setting the documentation URL in the `documentation.htm` file

After you configure an existing non-Abaqus web server on Windows platforms, you can modify the following file to access the Abaqus HTML documentation from the **Start** menu:

```
doc_install_dir/Documentation/installation_info/v2016/documentation.htm
```

where `doc_install_dir` refers to the HTML documentation installation parent directory. The line in the `documentation.htm` file should have the following form:

```
<META HTTP-EQUIV="Refresh" CONTENT="0;
  URL=http://computername:port_number/v2016">
```

where `computername` is the fully qualified domain name of your web server and `port_number` is the port number that your web server is running on (`:port_number` needs to be specified only if your web server is not running on the default port of 80). For example, a valid parameter definition is

```
<META HTTP-EQUIV="Refresh" CONTENT="0;
  URL=http://velouria.xyz.com:2080/v2016">
```

B.2 Setting the documentation URL after installation

If you install Abaqus products before installing Abaqus HTML documentation or if you do not specify the URL for HTML documentation during the product installation, you must add the URL to the `abaqus_v6.env` and `url.js` files.

B.2.1 Setting the `doc_root` parameter in the `abaqus_v6.env` file

The documentation URL is required in the `abaqus_v6.env` file to provide access to the HTML documentation from Abaqus. The `abaqus_v6.env` file is located in the `solvers_install_dir/os/SMA/site/` subdirectory. The required parameter definition is as follows:

- HTML and PDF documentation installation with web server

```
doc_root="http://computername:port_number/v2016"
```

where `computername` is the fully qualified domain name of the web server and `port_number` is the port number used by the web server (if required). For example, a valid parameter definition is

```
doc_root="http://willow.xyz.com:2080/v2016"
```

- HTML and PDF documentation installation with no web server

```
doc_root="file:///doc_install_dir/Documentation/docs/
  v2016/index.html"
```

where *doc_install_dir* is the HTML documentation installation parent directory. For example, valid parameter definitions are

Linux platforms

```
doc_root="file:///usr/local/SIMULIA/Documentation/docs/
v2016/index.html"
```

Windows platforms

```
doc_root="file:///d:/SIMULIA/Documentation/docs/
v2016/index.html"
```

- PDF-only documentation installation (no web server). In this case you must set the values of two parameters as follows:

```
doc_root_type="pdf"
doc_root="doc_install_dir"
```

where *doc_install_dir* is the path to the installation base directory.

B.2.2 Setting the documentation URL in the `url.js` file

The documentation URL is required in the `url.js` file to enable links from help windows in Abaqus/CAE to associated topics in the Abaqus HTML documentation. The `url.js` file is located in the *cae_install_dir/os/CAEhelp/redirects/* directory. The line in the `url.js` file should have the following form:

- HTML documentation installation with web server

```
var baseURL="http://computername:port_number/v2016/books/";
```

where *computername* is the fully qualified domain name of the web server and *port_number* is the port number used by the web server (if required). For example, a valid definition is

```
var baseURL="http://willow.xyz.com:2080/v2016/books/";
```

- HTML documentation installation with no web server

```
var baseURL="file:///doc_install_dir/Documentation/docs/
v2016/books/";
```

where *doc_install_dir* is the HTML documentation installation parent directory. For example, valid definitions are

Linux platforms

```
var baseURL="file:///usr/local/SIMULIA/Documentation/docs/
v2016/books/";
```

Windows platforms

```
var baseURL="file:///d:/SIMULIA/Documentation/docs/  
v2016/books/";
```


Appendix C: Accessing remote file systems for installation and execution

This appendix describes accessing remote file systems for Abaqus installation and execution.

C.1 Running Abaqus remotely on Linux

It is possible to run Abaqus on NFS-mounted file systems or to export the display from a remote computer; however, users may experience performance and reliability problems with these alternatives.

C.1.1 NFS-mounted file systems

There are several scenarios for running Abaqus on NFS-mounted file systems; the impact of using NFS-mounted file systems varies with the method of NFS use. The most common scenarios for running Abaqus on NFS-mounted file systems are as follows:

- Abaqus is installed on a remote file system. The CPU, save directory (see “Job variables,” Section 4.1.11), and scratch directory are local (on the computer where Abaqus will be run).
In this scenario Abaqus executables and shared libraries are loaded into local memory as they are needed across the network from the file system where Abaqus is installed. Processing and output occur locally. If there is sufficient local memory to prevent frequent paging of the code, network traffic will be relatively light. When local memory is insufficient to prevent paging, performance will suffer and reliability may be impacted.
- Abaqus is installed on a remote file system. The save directory and/or scratch directory is/are not local to the CPU where Abaqus is executed.
In this scenario program files and data used during execution are written across the network. Potentially large volumes of data will be transferred across the network, and performance may be impacted adversely. Abaqus performance may be extremely slow, and other users of the network may be affected because the network could become saturated. In addition, Abaqus does not trap file errors arising from NFS, so a failure in accessing NFS-mounted files, even temporary, will cause the Abaqus job to fail. The use of Abaqus in this configuration should be avoided whenever possible. If this configuration is required for an Abaqus/Standard job, the user should move the save directory to the NFS-mounted file system before moving the scratch directory.

C.1.2 Exporting the display

This configuration is relevant only for Abaqus/CAE and Abaqus/Viewer. In this configuration Abaqus is installed on a remote file system and the CPU, save directory, and scratch directory are located on the same remote computer, which is accessed through remote login. All processing occurs on the remote computer, and only output messages or graphics are exported to the local computer and display. This

method is known to cause performance problems. Minor incompatibilities between OpenGL and GLX libraries can introduce significant graphics problems and, in some cases, can prohibit Abaqus/CAE or Abaqus/Viewer from running. The use of Abaqus in this configuration is not supported and should be avoided whenever possible.

C.2 Using a network ODB connector

Users can create a network ODB connector to access an output database on a remote computer (see “Accessing an output database on a remote computer,” Section 9.3 of the Abaqus/CAE User’s Guide). Abaqus/CAE or Abaqus/Viewer can start the server on the remote system and assign port numbers if the following are true:

- The user name on the remote host is the same as the user name on the local system.
- The remote shell command (**rsh**) or the secure shell command (**ssh**) is configured so that it does not prompt the user for a password.

Abaqus checks the security of the connection by passing a key back and forth between the server and the client. For information about how the key is generated, see the Dassault Systèmes Knowledge Base at www.3ds.com/support/knowledge-base. If a file called **.abaqus_net_passwd** is present in the user’s home directory on the remote system, Abaqus uses the password in the file for authentication instead of the key generated by Abaqus. Abaqus checks that the user is the only account with permission to read and write to the password file. In addition, the user must update the file after 30 days, and the password must be at least eight characters long. These files are described in “Network output database file connector,” Section 3.2.26 of the Abaqus Analysis User’s Guide.

If users experience problems establishing communication or if the user names are different, they can start the network ODB server manually from the command line using the **abaqus networkDBConnector** execution procedure on the remote computer. Abaqus uses the password in **.abaqus_net_passwd** to authenticate the connection between the client and the server if the user starts the network ODB server manually. If **.abaqus_net_passwd** does not exist in the user’s directory on the remote system, the user cannot start the network ODB server from the command line.

You can disable network odb connectors by removing **dmbwtr** and **dmbwtrd** from the *solvers_install_dir/os/code/bin/SMAExternal/dmbwtr/* directory.

Appendix D: Verification procedure

The verification procedure checks the installation of all licensed Abaqus products and reports on the success or failure of verification for each product. The verification procedure runs automatically after the Abaqus installer has finished, but only a subset of the products are verified. The verification procedure can also be run as an Abaqus command option.

The procedure runs verification problems for each licensed product and compares the results to reference values. The command line options are not affected by license type; that is, the verification procedure attempts to verify all products named in the command. Before the verification procedure is run, licensing requirements are checked for the selected product(s). If a teaching academic license is detected, the verification procedure that is run during installation checks only Abaqus/Standard, Abaqus/Explicit, and Abaqus/CAE.

Product verification is skipped if the product is not licensed. The command line option **-NoLicenseCheck** can be used to bypass these checks. The verification problems for all Abaqus products are extracted automatically from the disk during the installation.

To run the verification procedure from the command line:

Run the procedure by typing the following command:

```
abaqus verify [-adams -all -ams -tosca -cae -catiaV4 -catiaV5
-cfd -contextHelp -cPerf -dcatiaV5 -design -docUrl -exp -foundation
-help -install -ioPerf -log -make -moldflow -noGui
-NoLicenseCheck -parallel -param -parasolid -proe
-retainFiles -scripting -std -swi -user_exp -user_std
-verbose -viewer]
```

Common options

-all	Verify all licensed products. All other verification options except log and NoLicenseCheck are ignored.
-help	Print summary of verify usage.
-install	Verify only the main Abaqus products. Abaqus/Standard with user subroutines and Abaqus/Explicit with user subroutines will be tested only if your system has a Fortran compiler available.
-log	Direct all output to a file named verify.log in the current working directory.

Product options

-ams	Verify Abaqus/AMS.
-tosca	Verify Tosca for Abaqus.

APPENDIX D: VERIFICATION PROCEDURE

-cae	Verify Abaqus/CAE.
-cfd	Verify Abaqus/CFD.
-design	Verify Abaqus/Design.
-exp	Verify Abaqus/Explicit.
-foundation	Verify Abaqus/Foundation.
-param	Verify parametric studies in Abaqus.
-std	Verify Abaqus/Standard.
-user_exp	Verify Abaqus/Explicit with user subroutines.
-user_std	Verify Abaqus/Standard with user subroutines.
-viewer	Verify Abaqus/Viewer.

Translator options

-adams	Verify Abaqus Interface for MSC.ADAMS.
-catia4	Verify Geometry Translator for CATIA V4.
-catia5	Verify Geometry Translator for CATIA V5.
-dcatia5	Verify Direct Geometry Import for CATIA V5 (Direct Geometry Import is a component of the CATIA V5 Associative Interface). Verifies geometry import capability in Abaqus/CAE; does not verify installation or functionality of the CATIA V5 Associative Interface plug-in.
-moldflow	Verify Abaqus Interface for Moldflow.
-parasolid	Verify Geometry Translator for Parasolid.
-proe	Verify Geometry Translator for Pro/ENGINEER (the Geometry Translator is a component of the Pro/ENGINEER Associative Interface). Verifies geometry import capability in Abaqus/CAE; does not verify installation or functionality of the Pro/ENGINEER Associative Interface plug-in.

-swi Verify Geometry Translator for SolidWorks (the Geometry Translator is a component of the SolidWorks Associative Interface). Verifies geometry import capability in Abaqus/CAE; does not verify installation or functionality of the SolidWorks Associative Interface plug-in.

Additional options

-contextHelp Verify Abaqus/CAE context-sensitive help functions.

-cPerf Verify Abaqus/CAE performance.

-docUrl Verify Abaqus HTML documentation URL.

-ioPerf Verify I/O performance.

-make Verify the **abaqus make** utility.

-noGui Verify the **-noGUI** option for Abaqus/CAE and Abaqus/Viewer.

-NoLicenseCheck Requires **-all**, **-install**, or a list of product options. Run the verification procedure for the specified products but bypass all licensing checks. The procedure attempts to verify all selected products regardless of licensing.

-parallel Verify Abaqus analysis jobs using parallelization.

-scripting Verify the Abaqus scripting interface.

-retainFiles Retain all verification files in the **verify** directory (by default, the files are deleted after a successful verification).

-verbose Include additional details for debugging purposes.

If a product selected for verification is not licensed, an Abaqus warning is displayed for the selected product and the verification of other selected products continues.

Reviewing and resolving verification procedure failures

If the verification procedure finishes successfully, all files and the **verify** directory are removed (unless you use the **retainFiles** option). Error diagnostics for all products that fail verification remain in the **verify** directory. It is very important that you review these error messages.

The **verify** directory and results can be found in the *cae_install_dir/InstallDate/logs/.../tmp/* directory.

The following suggestions may help you to correct common installation errors that cause the verification procedure to fail:

- Make sure that the license file was installed properly. If there are problems with the license file, error messages will be written to standard output.
- Make sure that you have not tried to execute Abaqus products for which you are not licensed.

APPENDIX D: VERIFICATION PROCEDURE

- Make sure that the operating system and compiler level are consistent with those specified for this release in the Program Directory. (See <http://media.3ds.com/support/progdir>. Choose SIMULIA Abaqus as the product **Line** and Abaqus 2016 as the **Level**, then choose **Prerequisites** in the left-hand pane.)

If the error messages and these suggestions are insufficient to verify the installation, check for information about installation problems, resolutions, and verification in the Dassault Systèmes Knowledge Base at www.3ds.com/support/knowledge-base.

If you are still unable to resolve the problem, contact your local office or representative for help. An overview of support options is available in the Dassault Systèmes Knowledge Base at www.3ds.com/support/knowledge-base.

Appendix E: Troubleshooting Abaqus FLEXnet licensing

This appendix addresses questions about FLEXnet network licensing. For more information on troubleshooting FLEXnet licensing, you can refer to the resources in the Dassault Systèmes Knowledge Base at www.3ds.com/support/knowledge-base.

E.1 General questions

This section addresses some general questions about Abaqus FLEXnet licensing.

E.1.1 Are there limits to network size for FLEXnet network licensing to work?

The size of a network is limited by the requirement that all computers running Abaqus must be able to connect directly and regularly with the FLEXnet license server. If communication with the server takes too long, the client will spend more time communicating with the server than it does processing. Failure to communicate regularly will result in clients losing tokens and waiting while they are reacquired. The most basic test of network communication is the **ping** command. If a client and server cannot ping each other by name and IP address, network licensing will not work between the two computers. (However, the fact that the **ping** command functions properly does not necessarily guarantee that network licensing will work.)

E.1.2 Must there be a name server running for FLEXnet network licensing to work?

Some form of host name resolution is required to set up network licensing. If there is no NIS or DNS name server functionality on your network, then host files must be manually configured on each machine. For more information, see “DNS name resolution” in the Dassault Systèmes Knowledge Base at www.3ds.com/support/knowledge-base.

E.1.3 How can I tell if a FLEXnet license file is installed?

The license file for Abaqus is usually installed in a directory called *flex_install_dir/os/code/bin/* and is typically named **simulia1m.lic**.

E.1.4 What is the FLEXnet license server log file, and where should it be stored?

The server log file records all the license requests for Abaqus jobs and sessions (clients). Usually the file is called **simulia1m.log**.

E.1.5 What is the FLEXnet license options file, and where should it be stored?

The options file allows the license administrator to control various operating parameters of FLEXnet Licensing, such as reserving a number of tokens for certain users, computers, or domains on a network or restricting access to Abaqus license tokens. Although it can be stored anywhere and can be given any name you like, the recommended location and name for the options file is *flex_install_dir/SIMULIADM.opt*. If you do not want to restrict access to Abaqus license tokens, you do not need a license options file.

After creating or modifying a license options file, you must use the **lmreread** tool to reread the license file and implement the new options (see “lmreread,” Section 3.6.6). The options file is also read when the license server is started. Some changes to the options file will not take immediate effect unless the license server is restarted.

A sample options file is shown in “Using the FLEXnet options file,” Section 3.7, and instructions for creating an options file are included in Chapter 5 of the FLEXnet Licensing End User Guide Version 11.6, listed under **Licensing** on the **Support** page at www.3ds.com/simulia.

E.2 FLEXnet license server operation

This section addresses questions that relate to the operation of the FLEXnet license server. The answers to these questions usually involve running one of the FLEXnet administration tools.

E.2.1 How do I update the license file for our FLEXnet license server?

To update a FLEXnet license file, you should terminate your license server, replace the old license file with the new license file, and restart the license server. If necessary, you can also update the license file without stopping the license server so that there will be no interruption to Abaqus users. Both techniques are described in detail in “Updating an Abaqus FLEXnet license file,” Section 3.2.

E.2.2 How can I tell if the FLEXnet license server is running?

The easiest way to see if a license server is running is to execute the command

```
abaqus licensing lmstat
```

This command will list the license servers that are running for a general broadcast. To see if a license server is running on a specific computer, you can enter the command

```
abaqus licensing lmstat -c [port]@license_server_host
```

where *license_server_host* is the hostname of the license server that you want to query. If the license server is using a port that is not between 27000 and 27009, you must specify the port number as well.

E.2.3 How can I restart the FLEXnet license server if it is not running?

You must login on the license server host to restart the server (on Windows platforms you must login as Administrator) and then run the **lmgrd** program (see “FLEXnet license server manager lmgrd,” Section 3.5). For an alternate method, see “Starting the FLEXnet server using **LMTTOOLS**,” Section 3.6.9.

E.2.4 How can I terminate an Abaqus FLEXnet license server?

You use the FLEXnet Licensing administration tool **lmdown** to terminate an Abaqus license server. If you use **lmdown** to terminate a license server that is part of a redundant license server configuration, all three license servers in the redundant configuration will be terminated. If you do not want to terminate all three servers, you must use an operating system utility, such as the **ps** and **kill** commands (Linux platforms) or the Windows **Task Manager**, to terminate the license server. Do not use the **-9** option when using the **kill** command. For more information, see “lmdown,” Section 3.6.2.

E.2.5 What if the job or session loses contact with the FLEXnet license server before it completes?

The client enters a state of “suspended animation.” The client will attempt to reestablish contact with the license server until a user-defined expiration time is exceeded (the default is to continue checking perpetually until contact is reestablished). The minimum expiration time is 60 minutes. If the predefined expiration time has not been set, the client may be terminated by using the standard **kill** command (Linux platforms) for the process. On Windows platforms the client can be terminated by entering **<Ctrl>-C** in the DOS window where the client was started if the job was run in interactive mode; otherwise, use the **Task Manager**.

E.2.6 What happens to FLEXnet licenses if the client ends abnormally?

Tokens are normally returned immediately. If the tokens are not returned immediately, ensure that the job is not running and stop the executable for the analysis by using **abaqus terminate** or the appropriate operating system utility, such as the **ps** and **kill** commands (Linux platforms) or the Windows **Task Manager**.

Do not use the **lmremove** tool to free license tokens used for analysis jobs, as it will not have the intended effect. You can use **lmremove** to free Abaqus/CAE or Abaqus/Viewer license tokens; you must first obtain token information using the **lmstat** tool (for an example, see “lmstat,” Section 3.6.7).

E.2.7 Can I see where my job is in the FLEXnet license queue?

No, only log file information is available. When an analysis job is in the license queue, information about the queue status is written to the log or status file for the job every 30 seconds. Abaqus/CAE and Abaqus/Viewer interactive sessions are not queued.

E.2.8 Can I see which users are using the Abaqus FLEXnet licenses?

The command `flex_install_dir/os/code/bin/lmstat -a` will list users and the license features that they are using. For more information, see “lmstat,” Section 3.6.7.

E.2.9 Do I have to update the FLEXnet license server when I install a new release of Abaqus?

Yes, if the new release of Abaqus uses a later version of FLEXnet Licensing than your existing license server. Abaqus 2016 uses FLEXnet Licensing Version 11.6.1. If you have a version of FLEXnet Licensing earlier than Version 11.6.1 installed on your system, you should use the licensing installer to automatically upgrade the license server. If you have an existing license server that uses a version of FLEXnet Licensing higher than Version 11.6.1, you should use the procedure described in “Updating an Abaqus FLEXnet license file,” Section 3.2, to update your license file without overwriting the existing license server. For more information, see “FLEXnet Licensing installation,” Section 2.1.5. If your Abaqus license file is integrated with FLEXnet license files for other products using a combined license file, you must ensure that the **lmgrd** version meets the FLEXnet version compatibility rules (listed in Appendix I of the FLEXnet Licensing End User Guide Version 11.6).

Starting with Abaqus 6.13, license keys will continue to work for two years of subsequent releases, so it may not be necessary to install new FLEXnet license keys with every new release.

You can check under **Licensing** on the **Support** page at www.3ds.com/simulia for the FLEXnet Licensing End User Guide.

E.3 Special situations

This section addresses questions concerning special operations that usually occur infrequently.

E.3.1 How do we add redundant servers if our network has a single FLEXnet license server?

You need to decide which other computers you wish to use as redundant license servers. The total number of redundant license servers is three, and reliable communication must be maintained at all times between the master server and at least one other backup server (shadow) for licensing to operate. The three server hosts must be on the same subnet.

1. Send the `lmhostid` for all three license server hosts to your local office or representative, and request a license file for redundant servers.
2. Install the redundant license file and start the Abaqus license server manually as described in “Redundant FLEXnet license server configurations,” Section 3.4.

3. For each installation of Abaqus, edit the Abaqus environment file (`abaqus_v6.env`) so that the line

```
abaquslm_license_file="[port]@license_server_host"
```

becomes

```
abaquslm_license_file="[port]@serverhost1:[port]@serverhost2:
[port]@serverhost3"
```

where *serverhost1*, *serverhost2*, and *serverhost3* are the names of the computers where the license servers are running (quotes are required). If a license server is using a port that is not between 27000 and 27009, you must include the port number with the server host name. Items in the list are colon-separated on Linux platforms and semicolon-separated on Windows platforms. Each entry must begin with an “@” symbol or a port number and an “@” symbol. Make sure that users make similar changes to any environment files in their home and local directories.

E.3.2 How can I tell if I am using a redundant FLEXnet server configuration?

View your Abaqus license file (`simulia.lic`) located in the directory `flex_install_dir/os/code/bin/`. If you are using a redundant server configuration, there will be three **SERVER** lines in the license file. The server names should be different for each line, although the port number may be the same for all three. For example:

```
SERVER bifrost.simulia.com 69084992 27003
SERVER tank.simulia.com 69094954 27003
SERVER willow.simulia.com 69094344 27003
VENDOR ABAQUSLM
```

E.3.3 What if I run the analysis job in a batch queue?

FLEXnet licenses are not checked out until the analysis actually begins, so tokens will not be held by a job in a batch queue. However, if a job executes in a batch queue and sufficient tokens are not available, the job will be placed in the license queue until the tokens are free, which will prevent the rest of the jobs in the queue from executing. To prevent wasted time in the batch queue, it is recommended that tokens be registered as a resource in third-party batch queuing systems. See the LSF example for token resource in “Deriving and using custom queue classes,” Section 4.3.6.

E.3.4 What is the difference between the license queue and a batch queue?

The batch queue controls how and when jobs are run on a certain computer (see “Defining analysis batch queues,” Section 4.2). The license queue is created by the license server when Abaqus jobs have

requested more tokens than are currently available. A job will not request tokens until it reaches the front of the batch queue and begins execution.

E.3.5 If we have multiple, nonredundant network license servers, how do I connect to a specific server?

The situation with multiple, nonredundant license servers is significantly different from the one with redundant servers. With redundant servers one acts as the master and the others are backups (shadows) in case the master goes down. With multiple, nonredundant servers each one is independent of the others. To connect to a specific server, you must set the **abaquslm_license_file** parameter in a local Abaqus environment file (**abaqus_v6.env**) either in your home directory or in the current working directory. The proper format for the parameter definition is as follows:

```
abaquslm_license_file=" [port]@license_server_host"
```

where *license_server_host* is the name of the computer on which the license server is running (the quotation marks should be included). If a license server is using a port that is not between 27000 and 27009, you must include the port number with the server host name. Multiple servers may be listed in the **abaquslm_license_file** parameter as follows:

```
abaquslm_license_file=" [port]@server1: [port]@server2:  
[port]@server3: [port]@server_n"
```

Abaqus jobs will try to connect to the servers in the order that they are listed.

E.3.6 Are there any special requirements for using FLEXnet Licensing within a DHCP-configured network?

The hostname on the server line of the license file must be the computer name, not the IP address. The client must refer to the license server by hostname in the **abaquslm_license_file** parameter in an Abaqus environment file (**abaqus_v6.env**).

E.4 Other licensing questions and problems

If you are still experiencing problems with Abaqus FLEXnet network licensing and you have checked all the foregoing FAQs and searched the knowledge database of answers and solutions to questions that we have answered, please submit a request for systems support. Instructions for submitting this request are available through “Initial procedure for reporting system support related problems” in the Dassault Systèmes Knowledge Base at www.3ds.com/support/knowledge-base. If you do not have access to this document, contact your local support office to obtain a copy of the systems support request form.

About SIMULIA

Dassault Systèmes SIMULIA applications, including Abaqus, Isight, Tosca, and Simulation Lifecycle Management, enable users to leverage physics-based simulation and high-performance computing to explore real-world behavior of products, nature, and life. As an integral part of Dassault Systèmes' **3DEXPERIENCE** platform, SIMULIA applications accelerate the process of making highly informed, mission-critical design and engineering decisions before committing to costly and time-consuming physical prototypes. www.3ds.com/simulia

Our 3DEXPERIENCE Platform powers our brand applications, serving 12 industries, and provides a rich portfolio of industry solution experiences.

Dassault Systèmes, the **3DEXPERIENCE** Company, provides business and people with virtual universes to imagine sustainable innovations. Its world-leading solutions transform the way products are designed, produced, and supported. Dassault Systèmes' collaborative solutions foster social innovation, expanding possibilities for the virtual world to improve the real world. The group brings value to over 170,000 customers of all sizes in all industries in more than 140 countries. For more information, visit www.3ds.com.

